

Yue Hao et al. (Eds.)

LNAI 3801

Computational Intelligence and Security

International Conference, CIS 2005
Xi'an, China, December 2005
Proceedings, Part I

1
Part I

 Springer

Lecture Notes in Artificial Intelligence 3801

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Yue Hao Jiming Liu Yuping Wang
Yiu-ming Cheung Hujun Yin
Licheng Jiao Jianfeng Ma
Yong-Chang Jiao (Eds.)

Computational Intelligence and Security

International Conference, CIS 2005
Xi'an, China, December 15-19, 2005
Proceedings, Part I

Volume Editors

Yue Hao

E-mail: yhao@xidian.edu.cn

Jiming Liu

E-mail: jiming@comp.hkbu.edu.hk

Yuping Wang

E-mail: ywang@xidian.edu.cn

Yiu-ming Cheung

E-mail: ymc@comp.hkbu.edu.hk

Hujun Yin

E-mail: hujun.yin@manchester.ac.uk

Licheng Jiao

E-mail: lchjiao@mail.xidian.edu.cn

Jianfeng Ma

E-mail: jfma@mail.xidian.edu.cn

Yong-Chang Jiao

E-mail: ychjiao@xidian.edu.cn

Library of Congress Control Number: 2005937069

CR Subject Classification (1998): I.2, H.3, H.4, H.5, F.2.2, I.4

ISSN 0302-9743

ISBN-10 3-540-30818-0 Springer Berlin Heidelberg New York

ISBN-13 978-3-540-30818-8 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

springeronline.com

© Springer-Verlag Berlin Heidelberg 2005

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper SPIN: 11596448 06/3142 5 4 3 2 1 0

Preface

The International Conference on Computational Intelligence and Security (CIS) is an annual international conference that brings together researchers, engineers, developers and practitioners from both academia and industry to share experience and exchange and cross-fertilize ideas on all areas of computational intelligence and information security. The conference serves as a forum for the dissemination of state-of-the-art research and the development, and implementations of systems, technologies and applications in these two broad, interrelated fields. This year CIS 2005 was co-organized by the IEEE (Hong Kong) Computational Intelligence Chapter and Xidian University, and co-sponsored by Hong Kong Baptist University, National Natural Science Foundation of China, Key Laboratory of Computer Networks and Information Security of the Ministry of Education of China, and Guangdong University of Technology. CIS 2005 received in total 1802 submissions from 41 countries and regions all over the world. All of them were strictly peer reviewed by the Program Committee and experts in the field. Finally, 337 high-quality papers were accepted yielding an acceptance rate of 18.7%. Among them, 84 papers are the extended papers and 253 are the regular papers. The conference was greatly enriched by a wide range of topics covering all areas of computational intelligence and information security. Furthermore, tutorials and workshops were held for discussions of the proposed ideas. Such practice is extremely important for the effective development of the two fields and computer science in general.

We would like to thank the organizers: the IEEE (Hong Kong) Computational Intelligence Chapter and Xidian University for their great contributions and efforts in this big event. Thanks also go to the sponsors, the Institute of Electrical and Electronics Engineers (IEEE), Hong Kong Baptist University (HKBU), National Natural Science Foundation of China, Key Laboratory of Computer Networks and Information Security of the Ministry of Education of China, Guangdong University of Technology (GDUT), and the publisher, Springer, for their unremitting support and collaboration to make CIS 2005 possible and successful. Furthermore, we would like to sincerely thank the Program Committee members and additional reviewers for their professional, efficient input to the review process. Last but not the least, the Organizing Committee is much appreciated for their enormous efforts and marvelous work.

October 2005

Yue Hao and Jiming Liu
General Co-chairs of CIS 2005
Yuping Wang, Yiu-ming Cheung
Hujun Yin and Licheng Jiao
Program Committee Co-chairs of CIS 2005
Jianfeng Ma and Yong-Chang Jiao
Organizing Committee Co-chairs of CIS 2005

Organization

CIS 2005 was organized by IEEE (Hong Kong) Computational Intelligence Chapter and Xidian University.

General Co-chairs

Yue Hao	Xidian University, China
Jiming Liu	Hong Kong Baptist University, Hong Kong, China

Steering Committee Chair

Yiu-ming Cheung	Hong Kong Baptist University, Hong Kong, China
-----------------	---

Organizing Committee

Jianfeng Ma	Xidian University, China (Co-chair)
Yong-Chang Jiao	Xidian University, China (Co-chair)
Hailin Liu	Guangdong University of Technology, China (Tutorial and Workshop Chair)
Hecheng Li	Xidian University, China (Treasurer)
Lixia Han	Xidian University, China (Publicity Chair)
Yuen-tan Hou	Hong Kong Baptist University, Hong Kong, China (Publicity Chair)
Liang Ming	Xidian University, China (Registration Chair)
Yuanyuan Zuo	Xidian University, China (Local Arrangement Chair)
Shuguang Zhao	Xidian University, China (Publication Chair)
Kapluk Chan	Nanyang Technological University, Singapore (Asia Liaison)
Yan Wu	Xidian University, China (Secretary)
Jingxuan Wei	Xidian University, China (Secretary)
Rongzu Yu	Xidian University, China (Web Master)

Program Committee

Yiu-ming Cheung (Co-chair)	(Hong Kong, China)
Licheng Jiao (Co-chair)	(China)

VIII Organization

Yuping Wang (Co-chair) (China)
Hujun Yin (Co-chair) (UK) Michel Abdalla (France)
Khurshid Ahmad (UK)
Francesco Amigoni (Italy)
Sherlock Au (Hong Kong, China)
Dunin-Keplicz Barbara (Poland)
Mike Barley (New Zealand)
Chaib-draa Brahim (Canada)
Tony Browne (UK)
Scott Buffett (Canada)
Matthew Casey (UK)
Dario Catalano (France)
Kapluk Chan (Singapore)
Keith Chun-chung Chan (Hong Kong, China)
Michael Chau (Hong Kong, China)
Sheng Chen (UK)
Songcan Chen (China)
Zheng Chen (China)
Xiaochun Cheng (UK)
William Cheung (Hong Kong, China)
Sungzoon Cho (Korea)
Paolo Ciancarini (Italy)
Stelvio Cimato (Italy)
Helder Coelho (Portugal)
Carlos Coello (USA)
Emilio Corchado (Spain)
Wei Dai (Australia)
Joerg Denzinger (Canada)
Tharam Dillon (Australia)
Tom Downs (Australia)
Richard Everson (UK)
Bin Fang (China)
Marcus Gallagher (Australia)
Matjaz Gams (Slovenia)
John Qiang Gan (UK)
Joseph A. Giampapa (USA)
Maria Gini (USA)
Eric Gregoire (France)
Heikki Helin (Finland)
Tony Holden (UK)
Vasant Honavar (USA)
Mike Howard (USA)
Huosheng Hu (UK)
Yupu Hu (China)
Marc van Hulle (Belgium)
Michael N. Huhns (USA)
Samuel Kaski (Finland)
Sokratis Katsikas (Greece)
Hiroyuki Kawano (Japan)
John Keane (UK)
Alvin Kwan (Hong Kong, China)
Kwok-Yan Lam (Singapore)
Loo Hay Lee (Singapore)
Bicheng Li (China)
Guoping Liu (UK)
Huan Liu (USA)
Zhe-Ming Lu (Germany)
Magdon-Ismail Malik (USA)
Xiamu Niu (China)
Wenjiang Pei (China)
Hartmut Pohl (Germany)
Javier Lopez (Spain)
V. J. Rayward-Smith (UK)
Henry H.Q. Rong (Hong Kong, China)
Guenter Rudolph (Germany)
Patrizia Scandurra (Italy)
Bernhard Sendhoff (Germany)
Michael Small (Hong Kong, China)
Vic Rayward Smith (UK)
Fischer-Huebner Simone (Sweden)
Stephanie Teufel (Switzerland)
Peter Tino (UK)
Christos Tjortjis (UK)
Vicenc Torra (Spain)
Kwok-ching Tsui (Hong Kong, China)
Bogdan Vrusias (UK)
Bing Wang (UK)
Ke Wang (Canada)
Haotian Wu (Hong Kong, China)
Gaoxi Xiao (Singapore)
Hongji Yang (UK)
Shuang-Hua Yang (UK)
Zheng Rong Yang (UK)
Xinfeng Ye (New Zealand)
Benjamin Yen (Hong Kong, China)
Dingli Yu (UK)
Jeffrey Yu (Hong Kong, China)
Qingfu Zhang (UK)

Additional Reviewers

Ailing Chen	Fei Liu	Hyungkyu Yang
Asim Karim	Fei Yu	Hyungwoo Kang
bian Yang	Feng Liu	Jeanshyan Wang
Bin Song	Fengzhan Tian	Jian Liao
Binsheng Liu	Fuyan Liu	Jian Wang
Bo An	Fuzheng Yang	Jian Zhao
Bo Chen	Guangming Shi	Jianming Fu
Bo Cheng	Guicheng Wang	Jianping Zeng
Bo Liu	Guiguang Ding	Jianyu Xiao
Bo Zhang	Guimin Chen	Jiawei Zhang
Bobby D. Gerardo	Gumin Jeong	Jieyu Meng
Byunggil Lee	Guogang Tian	Jiguo Li
Changan Yuan	Guojiang Fu	Jing Han
Changan Kim	Guowei Yang	Jing Liu
Changji Wang	Haewon Choi	Jingmei liu
Changjie Wang	Haiguang Chen	Jinlong Wang
Changsheng Yi	Haiyan Jin	Jinqian Liang
Chanyun Yang	Hansuh Koo	Jin-Seon Lee
Chiachen Lin	Heejo Lee	Jinwei Wang
Chienchih Yu	Heejun Song	Jongwan Kim
Chinhung Liu	Hong Liu	Juan Zhang
Chong Liu	Hong Zhang	Jun Kong
Chong Wang	Hongbin Shen	Jun Ma
Chong Wu	Hongbing Liu	Jun Zhang
Chongzhao Han	Hongfang Zhou	Junbo Gao
Chundong Wang	Hongsheng Su	Juncheol Park
Chunhong Cao	Hongtao Wu	Junying Zhang
Chunxiang Gu	Hongwei Gao	K. W. Chau
Chunxiang Xu	Hongwei Huo	Kang-Soo You
Chunyan Liang	Hongxia Cai	Kanta Matsuura
Congfu Xu	Hongzhen Zheng	Ke Liao
Cuiran Li	Horong Henry	Keisuke Takemori
Daoyi Dong	Hsinhung Wu	Kihyeon Kwon
Darong Huang	Hua Xu	Kongfa Hu
Dayong Deng	Hua Zhong	Kyoungmi Lee
Dechang Pi	Huafeng Deng	Lei Sun
Deji Wang	Huanjun Liu	Li Wang
Dong Zhang	Huantong Geng	Liang Zhang
Dongmei Fu	Huaqiu Wang	Liangli Ma
Enhong Chen	Hui Wang	Liangxiao Jiang
Eunjun Yoon	Huilian Zhang	Lianying Zhou
Fan Zhang	Huiying Li	Liaojun Pang
Fang Liu	Hyun Sun Kang	Libiao Zhang

Licheng Wang	Shangmin Luan	Xiaofeng Liu
Liefeng Bo	Shaowei Wang	Xiaofeng Rong
Lijun Wu	Shaozhen Chen	Xiaohong Hao
Limin Wang	Shengfeng Tian	Xiaohui Yuan
Lin Lin	Shenghui Su	Xiaoliang He
Ling Chen	Shi Min	Xiaoyan Tao
Ling Zhang	Shifeng Rui	Xiaozhu Lin
Lingfang Zeng	Shiguo Lian	Xijian Ping
Linhuan Wang	Shuping Yao	Xinbo Gao
Liping Yan	Sinjae Kang	Xinchen Zhang
Meng Zhang	Songwook Lee	Xingzhou Zhang
Mengjie Yu	Sooyeon Shin	Xinling Shi
Ming Dong	Sunghae Jun	Xinman Zhang
Ming Li	Sungjune Hong	XiuPing Guo
Ming Li	Suresh Sundaram	Xiuqin Chu
Minxia Luo	Tangfei Tao	Xuebing Zhou
Murat Ekinci	Tao Guan	Xuelong Chen
Ni Zhang	Tao Peng	Yajun Guo
Noria Foukia	Teemupekka Virtanen	Yan Wang
Omran Mahamed	Terry House	Yanchun Liang
Pengfei Liu	Tianding Chen	Yao Wang
Ping Guo	Tianyang Dong	Yeonseung Ryu
Puru Su	Tieli Sun	Yi Li
Qi Liu	Tingzhu Huangy	Yibo Zhang
Qi Xia	W. X. Wang	Yichun Liu
Qian Chen	Wei Chen	Yifei Pu
Qiang Guo	Wei Hu	Yijia Zhang
Qiang Wang	Wei Yan	Yijuan Shi
Qiang Wei	Wei Zhang	Yijun Mo
Qiang Zhang	Weiguo Han	Yilin Lin
Qin Wang	Weijun Chen	Yiming Zhou
Qinghao Meng	Weimin Xue	Yinliang Zhao
Qinghua Hu	Weixing Wang	Yintian Liu
Qiuhua Zheng	Weizhen Yan	Yong Xu
Qizhi Zhang	Wencang Zhao	Yong Zhang
Ravi Prakash	Wenjie Li	Yongjie Wang
Ronghua Shang	Wen Quan	Yongjun Li
Rongjun Li	Wensen An	Yuan Shu
Rongqing Yi	Wenyuan Wang	Yuan Yuan
Rongxing Lu	Xia Xu	Yubao Liu
Ruo Hu	Xiangchong Liu	Yufeng Zhang
Sangho Park	Xiangchu Feng	Yufu Ning
Sangyoung Lee	Xiangyong Li	Yukun Cao
Seonghoon Lee	Xianhua Dai	Yunfeng Li
Seungwan Lee	Xiaofeng Chen	Yuqian Zhao

Zaobin Gan	Zhian Cheng	Zhiwei Ni
Zhansheng Liu	Zhicheng Chen	Zhiwu Liao
Zhaofeng Ma	Zhigang Ma	Zhong Liu
Zhenchuan Chai	Zhihong Deng	Ziyi Chen
Zhengtao Jiang	Zhijia Cai	Zongben Xu
Zhenlong Du	Zhiqing Meng	
Zhi Liu	Zhisong Pan	

Sponsoring Institutions

IEEE (Hong Kong) Computational Intelligence Chapter
Xidian University
Hong Kong Baptist University
National Natural Science Foundation of China
Guangdong University of Technology

Table of Contents – Part I

Learning and Fuzzy Systems

Empirical Analysis of Database Privacy Using Twofold Integrals <i>Jordi Nin, Vicenç Torra</i>	1
Intrusion Detection Alert Verification Based on Multi-level Fuzzy Comprehensive Evaluation <i>Chengpo Mu, Houkuan Huang, Shengfeng Tian</i>	9
Improving the Scalability of Automatic Programming <i>Henrik Berg, Roland Olsson</i>	17
Texture Segmentation by Unsupervised Learning and Histogram Analysis Using Boundary Tracing <i>Woobeom Lee, Wookhyun Kim</i>	25
An Improved Bayesian Network Learning Algorithm Based on Dependency Analysis <i>Fengzhan Tian, Shengfeng Tian, Jian Yu, Houkuan Huang</i>	33
Mining Common Patterns on Graphs <i>Ivan Olmos, Jesus A. Gonzalez, Mauricio Osorio</i>	41
Moderated Innovations in Self-poised Ensemble Learning <i>Ricardo Nanculef, Carlos Valle, Héctor Allende, Claudio Moraga</i>	49
An Adaptive Framework for Solving Multiple Hard Problems Under Time Constraints <i>Sandip Aine, Rajeev Kumar, P.P. Chakrabarti</i>	57
An RLS-Based Natural Actor-Critic Algorithm for Locomotion of a Two-Linked Robot Arm <i>Jooyoung Park, Jongho Kim, Daesung Kang</i>	65
Dynamic Clustering Using Multi-objective Evolutionary Algorithm <i>Enhong Chen, Feng Wang</i>	73
Multimodal FeedForward Self-organizing Maps <i>Andrew P. Papliński, Lennart Gustafsson</i>	81

Decision Fusion Based Unsupervised Texture Image Segmentation <i>Hua Zhong, Licheng Jiao</i>	89
Speaker Adaptation Techniques for Speech Recognition with a Speaker-Independent Phonetic Recognizer <i>Weon-Goo Kim, MinSeok Jang</i>	95
Fuzzy QoS Controllers in Diff-Serv Scheduler Using Genetic Algorithms <i>Baolin Sun, Qiu Yang, Jun Ma, Hua Chen</i>	101
Neural Network Based Algorithms for Diagnosis and Classification of Breast Cancer Tumor <i>In-Sung Jung, Devinder Thapa, Gi-Nam Wang</i>	107
New Learning Algorithm for Hierarchical Structure Learning Automata Operating in P-Model Stationary Random Environment <i>Yoshio Mogami</i>	115
A TFN-Based AHP Model for Solving Group Decision-Making Problems <i>Jian Cao, Gengui Zhou, Feng Ye</i>	121
A Tactics for Robot Soccer with Fuzzy Logic Mediator <i>Jeongjun Lee, Dongmin Ji, Wonchang Lee, Geuntaek Kang, Moon G. Joo</i>	127
Gait Control for Biped Robot Using Fuzzy Wavelet Neural Network <i>Pengfei Liu, Jiuqiang Han</i>	133
A New Approach for Regression: Visual Regression Approach <i>Deyu Meng, Chen Xu, Wenfeng Jing</i>	139
Orthogonally Rotational Transformation for Naive Bayes Learning <i>Limin Wang, Chunhong Cao, Haijun Li, Haixia Chen, Liyang Dong</i>	145
Efficient Learning Bayesian Networks Using PSO <i>Tao Du, S.S. Zhang, Zongjiang Wang</i>	151
Improving K-Modes Algorithm Considering Frequencies of Attribute Values in Mode <i>Zengyou He, Shengchun Deng, Xiaofei Xu</i>	157
Distance Protection of Compensated Transmission Line Using Computational Intelligence <i>S.R. Samantaray, P.K. Dash, G. Panda, B.K. Panigrahi</i>	163

Computational Intelligence for Network Intrusion Detection: Recent Contributions <i>Asim Karim</i>	170
---	-----

Evolutionary Computation

Design of a Switching PID Controller Using Advanced Genetic Algorithm for a Nonlinear System <i>Jung-Shik Kong, Bo-Hee Lee, Jin-Geol Kim</i>	176
Preference Bi-objective Evolutionary Algorithm for Constrained Optimization <i>Yuping Wang, Dalian Liu, Yiu-Ming Cheung</i>	184
Self-adaptive Differential Evolution <i>Mahamed G.H. Omran, Ayed Salman, Andries P. Engelbrecht</i>	192
Steady-State Evolutionary Algorithm for Multimodal Function Global Optimization <i>Ziyi Chen, Lishan Kang</i>	200
Finding Optimal Addition Chains Using a Genetic Algorithm Approach <i>Nareli Cruz-Cortés, Francisco Rodríguez-Henríquez, Raúl Juárez-Morales, Carlos A. Coello Coello</i>	208
Using Reconfigurable Architecture-Based Intrinsic Incremental Evolution to Evolve a Character Classification System <i>Jin Wang, Je Kyo Jung, Yong-min Lee, Chong Ho Lee</i>	216
On the Relevance of Using Gene Expression Programming in Destination-Based Traffic Engineering <i>Antoine B. Bagula, Hong F. Wang</i>	224
Model and Convergence for the Combination of Genetic Algorithm and Ant Algorithm <i>Jianli Ding, Wansheng Tang, Yufu Ning</i>	230
Moving Block Sequence and Organizational Evolutionary Algorithm for General Floorplanning <i>Jing Liu, Weicai Zhong, Licheng Jiao</i>	238
Integrating the Simplified Interpolation into the Genetic Algorithm for Constrained Optimization Problems <i>Hong Li, Yong-Chang Jiao, Yuping Wang</i>	247

Using Ensemble Method to Improve the Performance of Genetic Algorithm <i>Shude Zhou, Zengqi Sun</i>	255
Parallel Mining for Classification Rules with Ant Colony Algorithm <i>Ling Chen, Li Tu</i>	261
A Genetic Algorithm Approach on Reverse Logistics Optimization for Product Return Distribution Network <i>Gengui Zhou, Zhenyu Cao, Jian Cao, Zhiqing Meng</i>	267
Multi-objective Evolutionary Design and Knowledge Discovery of Logic Circuits with an Improved Genetic Algorithm <i>Shuguang Zhao, Licheng Jiao, Jun Zhao</i>	273
Robust Mobile Robot Localization Using an Evolutionary Particle Filter <i>Bo Yin, Zhiqiang Wei, Xiaodong Zhuang</i>	279
Hybrid Genetic Algorithm for Solving the Degree-Constrained Minimal Bandwidth Multicast Routing Problem <i>Yun Pan, Zhenwei Yu, Licheng Wang</i>	285
Using Fuzzy Possibilistic Mean and Variance in Portfolio Selection Model <i>Weiguo Zhang, Yingluo Wang</i>	291
A Novel Genetic Algorithm for Multi-criteria Minimum Spanning Tree Problem <i>Lixia Han, Yuping Wang</i>	297
 Intelligent Agents and Systems	
A Software Architecture for Multi-agent Systems <i>Vasu S. Alagar, Mao Zheng</i>	303
User-Oriented Multimedia Service Using Smart Sensor Agent Module in the Intelligent Home <i>Jong-Hyuk Park, Jun Choi, Sang-Jin Lee, Hye-Ung Park, Deok-Gyu Lee</i>	313
Meta-learning Experiences with the MINDFUL System <i>Ciro Castiello, Anna Maria Fanelli</i>	321
Learning Cooperation from Classifier Systems <i>Trung Hau Tran, Cédric Sanza, Yves Duthen</i>	329

Location Management Using Hierarchical Structured Agents for Distributed Databases <i>Romeo Mark A. Mateo, Bobby D. Gerardo, Jaewan Lee</i>	337
On_line Measurement System of Virtual Dielectric Loss Based on Wavelets and LabVIEW and Correlation Technics <i>BaoBao Wang, Ye Wang</i>	343
Model Checking Temporal Logics of Knowledge and Its Application in Security Verification <i>Lijun Wu, Kaile Su, Qingliang Chen</i>	349
A Computational Approach for Belief Change <i>Shangmin Luan, Guozhong Dai</i>	355
Feature Selection by Fuzzy Inference and Its Application to Spam-Mail Filtering <i>Jong-Wan Kim, Sin-Jae Kang</i>	361
Design of Multiagent Framework for Cellular Networks <i>A.K. Sharma, Dimple Juneja</i>	367
Transitive Dependence Based Formation of Virtual Organizations <i>Bo An, Chunyan Miao, Zhiqi Shen, Yuan Miao, Daijie Cheng</i>	375
An Agent Based Education Resource Purvey System <i>Xiaochun Cheng, Xin He, Xiaoqi Ma, Dongdai Zhou, Peijun Duan, Shaochun Zhong</i>	381
Model of Game Agent for Auction-Based Negotiation <i>Jun Hu, Chun Guan</i>	387
An Autonomous Mobile Robot Based on Quantum Algorithm <i>Daoyi Dong, Chunlin Chen, Chenbin Zhang, Zonghai Chen</i>	393
A MPC and Genetic Algorithm Based Approach for Multiple UAVs Cooperative Search <i>Jing Tian, Yanxing Zheng, Huayong Zhu, Lincheng Shen</i>	399
Self-organization Evolution of Supply Networks: System Modeling and Simulation Based on Multi-agent <i>Gang Li, Linyan Sun, Ping Ji, Haiquan Li</i>	405
Modeling and Analysis of Multi-agent Systems Based on π -Calculus <i>Fenglei Liu, Zhenhua Yu, Yuanli Cai</i>	410

A Cooperation Mechanism in Agent-Based Autonomic Storage Systems <i>Jingli Zhou, Gang Liu, Shengsheng Yu, Yong Su</i>	416
A Mobile Agent Based Spam Filter System <i>Xiaochun Cheng, Xiaoqi Ma, Long Wang, Shaochun Zhong</i>	422
Hexagon-Based Q-Learning to Find a Hidden Target Object <i>Han-Ul Yoon, Kwee-Bo Sim</i>	428

Intelligent Information Retrieval

A Naive Statistics Method for Electronic Program Guide Recommendation System <i>Jin An Xu, Kenji Araki</i>	434
A Hybrid Text Classification System Using Sentential Frequent Itemsets <i>Shizhu Liu, Heping Hu</i>	442
An Approach of Information Extraction from Web Documents for Automatic Ontology Generation <i>Ki-Won Yeom, Ji-Hyung Park</i>	450
Improving Text Categorization Using the Importance of Words in Different Categories <i>Zhihong Deng, Ming Zhang</i>	458
Image Copy Detection with Rotating Tolerance <i>Mingni Wu, Chiachen Lin, Chinchun Chang</i>	464
Interactive and Adaptive Search Context for the User with the Exploration of Personalized View Reformulation <i>Supratip Ghose, Geun-Sik Jo</i>	470
Integrating Collaborate and Content-Based Filtering for Personalized Information Recommendation <i>Zhiyun Xin, Jizhong Zhao, Ming Gu, Jiaguang Sun</i>	476
Interest Region-Based Image Retrieval System Based on Graph-Cut Segmentation and Feature Vectors <i>Dongfeng Han, Wenhui Li, Xiaomo Wang, Yanjie She</i>	483
A Method for Automating the Extraction of Specialized Information from the Web <i>Ling Lin, Antonio Liotta, Andrew Hippisley</i>	489

An Incremental Updating Method for Clustering-Based High-Dimensional Data Indexing <i>Ben Wang, John Q. Gan</i>	495
--	-----

Support Vector Machine

Typhoon Track Prediction by a Support Vector Machine Using Data Reduction Methods <i>Hee-Jun Song, Sung-Hoe Huh, Joo-Hong Kim, Chang-Hoi Ho, Seon-Ki Park</i>	503
Forecasting Tourism Demand Using a Multifactor Support Vector Machine Model <i>Ping-Feng Pai, Wei-Chiang Hong, Chih-Sheng Lin</i>	512
A Study of Modelling Non-stationary Time Series Using Support Vector Machines with Fuzzy Segmentation Information <i>Shaomin Zhang, Lijia Zhi, Shukuan Lin</i>	520
Support Vector Machine Based Trajectory Metamodel for Conceptual Design of Multi-stage Space Launch Vehicle <i>Saqlain Akhtar, He Linshu</i>	528
Transductive Support Vector Machines Using Simulated Annealing <i>Fan Sun, Maosong Sun</i>	536
Input Selection for Support Vector Machines Using Genetic Algorithms <i>Hee-Jun Song, Seon-Gu Lee, Sung-Hoe Huh</i>	544
Associating k NN and SVM for Higher Classification Accuracy <i>Che-Chang Hsu, Chan-Yun Yang, Jr-Syu Yang</i>	550
Multi-class SVMs Based on SOM Decoding Algorithm and Its Application in Pattern Recognition <i>Xiaoyan Tao, Hongbing Ji</i>	556
Selective Dissemination of XML Documents Using GAs and SVM <i>K.G. Srinivasa, S. Sharath, K.R. Venugopal, Lalit M. Patnaik</i>	562
A Smoothing Support Vector Machine Based on Exact Penalty Function <i>Zhiqing Meng, Gengui Zhou, Yihua Zhu, Lifang Peng</i>	568
Speech Acts Tagging System for Korean Using Support Vector Machines <i>Songwook Lee, Jongmin Eun, Jungyun Seo</i>	574

A New Support Vector Machine for Multi-class Classification
Zhiquan Qi, Yingjie Tian, Naiyang Deng 580

Support Vector Classification with Nominal Attributes
Yingjie Tian, Naiyang Deng 586

A New Smooth Support Vector Machine
Yubo Yuan, Chunzhong Li 592

The Application of Support Vector Machine in the Potentiality
 Evaluation for Revegetation of Abandoned Lands from Coal Mining
 Activities
Chuanli Zhuang, Zetian Fu, Ping Yang, Xiaoshuan Zhang..... 598

Prediction of T-cell Epitopes Using Support Vector Machine and
 Similarity Kernel
Feng Shi, Jing Huang 604

Radial Basis Function Support Vector Machine Based Soft-Magnetic
 Ring Core Inspection
Liangjiang Liu, Yaonan Wang..... 609

Direct Adaptive NN Control of a Class of Feedforward Systems
Wang Dongliang, Liu Bin, Zhang Zengke 616

Swarm Intelligence

Performance of an Ant Colony Optimization (ACO) Algorithm on the
 Dynamic Load-Balanced Clustering Problem in Ad Hoc Networks
Chin Kuan Ho, Hong Tat Ewe..... 622

Hybrid Particle Swarm Optimization for Flow Shop Scheduling with
 Stochastic Processing Time
Bo Liu, Ling Wang, Yi-hui Jin 630

Particle Swarm Optimizer with C-Pg Mutation
Guojiang Fu, Shaomei Wang, Mingjun Chen, Ning Li 638

Algal Bloom Prediction with Particle Swarm Optimization Algorithm
K. W. Chau 645

Synthesis of the Antenna Array Using a Modified Particle Swarm
 Optimization Algorithm
Tengbo Chen, Yong-Chang Jiao, Fushun Zhang 651

An Ant Colony Optimization Approach to the Degree-Constrained Minimum Spanning Tree Problem <i>Y.T. Bau, C.K. Ho, H.T. Ewe</i>	657
Crowd Avoidance Strategy in Particle Swarm Algorithm <i>Guimin Chen, Qi Han, Jianyuan Jia, Wenchao Song</i>	663
Particle Swarm Optimization with Multiscale Searching Method <i>Xiaohui Yuan, Jing Peng, Yasumasa Nishiura</i>	669
Outcome-Space Branch and Bound Algorithm for Solving Linear Multiplicative Programming <i>Yuelin Gao, Chengxian Xu, Yueting Yang</i>	675
A Binary Ant Colony Optimization for the Unconstrained Function Optimization Problem <i>Min Kong, Peng Tian</i>	682
Data Mining	
Mining Dynamic Association Rules in Databases <i>Jinfeng Liu, Gang Rong</i>	688
A Novel Typical-Sample-Weighted Clustering Algorithm for Large Data Sets <i>Jie Li, Xinbo Gao, Licheng Jiao</i>	696
Mining Weighted Generalized Fuzzy Association Rules with Fuzzy Taxonomies <i>Shen Bin, Yao Min, Yuan Bo</i>	704
Concept Chain Based Text Clustering <i>Shaoyu Song, Jian Zhang, Chunping Li</i>	713
An Efficient Range Query Under the Time Warping Distance <i>Chuyu Li, Long Jin, Sungbo Seo, Keun Ho Ryu</i>	721
Robust Scene Boundary Detection Based on Audiovisual Information <i>Soon-tak Lee, Joon-sik Baek, Joong-hwan Baek</i>	729
An FP-Tree Based Approach for Mining All Strongly Correlated Item Pairs <i>Zengyou He, Shengchun Deng, Xiaofei Xu</i>	735

An Improved kNN Algorithm – Fuzzy kNN
Wenqian Shang, Houkuan Huang, Haibin Zhu, Yongmin Lin, Zhihai Wang, Youli Qu 741

A New Integrated Personalized Recommendation Algorithm
Hongfang Zhou, Boqin Feng, Lintao Lv, Zhurong Wang 747

An Improved EMASK Algorithm for Privacy-Preserving Frequent Pattern Mining
Congfu Xu, Jinlong Wang, Hongwei Dan, Yunhe Pan 752

CR*-Tree: An Improved R-Tree Using Cost Model
Haibo Chen, Zhanquan Wang 758

Grid-ODF: Detecting Outliers Effectively and Efficiently in Large Multi-dimensional Databases
Wei Wang, Ji Zhang, Hai Wang 765

Clustering XML Documents by Structure Based on Common Neighbor
Xizhe Zhang, Tianyang Lv, Zhengxuan Wang, Wanli Zuo 771

A Generalized Global Convergence Theory of Projection-Type Neural Networks for Optimization
Rui Zhang, Zongben Xu 777

Hierarchical Recognition of English Calling Card by Using Multiresolution Images and Enhanced Neural Network
Kwang-Baek Kim, Sungshin Kim 785

An Novel Artificial Immune Systems Multi-objective Optimization Algorithm for 0/1 Knapsack Problems
Wenping Ma, Licheng Jiao, Maoguo Gong, Fang Liu 793

RD-Based Seeded Region Growing for Extraction of Breast Tumor in an Ultrasound Volume
Jong In Kwak, Sang Hyun Kim, Nam Chul Kim 799

Improving Classification for Microarray Data Sets by Constructing Synthetic Data
Shun Bian, Wenjia Wang 809

A Method to Locate the Position of Mobile Robot Using Extended Kalman Filter
Ping Wei, Chengxian Xu, Fengji Zhao 815

Simulated Annealing with Injecting Star-Alignment for Multiple Sequence Alignments <i>Hongwei Huo, Hua Ming</i>	821
A Noise-Insensitive Hierarchical Min-Max Octree for Visualization of Ultrasound Datasets <i>Sukhyun Lim, Kang-hee Seo, Byeong-Seok Shin</i>	827
A Novel Fusing Algorithm for Retinal Fundus Images <i>Bin Fang, Xinge You, Yuan Yan Tang</i>	833
Improving PSO-Based Multiobjective Optimization Using Competition and Immunity Clonal <i>Xiaohua Zhang, Hongyun Meng, Licheng Jiao</i>	839
Clonal Selection Algorithm for Dynamic Multiobjective Optimization <i>Ronghua Shang, Licheng Jiao, Maoguo Gong, Bin Lu</i>	846
Key Frame Extraction Based on Evolutionary Artificial Immune Network <i>Fang Liu, Xiaoying Pan</i>	852
Clonal Selection Algorithm with Immunologic Regulation for Function Optimization <i>Hang Yu, Maoguo Gong, Licheng Jiao, Bin Zhang</i>	858
A Fault-Tolerant and Minimum-Energy Path-Preserving Topology Control Algorithm for Wireless Multi-hop Networks <i>Zhong Shen, Yilin Chang, Can Cui, Xin Zhang</i>	864
Computational Biomechanics and Experimental Verification of Vascular Stent <i>Yuxuan Wang, Hong Yi, Zhonghua Ni</i>	870
Numerical Computing of Brain Electric Field in Electroencephalogram <i>Dexin Zhao, Zhiyong Feng, Wenjie Li, Shugang Tang</i>	878
A Novel Multi-stage 3D Medical Image Segmentation: Methodology and Validation <i>Jianfeng Xu, Lixu Gu, Xiahai Zhuang, Terry Peters</i>	884
Medical Image Alignment by Normal Vector Information <i>Xiahai Zhuang, Lixu Gu, Jianfeng Xu</i>	890
Global Exponential Stability of Non-autonomous Delayed Neural Networks <i>Qiang Zhang, Dongsheng Zhou, Xiaopeng Wei</i>	896

A Prediction Method for Time Series Based on Wavelet Neural Networks <i>Xiaobing Gan, Ying Liu, Francis R. Austin</i>	902
Training Multi-layer Perceptrons Using MiniMin Approach <i>Liefeng Bo, Ling Wang, Licheng Jiao</i>	909
Two Adaptive Matching Learning Algorithms for Independent Component Analysis <i>Jinwen Ma, Fei Ge, Dengpan Gao</i>	915
Bioprocess Modeling Using Genetic Programming Based on a Double Penalty Strategy <i>Yanling Wu, Jiangang Lu, Youxian Sun, Peifei Yu</i>	921
An Improved Gibbs Sampling Algorithm for Finding TFBS <i>Caisheng He, Xianhua Dai</i>	927
 Pattern Recognition	
A Novel Fisher Criterion Based S_t -Subspace Linear Discriminant Method for Face Recognition <i>Wensheng Chen, Pong C. Yuen, Jian Huang, Jianhuang Lai</i>	933
EmoEars: An Emotion Recognition System for Mandarin Speech <i>Bo Xie, Ling Chen, Gen-Cai Chen, Chun Chen</i>	941
User Identification Using User's Walking Pattern over the ubiFloorII <i>Jaeseok Yun, Woontack Woo, Jeha Ryu</i>	949
Evolving RBF Neural Networks for Pattern Classification <i>Zheng Qin, Junying Chen, Yu Liu, Jiang Lu</i>	957
Discrimination of Patchouli of Different Geographical Origins with Two-Dimensional IR Correlation Spectroscopy and Wavelet Transform <i>Daqi Zhan, Suqin Sun, Yiu-ming Cheung</i>	965
Gait Recognition Using View Distance Vectors <i>Murat Ekinici, Eyup Gedikli</i>	973
HMM Parameter Adaptation Using the Truncated First-Order VTS and EM Algorithm for Robust Speech Recognition <i>Haifeng Shen, Qunxia Li, Jun Guo, Gang Liu</i>	979

Model Type Recognition Using De-interlacing and Block Code Generation <i>Cheol-Ki Kim, Sang-Gul Lee, Kwang-Baek Kim</i>	985
R-functions Based Classification for Abnormal Software Process Detection <i>Anton Bougaev, Aleksey Urmanov</i>	991
A Unified Framework for Shot Boundary Detection <i>Bing Han, Xinbo Gao, Hongbing Ji</i>	997
Image Recognition with LPP Mixtures <i>SiBao Chen, Min Kong, Bin Luo</i>	1003
Line-Based Camera Calibration <i>Xiuqin Chu, Fangming Hu, Yushan Li</i>	1009
Shot Boundary Detection Based on SVM and TMRA <i>Wei Fang, Sen Liu, Huamin Feng, Yong Fang</i>	1015
Robust Pattern Recognition Scheme for Devanagari Script <i>Amit Dhurandhar, Kartik Shankarnarayanan, Rakesh Jawale</i>	1021
Credit Evaluation Model and Applications Based on Probabilistic Neural Network <i>Sulin Pang</i>	1027
Fingerprint Ridge Line Extraction Based on Tracing and Directional Feedback <i>Rui Ma, Yaxuan Qi, Changshui Zhang, Jiaxin Wang</i>	1033
A New Method for Human Gait Recognition Using Temporal Analysis <i>Han Su, Fenggang Huang</i>	1039
Microcalcification Patterns Recognition Based Combination of Autoassociator and Classifier <i>Wencang Zhao, Xinbo Yu, Fengxiang Li</i>	1045
Improved Method for Gradient-Threshold Edge Detector Based on HVS <i>Fuzheng Yang, Shuai Wan, Yilin Chang</i>	1051
MUSC: Multigrid Shape Codes and Their Applications to Image Retrieval <i>Arindam Biswas, Partha Bhowmick, Bhargab B. Bhattacharya</i>	1057

Applications

Adaptation of Intelligent Characters to Changes of Game Environments <i>Byeong Heon Cho, Sung Hoon Jung, Kwang-Hyun Shim, Yeong Rak Seong, Ha Ryoung Oh</i>	1064
An Knowledge Model for Self-regenerative Service Activations Adaptation Across Standards <i>Mengjie Yu, David Llewellyn Jones, A. Taleb-Bendiab</i>	1074
An Agent for the HCARD Model in the Distributed Environment <i>Bobby D. Gerardo, Jae-Wan Lee, Jae-jeong Hwang, Jung-Eun Kim</i>	1082
A New Class of Filled Functions for Global Minimization <i>Xiaoliang He, Chengxian Xu, Chuanchao Zhu</i>	1088
Modified PSO Algorithm for Deadlock Control in FMS <i>Hesuan Hu, Zhiwu Li, Weidong Wang</i>	1094
Optimization Design of Controller Periods Using Evolution Strategy <i>Hong Jin, Hui Wang, Hongan Wang, Guozhong Dai</i>	1100
Application of Multi-objective Evolutionary Algorithm in Coordinated Design of PSS and SVC Controllers <i>Zhenyu Zou, Quanyuan Jiang, Pengxiang Zhang, Yijia Cao</i>	1106
Author Index	1113

Table of Contents – Part II

Cryptography and Coding

A Fast Inversion Algorithm and Low-Complexity Architecture over $GF(2^m)$ <i>Sosun Kim, Nam Su Chang, Chang Han Kim, Young-Ho Park, Jongin Lim</i>	1
An ID-Based Optimistic Fair Signature Exchange Protocol from Pairings <i>Chunxiang Gu, Yuefei Zhu, Yajuan Zhang</i>	9
FMS Attack-Resistant WEP Implementation Is Still Broken <i>Toshihiro Ohigashi, Yoshiaki Shiraiishi, Masakatu Morii</i>	17
Design of a New Kind of Encryption Kernel Based on RSA Algorithm <i>Ping Dong, Xiangdong Shi, Jiehui Yang</i>	27
On the Security of Condorcet Electronic Voting Scheme <i>Yoon Cheol Lee, Hiroshi Doi</i>	33
Special Distribution of the Shortest Linear Recurring Sequences in $Z/(p)$ Field <i>Qian Yin, Yunlun Luo, Ping Guo</i>	43
Cryptanalysis of a Cellular Automata Cryptosystem <i>Jingmei Liu, Xiangguo Cheng, Xinmei Wang</i>	49
A New Conceptual Framework Within Information Privacy: Meta Privacy <i>Geoff Skinner, Song Han, Elizabeth Chang</i>	55
Error Oracle Attacks on Several Modes of Operation <i>Fengtong Wen, Wenling Wu, Qiaoyan Wen</i>	62
Stability of the Linear Complexity of the Generalized Self-shrinking Sequences <i>Lihua Dong, Yong Zeng, Yupu Hu</i>	68
On the Construction of Some Optimal Polynomial Codes <i>Yajing Li, Weihong Chen</i>	74

Perceptual Hashing of Video Content Based on Differential Block Similarity
Xuebing Zhou, Martin Schmucker, Christopher L. Brown 80

Cryptographic Protocols

Secure Software Smartcard Resilient to Capture
Seung Wook Jung, Christoph Ruland 86

Revised Fischlin’s (Blind) Signature Schemes
Kewei Lv 96

Certificateless Threshold Signature Schemes
Licheng Wang, Zhenfu Cao, Xiangxue Li, Haifeng Qian 104

An Efficient Certificateless Signature Scheme
M. Choudary Gorantla, Ashutosh Saxena 110

ID-Based Restrictive Partially Blind Signatures
Xiaofeng Chen, Fangguo Zhang, Shengli Liu 117

Batch Verification with DSA-Type Digital Signatures for Ubiquitous Computing
Seungwon Lee, Seongje Cho, Jongmoo Choi, Yookun Cho 125

On Anonymity of Group Signatures
Sujing Zhou, Dongdai Lin 131

The Running-Mode Analysis of Two-Party Optimistic Fair Exchange Protocols
Yuqing Zhang, Zhiling Wang, Bo Yang 137

Password-Based Group Key Exchange Secure Against Insider Guessing Attacks
Jin Wook Byun, Dong Hoon Lee, Jongin Lim 143

On the Security of Some Password-Based Key Agreement Schemes
Qiang Tang, Chris J. Mitchell 149

A New Group Rekeying Method in Secure Multicast
Yong Xu, Yuxiang Sun 155

Pairing-Based Provable Blind Signature Scheme Without Random Oracles
Jian Liao, Yinghao Qi, Peiwei Huang, Mentian Rong 161

Efficient ID-Based Proxy Signature and Proxy Signcryption Form Bilinear Pairings <i>Qin Wang, Zhenfu Cao</i>	167
An Identity-Based Threshold Signcryption Scheme with Semantic Security <i>Changgen Peng, Xiang Li</i>	173
A Token-Based Single Sign-On Protocol <i>Li Hui, Shen Ting</i>	180
Simple Threshold RSA Signature Scheme Based on Simple Secret Sharing <i>Shaohua Tang</i>	186
Efficient Compilers for Authenticated Group Key Exchange <i>Qiang Tang, Chris J. Mitchell</i>	192
Insider Impersonation-MIM Attack to Tripartite Key Agreement Scheme and an Efficient Protocol for Multiple Keys <i>Lihua Wang, Takeshi Okamoto, Tsuyoshi Takagi, Eiji Okamoto</i>	198
Intrusion Detection	
An Immune System Inspired Approach of Collaborative Intrusion Detection System Using Mobile Agents in Wireless Ad Hoc Networks <i>Ki-Won Yeom, Ji-Hyung Park</i>	204
A New User-Habit Based Approach for Early Warning of Worms <i>Ping Wang, Binxing Fang, Xiaochun Yun</i>	212
A Multi-gigabit Virus Detection Algorithm Using Ternary CAM <i>Il-Seop Song, Youngseok Lee, Taek-Geun Kwon</i>	220
Sampling Distance Analysis of Gigantic Data Mining for Intrusion Detection Systems <i>Yong Zeng, Jianfeng Ma</i>	228
Hardware-Software Hybrid Packet Processing for Intrusion Detection Systems <i>Saraswathi Sachidananda, Srividya Gopalan, Sridhar Varadarajan</i> ...	236
D-S Evidence Theory and Its Data Fusion Application in Intrusion Detection <i>Junfeng Tian, Weidong Zhao, Ruizhong Du</i>	244

A New Network Anomaly Detection Technique Based on Per-Flow and Per-Service Statistics <i>Yuji Waizumi, Daisuke Kudo, Nei Kato, Yoshiaki Nemoto</i>	252
<i>SoIDPS: Sensor Objects-Based Intrusion Detection and Prevention System and Its Implementation</i> <i>SeongJe Cho, Hye-Young Chang, HongGeun Kim, WoongChul Choi</i>	260
A Statistical Model for Detecting Abnormality in Static-Priority Scheduling Networks with Differentiated Services <i>Ming Li, Wei Zhao</i>	267
Tamper Detection for Ubiquitous RFID-Enabled Supply Chain <i>Vidyasagar Potdar, Chen Wu, Elizabeth Chang</i>	273
Measuring the Histogram Feature Vector for Anomaly Network Traffic <i>Wei Yan</i>	279
Efficient Small Face Detection in Surveillance Images Using Major Color Component and LDA Scheme <i>Kyunghwan Baek, Heejun Jang, Youngjun Han, Hernsoo Hahn</i>	285
Fast Motion Detection Based on Accumulative Optical Flow and Double Background Model <i>Jin Zheng, Bo Li, Bing Zhou, Wei Li</i>	291
Reducing Worm Detection Time and False Alarm in Virus Throttling <i>Jangbok Kim, Jaehong Shim, Gihyun Jung, Kyunghee Choi</i>	297
Protection Against Format String Attacks by Binary Rewriting <i>Jin Ho You, Seong Chae Seo, Young Dae Kim, Jun Yong Choi, Sang Jun Lee, Byung Ki Kim</i>	303
Masquerade Detection System Based on Principal Component Analysis and Radial Basics Function <i>Zhanchun Li, Zhitang Li, Yao Li, Bin Liu</i>	309
Anomaly Detection Method Based on HMMs Using System Call and Call Stack Information <i>Cheng Zhang, Qinke Peng</i>	315
Parallel Optimization Technology for Backbone Network Intrusion Detection System <i>Xiaojuan Sun, Xinliang Zhou, Ninghui Sun, Mingyu Chen</i>	322

Attack Scenario Construction Based on Rule and Fuzzy Clustering <i>Linru Ma, Lin Yang, Jianxin Wang</i>	328
A CBR Engine Adapting to IDS <i>Lingjuan Li, Wenyu Tang, Ruchuan Wang</i>	334
Application of Fuzzy Logic for Distributed Intrusion Detection <i>Hee Suk Seo, Tae Ho Cho</i>	340
Security Models and Architecture	
Dynamic Access Control for Pervasive Grid Applications <i>Syed Naqvi, Michel Riguidel</i>	348
On the Security of the Canetti-Krawczyk Model <i>Xinghua Li, Jianfeng Ma, SangJae Moon</i>	356
A Novel Architecture for Detecting and Defending Against Flooding-Based DDoS Attacks <i>Yi Shi, Xinyu Yang</i>	364
A Variant of Poly1305 MAC and Its Security Proof <i>Dayin Wang, Dongdai Lin, Wenling Wu</i>	375
Covert Channel Identification Founded on Information Flow Analysis <i>Jianjun Shen, Sihan Qing, Qingni Shen, Liping Li</i>	381
Real-Time Risk Assessment with Network Sensors and Intrusion Detection Systems <i>André Årnes, Karin Sallhammar, Kjetil Haslum, Tønnes Brekne, Marie Elisabeth Gaup Moe, Svein Johan Knapskog</i>	388
Design and Implementation of a Parallel Crypto Server <i>Xiaofeng Rong, Xiaojuan Gao, Ruidan Su, Lihua Zhou</i>	398
Survivability Computation of Networked Information Systems <i>Xuegang Lin, Rongsheng Xu, Miaoliang Zhu</i>	407
Assessment of Windows System Security Using Vulnerability Relationship Graph <i>Yongzheng Zhang, Binxing Fang, Yue Chi, Xiaochun Yun</i>	415
A New (t, n) -Threshold Multi-secret Sharing Scheme <i>HuiXian Li, ChunTian Cheng, LiaoJun Pang</i>	421

An Efficient Message Broadcast Authentication Scheme for Sensor Networks <i>Sang-ho Park, Taekyoung Kwon</i>	427
Digital Image Authentication Based on Error-Correction Codes <i>Fan Zhang, Xinhong Zhang, Zhiguo Chen</i>	433
Design and Implementation of Efficient Cipher Engine for IEEE 802.11i Compatible with IEEE 802.11n and IEEE 802.11e <i>Duhyun Bae, Gwanyeon Kim, Jiho Kim, Sehyun Park, Ohyoung Song</i>	439
Secure Delegation-by-Warrant ID-Based Proxy Signcryption Scheme <i>Shanshan Duan, Zhenfu Cao, Yuan Zhou</i>	445
Building Security Requirements Using State Transition Diagram at Security Threat Location <i>Seong Chae Seo, Jin Ho You, Young Dae Kim, Jun Yong Choi, Sang Jun Lee, Byung Ki Kim</i>	451
Study on Security iSCSI Based on SSH <i>Weiping Liu, Wandong Cai</i>	457
A Scheduling Algorithm Based on a Trust Mechanism in Grid <i>Kenli Li, Yan He, Renfa Li, Tao Yang</i>	463
Enhanced Security and Privacy Mechanism of RFID Service for Pervasive Mobile Device <i>Byungil Lee, Howon Kim</i>	469
Worm Propagation Modeling and Analysis on Network <i>Yunkai Zhang, Fangwei Wang, Changguang Wang, Jianfeng Ma</i>	476
An Extensible AAA Infrastructure for IPv6 <i>Hong Zhang, Haixin Duan, Wu Liu, Jianping Wu</i>	482
The Security Proof of a 4-Way Handshake Protocol in IEEE 802.11i <i>Fan Zhang, Jianfeng Ma, SangJae Moon</i>	488
A Noble Key Pre-distribution Scheme with LU Matrix for Secure Wireless Sensor Networks <i>Chang Won Park, Sung Jin Choi, Hee Yong Youn</i>	494

Security Management

A Virtual Bridge Certificate Authority Model <i>Haibo Tian, Xi Sun, Yumin Wang</i>	500
Weak Signals in Information Security Management <i>Jorma Kajava, Reijo Savola, Rauno Varonen</i>	508
PDTM: A Policy-Driven Trust Management Framework in Distributed Systems <i>Wu Liu, Haixin Duan, Jianping Wu, Xing Li</i>	518
Methodology of Quantitative Risk Assessment for Information System Security <i>Mengquan Lin, Qiangmin Wang, Jianhua Li</i>	526
A Secure and Efficient (t, n) Threshold Verifiable Multi-secret Sharing Scheme <i>Mei-juan Huang, Jian-zhong Zhang, Shu-cui Xie</i>	532
Improvement on an Optimized Protocol for Mobile Network Authentication and Security <i>ChinChen Chang, JungSan Lee</i>	538
Neural Network Based Flow Forecast and Diagnosis <i>Qianmu Li, Manwu Xu, Hong Zhang, Fengyu Liu</i>	542
Protecting Personal Data with Various Granularities: A Logic-Based Access Control Approach <i>Bat-Odon Purevji, Masayoshi Aritsugi, Sayaka Imai, Yoshinari Kanamori, Cherri M. Pancake</i>	548
Enhancement of an Authenticated Multiple-Key Agreement Protocol Without Using Conventional One-Way Function <i>Huifeng Huang, Chinchun Chang</i>	554
Topology-Based Macroscopical Response and Control Technology for Network Security Event <i>Hui He, Mingzeng Hu, Weizhe Zhang, Hongli Zhang, Zhi Yang</i>	560

Watermarking and Information Hiding

Adaptive Hiding Scheme Based on VQ-Indices Using Commutable Codewords <i>Chinchun Chang, Chiachen Lin, Junbin Yeh</i>	567
--	-----

Reversible Data Hiding for Image Based on Histogram Modification of Wavelet Coefficients <i>Xiaoping Liang, Xiaoyun Wu, Jiwu Huang</i>	573
An Image Steganography Using Pixel Characteristics <i>Young-Ran Park, Hyun-Ho Kang, Sang-Uk Shin, Ki-Ryong Kwon</i>	581
Alternatives for Multimedia Messaging System Steganography <i>Konstantinos Papapanagiotou, Emmanouel Kellinis, Giannis F. Marias, Panagiotis Georgiadis</i>	589
Error Concealment for Video Transmission Based on Watermarking <i>Shuai Wan, Yilin Chang, Fuzheng Yang</i>	597
Applying the AES and Its Extended Versions in a General Framework for Hiding Information in Digital Images <i>Tran Minh Triet, Duong Anh Duc</i>	605
An Image Hiding Algorithm Based on Bit Plane <i>Bin Liu, Zhitang Li, Zhanchun Li</i>	611
A Blind Audio Watermarking Algorithm Robust Against Synchronization Attack <i>Xiangyang Wang, Hong Zhao</i>	617
Semi-fragile Watermarking Algorithm for Detection and Localization of Temper Using Hybrid Watermarking Method in MPEG-2 Video <i>Hyun-Mi Kim, Ik-Hwan Cho, A-Young Cho, Dong-Seok Jeong</i>	623
Public Watermarking Scheme Based on Multiresolution Representation and Double Hilbert Scanning <i>Zhiqiang Yao, Liping Chen, Rihong Pan, Boxian Zou, Licong Chen</i>	629
Performance Evaluation of Watermarking Techniques for Secure Multimodal Biometric Systems <i>Daesung Moon, Taehae Kim, SeungHwan Jung, Yongwha Chung, Kiyoung Moon, Dosung Ahn, Sang-Kyoon Kim</i>	635
An Improvement of Auto-correlation Based Video Watermarking Scheme Using Independent Component Analysis <i>Seong-Whan Kim, Hyun-Sung Sung</i>	643
A Digital Watermarking Technique Based on Wavelet Packages <i>Chen Xu, Weiqiang Zhang, Francis R. Austin</i>	649

A Spectral Images Digital Watermarking Algorithm <i>Long Ma, Changjun Li, Shuni Song</i>	655
Restoration in Secure Text Document Image Authentication Using Erasable Watermarks <i>Niladri B. Puhon, Anthony T.S. Ho</i>	661
Web and Network Applications	
The Study of RED Algorithm Used Multicast Router Based Buffer Management <i>Won-Hyuck Choi, Doo-Hyun Kim, Kwnag-Jae Lee, Jung-Sun Kim</i>	669
Genetic Algorithm Utilized in Cost-Reduction Driven Web Service Selection <i>Lei Cao, Jian Cao, Minglu Li</i>	679
MacroOS: A Pervasive Computing Platform Supporting Context Awareness and Context Management <i>Xiaohua Luo, Kougen Zheng, Zhaohui Wu, Yunhe Pan</i>	687
A Frame for Selecting Replicated Multicast Servers Using Genetic Algorithm <i>Qin Liu, Chanle Wu</i>	695
On a Novel Methodology for Estimating Available Bandwidth Along Network Paths <i>Shaohe Lv, Jianping Yin, Zhiping Cai, Chi Liu</i>	703
A New AQM Algorithm for Enhancing Internet Capability Against Unresponsive Flows <i>Liyuan Zhao, Keqin Liu, Jun Zheng</i>	711
Client Server Access: Wired vs. Wireless LEO Satellite-ATM Connectivity; A (MS-Ro-BAC) Experiment <i>Terry C. House</i>	719
An Algorithm for Automatic Inference of Referential Integrities During Translation from Relational Database to XML Schema <i>Jinhyung Kim, Dongwon Jeong, Doo-Kwon Baik</i>	725
A Fuzzy Integral Method to Merge Search Engine Results on Web <i>Shuning Cui, Boqin Feng</i>	731

The Next Generation PARLAY X with QoS/QoE
Sungjune Hong, Sunyoung Han 737

A Design of Platform for QoS-Guaranteed Multimedia Services
 Provisioning on IP-Based Convergence Network
Seong-Woo Kim, Young-Chul Jung, Young-Tak Kim..... 743

Introduction of Knowledge Management System for Technical Support
 in Construction Industries
Tai Sik Lee, Dong Wook Lee, Jeong Hyun Kim 749

An Event Correlation Approach Based on the Combination of IHU and
 Codebook
Qihua Zheng, Yuntao Qian..... 757

Image and Signal Processing

Face Recognition Based on Support Vector Machine Fusion and
 Wavelet Transform
Bicheng Li, Hujun Yin 764

A Dynamic Face and Fingerprint Fusion System for Identity
 Authentication
Jun Zhou, Guangda Su, Yafeng Deng, Kai Meng, Congcong Li 772

Image Recognition for Security Verification Using Real-Time Joint
 Transform Correlation with Scanning Technique
Kyu B. Doh, Jungho Ohn, Ting-C Poon 780

Binarized Revocable Biometrics in Face Recognition
Ying-Han Pang, Andrew Teoh Beng Jin, David Ngo Chek Ling 788

Short Critical Area Computational Method Using Mathematical
 Morphology
Junping Wang, Yue Hao 796

A Robust Lane Detection Approach Based on MAP Estimate and
 Particle Swarm Optimization
Yong Zhou, Xiaofeng Hu, Qingtai Ye 804

MFCC and SVM Based Recognition of Chinese Vowels
Fuhai Li, Jinwen Ma, Dezhi Huang 812

A Spatial/Frequency Hybrid Vector Quantizer Based on a Classification in the DCT Domain <i>Zhe-Ming Lu, Hui Pei, Hans Burkhardt</i>	820
Removing of Metal Highlight Spots Based on Total Variation Inpainting with Multi-sources-flashing <i>Ji Bai, Lizhuang Ma, Li Yao, Tingting Yao, Ying Zhang</i>	826
Component-Based Online Learning for Face Detection and Verification <i>Kyoung-Mi Lee</i>	832
SPIHT Algorithm Based on Fast Lifting Wavelet Transform in Image Compression <i>Wenbing Fan, Jing Chen, Jina Zhen</i>	838
Modified EZW Coding for Stereo Residual <i>Han-Suh Koo, Chang-Sung Jeong</i>	845
Optimal Prototype Filters for Near-Perfect-Reconstruction Cosine-Modulated Filter Banks <i>Xuemei Xie, Guangming Shi, Xuyang Chen</i>	851
Fast Motion Estimation Scheme for Real Time Multimedia Streaming with H.264 <i>Chan Lim, Hyun-Soo Kang, Tae-Yong Kim</i>	857
Motion-Compensated 3D Wavelet Video Coding Based on Adaptive Temporal Lifting Filter Implementation <i>Guiguang Ding, Qionghai Dai, Wenli Xu</i>	863
Accurate Contouring Technique for Object Boundary Extraction in Stereoscopic Imageries <i>Shin Hyoung Kim, Jong Whan Jang, Seung Phil Lee, Jae Ho Choi</i>	869
Robust Object Tracking Based on Uncertainty Factorization Subspace Constraints Optical Flow <i>Yunshu Hou, Yanning Zhang, Rongchun Zhao</i>	875
Bearings-Only Target Tracking Using Node Selection Based on an Accelerated Ant Colony Optimization <i>Benlian Xu, Zhiquan Wang</i>	881
Image Classification and Delineation of Fragments <i>Weixing Wang</i>	887

A Novel Wavelet Image Coding Based on Non-uniform Scalar Quantization <i>Guoyuo Wang, Wentao Wang</i>	893
A General Image Based Nematode Identification System Design <i>Bai-Tao Zhou, Won Nah, Kang-Woong Lee, Joong-Hwan Baek</i>	899
A Novel SVD-Based RLS Blind Adaptive Multiuser Detector for CDMA Systems <i>Ling Zhang, Xian-Da Zhang</i>	905
New Electronic Digital Image Stabilization Algorithm in Wavelet Transform Domain <i>Jung-Youp Suk, Gun-Woo Lee, Kuhn-Il Lee</i>	911
Line Segments and Dominate Points Detection Based on Hough Transform <i>Z.W. Liao, S.X. Hu, T.Z. Huang</i>	917
The Study of the Auto Color Image Segmentation <i>Jian Zhuang, Haifeng Du, Jinhua Zhang, Sun'an Wang</i>	923
Regularized Image Restoration by Means of Fusion for Digital Auto Focusing <i>Vivek Maik, Jeongho Shin, Joonki Paik</i>	929
Fast Ray-Space Interpolation Based on Occlusion Analysis and Feature Points Detection <i>Gangyi Jiang, Liangzhong Fan, Mei Yu, Rangding Wang, Xien Ye, Yong-Deak Kim</i>	935
Non-parametric ICA Algorithm for Hybrid Sources Based on GKNN Estimation <i>Fasong Wang, Hongwei Li, Rui Li, Shaoquan Yu</i>	941
SUSAN Window Based Cost Calculation for Fast Stereo Matching <i>Kyu-Yeol Chae, Won-Pyo Dong, Chang-Sung Jeong</i>	947
An Efficient Adaptive De-blocking Algorithm <i>Zhiliang Xu, Shengli Xie, Youjun Xiang</i>	953
Facial Features Location by Analytic Boosted Cascade Detector <i>Lei Wang, Beiji Zou, Jiaguang Sun</i>	959

New Approach for Segmentation and Pattern Recognition of Jacquard Images <i>Zhilin Feng, Jianwei Yin, Zhaoyang He, Wuheng Zuo, Jinxiang Dong</i>	965
Nonstationarity of Network Traffic Within Multi-scale Burstiness Constraint <i>Jinwu Wei, Jiangxing Wu</i>	971
Principle of Image Encrypting Algorithm Based on Magic Cube Transformation <i>Li Zhang, Shiming Ji, Yi Xie, Qiaoling Yuan, Yuehua Wan, Guanjun Bao</i>	977
A Study on Motion Prediction and Coding for In-Band Motion Compensated Temporal Filtering <i>Dongdong Zhang, Wenjun Zhang, Li Song, Hongkai Xiong</i>	983
Adaptive Sampling for Monte Carlo Global Illumination Using Tsallis Entropy <i>Qing Xu, Shiqiang Bao, Rui Zhang, Ruijuan Hu, Mateu Sbert</i>	989
Applications	
Incremental Fuzzy Decision Tree-Based Network Forensic System <i>Zaiqiang Liu, Dengguo Feng</i>	995
Robust Reliable Control for a Class of Fuzzy Dynamic Systems with Time-Varying Delay <i>Youqing Wang, Donghua Zhou</i>	1003
Using Concept Taxonomies for Effective Tree Induction <i>Hong Yan Yi, B. de la Iglesia, V.J. Rayward-Smith</i>	1011
A Similarity-Based Recommendation Filtering Algorithm for Establishing Reputation-Based Trust in Peer-to-Peer Electronic Communities <i>Jingtao Li, Yinan Jing, Peng Fu, Gendu Zhang, Yongqiang Chen</i>	1017
Automatic Classification of Korean Traditional Music Using Robust Multi-feature Clustering <i>Kyu-Sik Park, Youn-Ho Cho, Sang-Hun Oh</i>	1025
A Private and Efficient Mobile Payment Protocol <i>Changjie Wang, Ho-fung Leung</i>	1030

Universal Designated-Verifier Proxy Blind Signatures for E-Commerce <i>Tianjie Cao, Dongdai Lin, Rui Xue</i>	1036
An Efficient Control Method for Elevator Group Control System <i>Ulvi Dagdelen, Aytekin Bagis, Dervis Karaboga</i>	1042
Next Generation Military Communication Systems Architecture <i>Qijian Xu, Naitong Zhang, Jie Zhang, Yu Sun</i>	1048
Early Warning for Network Worms <i>Antti Tikkanen, Teemupekka Virtanen</i>	1054
Skeleton Representation of Character Based on Multiscale Approach <i>Xinhua You, Bin Fang, Xinge You, Zhenyu He, Dan Zhang, Yuan Yan Tang</i>	1060
Channel Equalization Based on Two Weights Neural Network <i>Wenming Cao, Wanfang Chai, Shoujue Wang</i>	1068
Assessment of Uncertainty in Mineral Prospectivity Prediction Using Interval Neutrosophic Set <i>Pawalai Kraipeerapun, Chun Che Fung, Warick Brown</i>	1074
Ring-Based Anonymous Fingerprinting Scheme <i>Qiang Lei, Zhengtao Jiang, Yumin Wang</i>	1080
Scalable and Robust Fingerprinting Scheme Using Statistically Secure Extension of Anti-collusion Code <i>Jae-Min Seol, Seong-Whan Kim</i>	1086
Broadcast Encryption Using Identity-Based Public-Key Cryptosystem <i>Lv Xixiang, Bo Yang</i>	1092
Multimedia Digital Right Management Using Selective Scrambling for Mobile Handset <i>Goo-Rak Kwon, Tea-Young Lee, Kyoung-Ho Kim, Jae-Do Jin, Sung-Jea Ko</i>	1098
Design and Implementation of Crypto Co-processor and Its Application to Security Systems <i>HoWon Kim, Mun-Kyu Lee, Dong-Kyue Kim, Sang-Kyoon Chung, Kyoil Chung</i>	1104
Continuous Speech Research Based on HyperSausage Neuron <i>Wenming Cao, Jianqing Li, Shoujue Wang</i>	1110

Variable-Rate Channel Coding for Space-Time Coded MIMO System <i>Changcai Han, Dongfeng Yuan</i>	1116
A New Watermarking Method Based on DWT <i>Xiang-chu Feng, Yongdong Yang</i>	1122
Efficient Point Rendering Method Using Sequential Level-of-Detail <i>Daniel Kang, Byeong-Seok Shin</i>	1127
Construction of a Class of Compactly Supported Biorthogonal Multiple Vector-Valued Wavelets <i>Tongqi Zhang, Qingjiang Chen</i>	1134
Metabolic Visualization and Intelligent Shape Analysis of the Hippocampus <i>Yoo-Joo Choi, Jeong-Sik Kim, Min-Jeong Kim, Soo-Mi Choi, Myoung-Hee Kim</i>	1140
Characteristic Classification and Correlation Analysis of Source-Level Vulnerabilities in the Linux Kernel <i>Kwangsun Ko, Insook Jang, Yong-hyeog Kang, Jinseok Lee, Young Ik Eom</i>	1149
Author Index	1157

Empirical Analysis of Database Privacy Using Twofold Integrals

Jordi Nin and Vicenç Torra

IIIA-CSIC Campus UAB s/n, 08193 Bellaterra, Catalonia, Spain
{jnin, vtorra}@iiia.csic.es

Abstract. Record linkage is a technique for linking records of different files that correspond to the same individual. Traditional record linkage methods needs that the files have some common variables to permit such link. In this paper we study the possibility of applying record linkage techniques when the files do not share variables. In this case we establish links based on structural information. For extracting this structural information we use in this paper twofold integrals.

1 Introduction

Information about individuals is usually distributed among databases and represented in an heterogeneous way. Due to this, there is an increasing need of data cleaning and data integration techniques. Nowadays there are a lot of methods for data integration [6]. Re-identification algorithms [14] are one of such methods. They are used to identify the structures that different files or databases share (*e.g.* identify records that belong to the same individual, identify variables that appear under different names in different datafiles). The goal of record linkage algorithms, one of such re-identification techniques, is to link the records that belong to the same individual but that can be found in two (or more) files or databases.

Such algorithms are used for several different purposes. One of their uses is risk assessment in privacy preserving data mining (PPDM) [1] and statistical disclosure control (SDC) [12]. In this case, record linkage methods permit to evaluate whether a protection mechanism provides enough protection to providers of sensitive information (to know whether disclosure is guaranteed to such data providers). Another use of record linkage is data integration. In this case, record linkage permits to join two (or more) files (or databases). Then, a new file is obtained that gathers all relevant information about the individuals contained in the files. Data is consolidated when inconsistencies are found in the files.

Classical record linkage methods need that the files to be linked contain some common variables. These variables are used for establishing the links between records. For these methods, the main difficulty for having a good performance is that the files can contain errors (*e.g.*, the age and the birthday of an individual is not consistent in the two files to be re-identified). These errors may be accidental (*e.g.*, due to an incorrect manipulation of the data) or intentional (*e.g.* to protect sensitive information as in PPDM).

One of the goals of advanced record linkage methods is the re-identification of records when files are not described using common variables. In this case, re-identification is based on the so-called structural information. In [11] we show that variable re-identification is possible using OWA operators [15]. OWA operators are used for extracting structural information from both files, and re-identification is done using such structural information.

Re-identification when files do not share variables is based in a set of assumptions [11]:

A.1 : Both files contain information on a common set of individuals.

A.2 : Data in both sources contain, implicitly, similar structural information.

We can study different forms to represent the structural information contained in the data. Depending on such structural information, we can develop different record linkage methods. In this work we consider the use of twofold integrals for re-identification. This is based on the following:

1. Aggregation operators can be used to build summaries from data.
2. Structural information is expressed in terms of some numerical aggregates.

So, we assume that the structural information can be expressed in terms of some numerical aggregations obtained using aggregation operators. And, more particularly, we consider the use of fuzzy integrals for this purpose.

Note that the first assumption above implies that we can re-identify some records because the files have records to link. The second assumption implies that there is some information about the individuals that is kept approximately constant in the two files and, thus, it is independent of the actual data in the files. We call *structural information* to this kind of *data-independent* information. The additional considerations that justify the use of aggregation operators in the re-identification process are described and justified in detail in [9].

In this paper we focus on the use of twofold integrals and we present an empirical analysis of their performance for extracting structural information. We show their effectiveness in extracting structural information. We show that using numerical values to represent structural information is useful for linking records and, thus, it permits to obtain an estimation of the level of privacy of a protected database. This work extends previous results [5, 9] where Sugeno and Choquet integrals were used for the same purpose. We will show that the use of the twofold integral leads also to good results and we compare its results with previous approaches.

The structure of the paper is as follows. In Section 2 we describe some elements that are needed latter on. Then, in Section 3 we introduce our approach to record linkage. Section 4 describes some of the experiments performed. Then, the paper finishes with some conclusions and description of future work.

2 Preliminaries

This section describes the twofold integral as well as standard re-identification methods. These methods are used by our algorithm as we transform the problem

of re-identification without non-common variables into the standard problem of re-identification with common variables.

We start with the definition of the twofold integral, and then, in Section 2.2 we review re-identification methods

2.1 The Twofold Integral

The twofold integral is a fuzzy integral that aggregates a function with respect to a fuzzy measure. Both definitions are given below. See *e.g.*, [3] for details on fuzzy measures and fuzzy integrals.

Definition 1. [7] Let X be a set, then a fuzzy measure μ on X is a real valued set function, $\mu : 2^X \rightarrow [0, 1]$ with the following properties:

1. $\mu(\emptyset) = 0$, $\mu(X) = 1$
2. $\mu(A) \leq \mu(B)$ when $A \subset B$ and $A, B \in 2^X$.

Definition 2. [10, 4] Let μ_C and μ_S be two fuzzy measures on X , then the twofold integral of a function $f : X \rightarrow [0, 1]$ with respect to the fuzzy measures μ_S and μ_C is defined by:

$$TI_{\mu_S, \mu_C}(f) = \sum_{i=1}^n \left(\left(\bigvee_{j=1}^i f(x_{s(j)}) \wedge \mu_S(A_{s(j)}) \right) (\mu_C(A_{s(i)}) - \mu_C(A_{s(i+1)})) \right)$$

where s in $f(x_{s(i)})$ indicates that the indices have been permuted so that

$$0 \leq f(x_{s(1)}) \leq \dots \leq f(x_{s(n)}) \leq 1, A_{s(i)} = \{x_{s(i)}, \dots, x_{s(n)}\}, A_{s(n+1)} = \emptyset.$$

2.2 Re-identification Methods

The goal of classical re-identification methods is to link records that while belonging to different files (two or more) correspond to the same individual. In the classical approach records are described using the same variables. *E.g.*, we have one file with the customers of a company and another file with the providers. Both files on customers and providers contain the fields *name*, *address*, *city*, *state*. Besides both files contain other data specific to either customers or providers. Record linkage algorithms permit to link customers with providers when they refer to the same individual (or company) using the common variables. Difficulties in the re-identification process are due to the errors in names, address, and so on. These errors can be due to misspellings (*e.g.*, John vs. Jonh) or changes of location (*e.g.*, Clark Kent previously living in Smallville has moved to Metropolis).

Two main approaches have been used for re-identification in the standard case: Probabilistic and distance-based record linkage. See [8, 13, 14] for more details.

2.3 Record Linkage Evaluation

To compare the performance of a record linkage method, there are different approaches. One of them is to compare the performance of the method with alternative ones. Another is to consider the probability of having a particular number of re-identifications using a *random strategy*. The next proposition establishes these probabilities:

Proposition 1. [11] *If A and B both contain n records corresponding to the same set of n individuals, the probability of correctly re-identifying exactly r individuals by a random strategy is*

$$\frac{\sum_{v=0}^{n-r} \frac{(-1)^v}{v!}}{r!} \quad (1)$$

For illustration of the order of these probabilities. We have that the probability of linking exactly 5 records over 100 (at random) is 0.00306566, and that the probability of linking 10 records over 100 (at random) is 1.01378E-7. Besides, the probability of linking more than 5 or more than 10 over 100 is, respectively, 0.00365985 and 1.11425E-7.

3 Record Linkage Based on Numerical Representatives

As we have explained in the introduction, in this paper we have considered the re-identification problem when the files do not share variables. In this case although we can not apply the methods showed in Section 2.2 we reduce the problem to one of such type.

From now on, we will consider that we have two files A and B , and that these files have different variables but that records belong to the same individuals. Then, if we want to apply traditional record linkage methods we need to transform the original files A and B in two different files A' and B' that have the same variables. For achieving this we use the aggregation operators as they permit to build summaries of the records in the original files.

In short, our approach works as follows:

1. Select a summarization function S with a particular parameter p
2. Apply the function S with parameter p to all records a in A , and to all records b in B . We obtain in this way, values $a'_{S,p} := S_p(a)$ and $b'_{S,p} := S_p(b)$.
3. Repeat the two previous steps n times
4. Define the file A' in term of the values $a'_{S,p}$ computed in the previous step.
Do the same to define the file B'
5. Apply standard record linkage to files A' and B'

Note that when we use the same n pairs of (S, p) to records in A and B , we obtain new records with n representatives each. Therefore, we can apply standard record linkage. This is, we can apply the distance-based and the probabilistic record linkage described in Section 2.2.

In a previous work we used Choquet and Sugeno integrals for the purpose of record summarization. In this work we study the use of the twofold integral. Choquet and Sugeno integrals as well as twofold integrals belong to the family of fuzzy integrals. The use of such integrals for building the representatives is motivated for the following reasons:

- Fuzzy integrals, with appropriate fuzzy measures, can be used to aggregate information regardless its dimensionality. This is useful in our case, as we can consider two files with a different number of variables.
- Fuzzy integrals, with appropriate fuzzy measures, are symmetric operators. This is adequate when there is no *a priori* information on the variables (*i.e.*, we cannot say that one variable is more relevant than another one).
- Fuzzy integrals are parametric (the fuzzy measure is their parameter), and thus, several different representatives (different summaries) can be built from the same data using different parameters.

The study of the twofold integral was motivated by the fact that this operator contains two parameters (instead of one as the other two fuzzy integrals – the Sugeno and the Choquet integral). Therefore, the twofold integral permits a greater degree of flexibility.

4 Experiments

To analyse our approach for record linkage based on structural information, we have tested it using eight different problems. These are the same problems studied in [9, 5]. Problems were obtained from the UCL repository and the Data Extraction System (DES) from the U. S. Census Bureau. In particular, the following files were considered from the UCL repository: Iris, Abalone, Ionosphere, Dermatology, Boston Housing Data, Waste-water treatment plant, Wisconsin Diagnostic Breast Cancer (WDBC); and the 1995 - Current Population Survey was considered from DES.

To test our approach we have partitioned the files downloaded into two different files in such a way that both files contained exactly the same records but only some of the attributes. The partition has been done considering only the numerical attributes with a high correlation coefficient (greater than 0.7). Other attributes have been discarded.

Before applying the record linkage method, we have included a pre-processing data step that consisted on the normalization/standardization of the raw data. The two following alternatives have been considered:

- Ranging: Translation into the $[0, 1]$ range.
- Standardization: Normalization using the mean and the standard deviation.

As shown in Definition 2, the twofold integral requires two fuzzy measures. In order to have a symmetric integral, the fuzzy measures should be symmetric. It is well known that all symmetric fuzzy measures can be expressed in terms of a distortion function f as follows: $\mu(A) = f(|A|/N)$ where $|A|$ denotes the

Table 1. Average results for 30 records (first column) and 100 records (second column) when using the twofold integral and average results for 30 records (third column) and 100 records (fourth column) when using the Choquet integral for extracting the structural information. Average of 10 tests for each parameter combination

	R/P	R/D	S/P	S/D	R/P	R/D	S/P	S/D	R/P	R/D	S/P	S/D	R/P	R/D	S/P	S/D
abalone	3.3	2.9	5.64	4.93	3.13	3.10	5.67	5.87	4.69	2.79	5.80	3.83	5.24	4.30	6.33	5.97
census	5.66	6.03	5.04	5.48	6.32	5.70	8.21	8.21	5.45	4.62	5.86	4.20	6.50	5.21	9.53	7.73
dermat.	0.84	1	1.12	1.04	1.03	1.06	1	1.15	1.00	1.00	1.70	1.18	1.60	1.70	1.17	1.00
housing	2.18	1.77	3.4	3.13	2	1.44	3.13	2.8	2.77	2.36	4.27	3.09	2.78	1.87	5.20	3.85
ionos.	8.35	7.38	5.54	3.83	17.29	15.39	8.48	7.44	7.89	5.75	5.00	2.70	16.90	11.17	7.93	4.10
iris	3.33	2.62	3.24	2.53	3.15	3.39	4.38	4.04	3.35	2.17	3.80	2.55	3.95	2.75	3.75	3.20
water-tr.	4.33	4.32	5.96	5.04	4.22	3.8	8.56	7.57	4.10	3.73	6.96	5.20	4.13	3.26	10.13	8.67
wdbc	5.27	5.97	6.26	6.84	5.5	7.1	9.64	10.64	4.76	4.80	8.73	7.50	5.10	6.06	15.00	13.73

cardinality of the set A , and N is the cardinality of the reference set X . Here, the function f should be non-decreasing (so that the measure is monotonic) and as such that $f(0) = 0$ and $f(1) = 1$ (so that boundary conditions hold).

In our case, we have considered the following three parametric distortion functions:

$$\begin{aligned}
 Q_1^\alpha(x) &= x^\alpha \text{ for } \alpha = 1/5, 3/5, \dots, 9/5 \\
 Q_2^\alpha(x) &= 1/(1 + e^{(\alpha-x)*10}) \text{ for } \alpha = \{0, 0.2, \dots, 0.8\} \\
 Q_3^\alpha(x) &= \begin{cases} 0 & \text{if } x \leq \alpha \\ 1 & \text{if } x > \alpha \end{cases}
 \end{aligned}$$

So, we have defined the measures $\mu_1^\alpha(A) = Q_1^\alpha(|A|/N)$, $\mu_2^\alpha(A) = Q_2^\alpha(|A|/N)$ and $\mu_3^\alpha(A) = Q_3^\alpha(|A|/N)$.

As the twofold integral requires two fuzzy measures, we have considered the 9 pairs that can be built from $Q_1^\alpha(x)$, $Q_2^\alpha(x)$ and $Q_3^\alpha(x)$. This is, $(\mu_1^\alpha, \mu_1^\alpha)$, $(\mu_1^\alpha, \mu_2^\alpha)$, \dots , $(\mu_3^\alpha, \mu_3^\alpha)$. Each pair has been applied to each of the problem tests described above, and for having statistical significance (as we select records at random), we have done the experiment 10 times. As described above, we have also considered 2 different types of normalization for each problem. Accordingly, we have done, in total, 360 tests ($9*10*2$) for each problem. The results of these tests are illustrated in Table 1. In this table, R and S stands for Ranging and Standardization; P and D stand for Probabilistic Record Linkage and Distance-based Record Linkage.

As we show in the Table 1, the approach obtains good results for several problems when the number of re-identifications are compared with the results obtained by random record linkage (the probabilities computed according to Proposition 1). In particular, the best results obtained with our approach are for the Ionosphere Database. For both the case of 30 records as well as the case of 100 records. In the first case, our approach obtains 8 hits over 30 records and 17 hits over 100 records. The probability to have these results at random is $9.124E-6$ in the case of 30 records and $1.34E-15$ in the case of 100 records.

Other problems where good results were obtained are the following: wdbc (Wisconsin Diagnostic Breast Cancer) problem, where we have obtained 10 hits over 100 records (Probability at random equal to $1.014E-7$) and 6 hits over 30

Table 2. Deviations for 30 records (first column) and 100 records (second column) when using the twofold integral and deviations for 30 records (third column) and 100 records (fourth column) when using the Choquet integral for extracting the structural information. Deviation of 10 test for each parameter combination

	R/P	R/D	S/P	S/D	R/P	R/D	S/P	S/D	R/P	R/D	S/P	S/D	R/P	R/D	S/P	S/D
abalone	2.91	3.07	3.84	3.54	2.65	2.6	3.86	3.94	3.09	2.52	3.80	3.65	2.30	2.53	4.56	5.45
census	2.34	2.35	1.15	2.46	2.39	2.86	2.15	2.89	2.47	2.93	2.01	2.80	2.38	3.71	2.31	3.67
dermat.	0.6	0.88	0.91	0.56	0.3	0.35	0.94	0.86	1.35	0.78	0.95	0.87	1.05	1.22	0.75	0.87
housing	1.49	1.47	2.03	1.71	1.51	1.06	1.9	1.64	1.63	1.49	2.16	1.51	1.70	0.87	1.79	1.84
ionos.	2.17	2.67	2.04	1.90	4.55	5.87	2.34	3.2	2.11	2.77	2.42	2.12	4.6	6.64	2.98	3.01
iris	1.05	0.97	0.84	1.19	1.16	1.35	1.09	1.3	1.00	1.13	0.95	1.50	1.05	1.16	1.52	1.24
water-tr.	1.81	2.3	1.89	1.78	2.27	1.71	2.78	2.53	2.09	2.62	2.76	2.37	2.84	2.33	1.99	2.34
wdbc	1.88	2.29	1.96	2.57	2.36	2.26	3.39	4.02	1.22	2.76	2.66	3.38	2.73	2.52	5.00	4.87

records ($P = 5.109E-10$ at random); Water-treatment and Census where we have obtained 8 hits over 100 records (Probability at random equal to $9.124E-6$); and Abalone (and also Census) where we have obtained 5 hits over the set of 30 records (Probability at random equal to $3.066E-3$). Note that the number of re-identifications here corresponds to the average over the 10 executions.

In a previous work [9], we studied the application of the Choquet and Sugeno integrals for the same problem. Comparing the results, we can observe that the results obtained with the twofold are slightly better than the ones of the Sugeno integral, but that they cannot improve the best results obtained for the Choquet integral.

Tables 1 and 2 show, respectively, the mean values and the deviations obtained for each of the test files using the Choquet integral. Table 1 gives the average number of re-identifications obtained with the twofold integral over 10 tests, and Table 2 give the corresponding deviations.

Although that in only a few parameterizations the twofold integral is better, we can observe that the deviation on the number of re-identified records is smaller in the case of the twofold integral than in the case of the other integrals. Therefore, we have that the twofold integral leads to more stable results than the Choquet integral. Thus, in general, the results of the Choquet integral are more dependent on the parameters selected, and so the selection of such parameters is more crucial. Instead, the results of the twofold integral are more robust.

5 Conclusions and Future Work

In this paper, we have introduced a new approach for extracting structural information from data using the twofold integral. We have shown that this kind of structural information permits some re-identifications when the files do not share variables.

To complete this work we have to consider additional tests that include some categorical attributes. Additionally, we should improve our software so that it outlines the correct links among all those proposed.

Acknowledgements

This work was partly funded by the Spanish Ministry of Education and Science under project SEG2004-04352-C04-02 "PROPRIETAS".

References

1. Agrawal, R., Srikant, R.: Privacy Preserving Data Mining. Proc. of the ACM SIGMOD Conference on Management of Data (2000) 439-450
2. Data Extraction System, U.S. Census Bureau, <http://www.census.gov/DES/www/welcome.html>
3. Mesiar, R., Mesiarová, A.: Fuzzy Integrals, Modeling Decisions for Artificial Intelligence. Lecture Notes in Artificial Intelligence 3131 (2004) 7-14
4. Narukawa, Y., Torra, V.: Twofold integral and Multi-step Choquet integral. Kybernetika, 40:1 (2004) 39-50
5. Nin, J., Torra, V.: Towards the use of OWA operators for record linkage. Proc. of the European Soc. on Fuzzy Logic and Technologies, in press (2005)
6. Rahm, E., Bernstein, P. A.: A survey of approaches to automatic schema matching. The VLDB Journal 10 (2001) 334-350
7. Sugeno, M.: *Theory of Fuzzy Integrals and its Applications*. (PhD Dissertation). Tokyo Institute of Technology, Tokyo, Japan (1974)
8. Torra, V., Domingo-Ferrer, J.: Record linkage methods for multidatabase data mining, in V. Torra (Ed.), *Information Fusion in Data Mining*, Springer (2003) 101-132
9. Torra, V., Nin, J.: Record linkage in databases not sharing attributes using fuzzy integrals, manuscript (2005)
10. Torra, V.: Twofold integral: A Choquet integral and Sugeno integral generalization, Butlletí de l'Associació Catalana d'Intel·ligència Artificial, 29 (2003) 13-19 (in Catalan). Preliminary version: IIIA Research Report TR-2003-08 (in English).
11. Torra V.: OWA operators in data modeling and re-identification, IEEE Trans. on Fuzzy Systems, 12:5 (2004) 652-660
12. Willenborg, L., de Waal, T.: *Elements of Statistical Disclosure Control*, Lecture Notes in Statistics, Springer-Verlag (2001)
13. Winkler, W. E.: Data Cleaning Methods, Proc. SIGKDD 2003, Washington (2003)
14. Winkler, W. E.: Re-identification methods for masked microdata, Privacy in Statistical Databases 2004, Lecture Notes in Computer Science 3050 (2004) 216-230
15. Yager, R. R.: On ordered weighted averaging aggregation operators in multi-criteria decision making. IEEE Trans. Syst., Man, Cybern., 18 (1988) 183-190

Intrusion Detection Alert Verification Based on Multi-level Fuzzy Comprehensive Evaluation

Chengpo Mu, Houkuan Huang, and Shengfeng Tian

School of Computer and Information Technology,
Beijing Jiaotong University, Beijing 100044, China
combinatorics@126.com

Abstract. Alert verification is a process which compares the information referred by an alert with the configuration and topology information of its target system in order to determine if the alert is relevant to its target system. It can reduce false positive alerts and irrelevant alerts. The paper presents an alert verification approach based on multi-level fuzzy comprehensive evaluation. It is effective in achieving false alert and irrelevant alerts reduction, which have been proved by our experiments. The algorithm can deal with the uncertainties better than other alert verification approaches. The relevance score vectors obtained from the algorithm facilitate the formulation of fine and flexible security policies, and further alert processing.

1 Introduction

Although intrusion detection systems have been studied and developed over 20 years, they are still on the stage of development. Many parts of the system are not perfect and need improvement. The primary weaknesses of present IDSs are as follows.

Firstly, most current IDSs generate a great number of false positive alerts and irrelevant alerts. Secondly, all the current IDSs focus on low-level attacks or anomalies; none of them can capture the logical steps or strategies behind these attacks [1]. Finally, the response capabilities of current IDSs are weak. After detecting attacks, most IDSs do nothing except send alerts.

The solution for the first problem is the foundation for solving other two problems. It is impossible to process IDS alerts automatically and in depth if there are too many false and irrelevant alerts. One of the important principles in the intrusion response field is to reduce the negative impact of intrusion response measures. Instead of protecting a network system in time, a response measure from an automated intrusion response system (AIRS) may result in more damage to the network system than a false positive alert itself because the decision on choosing the response measure is based on the false alert. Irrelevant alerts cause unnecessary responses that impact on the performance of protected systems.

2 Related Work

Since the beginning of the 21st century, more attention has been paid to research into post-intrusion detection techniques. Post-intrusion detection techniques, including alert verification, alert aggregation, alert correlation, intrusion response, forensics, etc., encompass algorithms and approaches to process alerts generated by IDSs. These techniques can greatly improve the performance of IDSs and make up for the shortcomings of pre-intrusion detection techniques. A few post-intrusion detection techniques have been proposed to reduce false positive alerts and irrelevant alerts, such as alert correlation [1, 2] and alert relevance verification [3, 4, 5, 6] etc.

Some of them process alerts only according to the information carried by alerts themselves. We argue that alerts should be processed according to all available information. Here we introduce a few alert processing models that are closely related to our approach that make use of both the information from IDSs alerts and the information of protected system. The idea of such an approach is to compare the information referred by an alert with the configuration information of its target host. Therefore, the approach is called alert verification.

Robert et al have developed the SCYLLARUS system to aggregate IDSs alerts based on their impact on the site's security goals [4]. The system makes use of both alert messages and information from protected systems to provide a comprehensive security overview.

The M2D2 model for IDS alert correlation supplies four information types: information of the monitored system, information about the vulnerabilities, information about the security tools used for the monitoring, and information about the events observed [5]. The four types of information allow rich alert correlation approaches that could provide high quality alerts with less irrelevant and false message in them.

M-Correlator proposed by Porras is designed to find those alerts that represent the greatest threat to the health and security of protected networks by ranking a stream of security alerts [6]. A relevance score, which assess per alert the likeli-

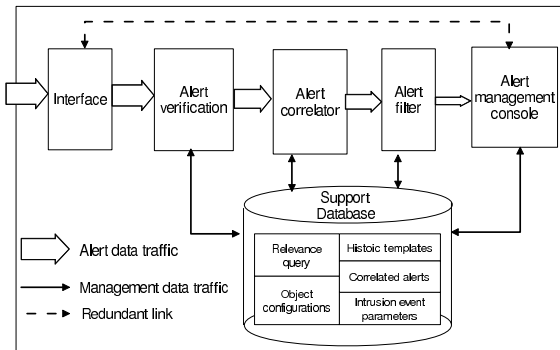


Fig. 1. The architecture of IDAMS

hood of successful intrusion, is produced through a comparison of the alert target's known topology against the vulnerability requirements of the incident type.

The paper [3] discusses the benefits and limitations of such an alert correlation approach based on vulnerabilities of protect systems. It categorizes the limitation of a system that correlates IDS and Vulnerability Assessment (VA) data to reduce false positive into nine categories.

Our proposed approach takes not only the vulnerability relevance between alerts and target system topologies or configuration but also the operating system relevance and network service relevance into account. The approach is based on multi-level fuzzy comprehensive evaluation that can deal with uncertainties better than above models. It has been used in the alert verification module of Intrusion Detection Alert Management System (IDAMS) developed by our lab. The architecture and components of IDAMS are shown in Fig. 1.

3 Alert Verification

3.1 Concepts and Idea of Alert Verifications

If an attack exploits a flaw, weakness or vulnerability that exists in the operating system or application service of a target host, then we conclude that the alert (or alerts) caused by the attack is relevant to the target system. The more relevant an alert is to a target system, the greater is the possibility that the attack will succeed. Here we introduce two notions.

Definition 1. The relevance information of an alert is related to the operating systems, the network service applications and the vulnerability messages referred by the alert.

IDAMS maintains a relevance information query table that is built from the ICAT vulnerability database [7] and manual inputs. ICAT uses the CVE/CAN naming convention and can provide more information than CVE. According to the IDS alert name and signature, the reference information for an alert can be acquired by querying the table.

Definition 2. The relevance score Rs is the degree of adjacency (or distance) between the reference information for an alert and the configuration of the target system.

The configuration of each host in the protected network is stored in the object configuration table of the support database. The table is maintained using a vulnerability assessment (VA) tool and a network management tool. The configuration information of a host is shown in Fig. 2.

Relevance scores represent the likelihood of a successful attack. According to alert relevance scores, alerts could be processed in different ways. For example, false positive alerts and irrelevant alerts can be filtered out or tagged for their low relevance scores. In contrast, those alerts that represent the greatest threat can be found by ranking a stream of IDS alerts according to their relevance scores. Therefore, the idea of building the Alert verification is to calculate relevance scores of alerts.

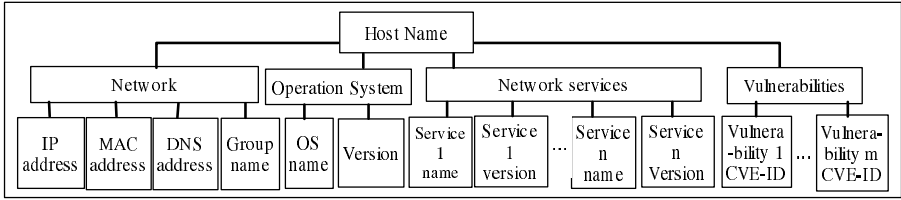


Fig. 2. Host configuration information

3.2 Fuzzy Comprehensive Evaluation (FCE)

When we calculate an alert’s relevance score, there are a lot of uncertainties and difficulties because of false positive, false negative and inexact information both in alerts of IDSs and reports of VA tools. The limitations of such a system are categorized in 9 categories that combine false positive, false negative and true IDS alerts with false positive, false negative and true vulnerabilities [3]. According to our research and experiences, the following facts can be obtained:

- *Vulnerability uncertainty > Service application uncertainty > Operating system uncertainty.*
- *Version uncertainty of OS or service > Name uncertainty of OS or service.*

A mathematical model based on FCE could resolve the problem of uncertainty among multiple factors and give an objective evaluation result by comprehensively considering multiple related factors.

Suppose $U = \{u_1, u_2, \dots, u_n\}$ and $V = \{v_1, v_2, \dots, v_m\}$ where U is the set of evaluation factors of FCE, and V is the set of evaluation remarks of FCE. From[8], we get the following model:

$$B = K \circ R \tag{1}$$

where $B = (b_1, b_2, \dots, b_m)$ is a subset on V , $K = (k_1, k_2, \dots, k_n)$ is a subset on U , R is a fuzzy matrix, as follow:

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1m} \\ r_{21} & r_{22} & \cdots & r_{2m} \\ \dots & \dots & \dots & \dots \\ r_{n1} & r_{n2} & \cdots & r_{nm} \end{bmatrix}.$$

Different mathematical models (e.g. $M(\vee, \wedge)$, $M(\bullet, +)$, $M(\bullet, \vee)$, $M(\wedge, \oplus)$) can be obtained by choosing different operators \circ in the formula (1). For the Alert verification in IDAMS, we choose the model $M(\wedge, \oplus)$, and get equation (2) from (1),

$$b_j = \bigoplus_{i=1}^n (k_i \wedge r_{ij}) = \sum_{i=1}^n k_i \wedge r_{ij} \tag{2}$$

where $r_{ij} (i = 1, 2, \dots, n; j = 1, 2, \dots, m;)$ represents the membership degree of the subject to remark v_j from the viewpoint of factor u_i . \wedge, \vee and \bullet are min, max and multiply operations respectively. \oplus is the operation whereby $a \oplus b = \min(1, a+b)$.

The feature of Equation (2) is that it emphasizes the influences of dominant factors with less uncertainties.

3.3 Evaluation Process

The determination of an alert relevance score by FCE approximately comprises the following steps:

Step 1. Determination of the set of evaluation factors and the set of evaluation remarks. In the alert verification process, a two-level FCE model and Eq. (2) are used. The evaluation factors are divided into three classes, as shown in Table 1.

$V = \{v_1, v_2, v_3\} = \{\textit{completely relevant}, \textit{relevant}, \textit{irrelevant}\}$ is the set of evaluation remarks. An attribute in the relevance information of an alert and an attribute in a host configuration are denoted as $R.attribute$ and $C.attribute$ respectively. Alerts that could not find corresponding relevance information and alerts that do not refer to any vulnerability (e.g. a scan) are dealt with as an unknown situation, as shown in the following tables. Here we follow the lowest risk principle to keep possible true alerts from being lost in this module.

Step 2. Calculation of membership degrees. The membership degrees of r_{ij} ($i = 1, 2, 3, 4, 5$ and $j = 1, 2, 3$) can be obtained from Tables 2 to 6.

Step 3. Determination of weight vectors. Weight vectors are usually decided based on the experience of experts. To reduce uncertainties in this two-level FCE model, we use the following idea to determine these weight vectors.

At the first level of FCE,

$K = (k_{os}, k_{service}, k_{vulnerability})$, and let $k_{os} \geq k_{service} \geq k_{vulnerability}$.

At the second level of FCE,

$K_1 = (k_{os.name}, k_{os.version})$, $K_2 = (k_{service.name}, k_{service.version})$, and make sure that $k_{os.name} \geq k_{os.version}$ and $k_{service.name} \geq k_{service.version}$.

In this way, OS relevance and service application relevance with fewer uncertainties play a more important role in determining the FCE results.

Step 4. Calculation of relevance scores. First, the evaluation is executed at the second level of the FCE:

$$K_1 = (k_{os.name}, k_{os.version}) \text{ and } R_1 = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \end{bmatrix}, \text{ then } B_1 = K_1 \circ R_1.$$

$$K_2 = (k_{service.name}, k_{service.version}) \text{ and } R_2 = \begin{bmatrix} r_{31} & r_{32} & r_{33} \\ r_{41} & r_{42} & r_{43} \end{bmatrix}, \text{ then } B_2 = K_2 \circ R_2.$$

$$K_3 = (k_{CVE-ID}) \text{ and } R_3 = [r_{51} \ r_{52} \ r_{53}], \text{ then } B_3 = K_3 \circ R_3.$$

Table 1. Evaluation factors

First-level factors	Second-level factors
OS relevance U_1	OS name relevance u_1
	OS version relevance u_2
Network service relevance U_2	Service name relevance u_3
	Service version relevance u_4
Vulnerability relevance U_3	CVE-ID relevance u_5

Table 2. Membership degrees of $r_{1j}(j = 1, 2, 3)$

Code	Conditions	r_{11}	r_{12}	r_{13}
<i>C11</i>	$C.OS.Name \in R.OS.Names$ ^a	1	0	0
<i>C12</i>	$R.OS.Name = unknown$	0	1	0
<i>C13</i>	$C.OS.Name \notin R.OS.Names$	0	0	1

^a The relevance information of an alert includes a list of OS names (e.g. Windows NT, Windows XP, etc.). This is also true for other attributes.

Table 3. Membership degrees of $r_{2j}(j = 1, 2, 3)$

Code	Conditions	r_{21}	r_{22}	r_{23}
<i>C21</i>	$C.OS.Version \in R.OS.Version$	1	0	0
<i>C22</i>	$R.OS.Version = unknown$	0	1	0
<i>C23</i>	$C.OS.Version \notin R.OS.Version$	0	0	1

Table 4. Membership degrees of $r_{3j}(j = 1, 2, 3)$

Code	Conditions	r_{31}	r_{32}	r_{33}
<i>C31</i>	$C.Service.Name \in R.Service.Names$	1	0	0
<i>C32</i>	$R.Service.Name = unknown$	0	1	0
<i>C33</i>	$C.Service.Name \notin R.Service.Names$	0	0	1

Table 5. Membership degrees of $r_{4j}(j = 1, 2, 3)$

Code	Conditions	r_{41}	r_{42}	r_{43}
<i>C41</i>	$C.Service.Version \in R.Service.Version$	1	0	0
<i>C42</i>	$R.Service.Version = unknown$	0	1	0
<i>C43</i>	$C.Service.Version \notin R.Service.Version$	0	0	1

Table 6. Membership degrees of $r_{5j}(j = 1, 2, 3)$

Code	Conditions	r_{51}	r_{52}	r_{53}
<i>C51</i>	$C.Vulnerability.CVE - IDs \cap R.Vulnerability.CVE - IDs \neq \phi$	1	0	0
<i>C52</i>	$R.Vulnerability.CVE - ID = unknown$	0	1	0
<i>C53</i>	$C.Vulnerability.CVE - IDs \cap R.Vulnerability.CVE - IDs = \phi$	0	0	1

The relevance score vector R_s can be decided at the first level of the FCE:

$$K = (k_{os}, k_{service}, k_{vulnerability}) \text{ and } R = \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}, \text{ then}$$

$$R_s = B = (b_1, b_2, b_3) = K \circ R. \quad (3)$$

Compared with traditional alert verification approaches that simply match a vulnerability exploited by an attack to a vulnerability present on a target system, which can produce only two different values (matched or unmatched), our approach can deal with 3^5 (243) different situations, which means that we can set up more detailed and flexible security policies to process alerts generated by IDSs. The model can overcome uncertainty in two aspects. First, it considers multiple factors rather than one, which can increase the algorithm robustness and objectivity. Second, the algorithm stresses the influence of dominant factors with fewer uncertainties and weakens the influence of factors with more uncertainties.

To reduce false positive alerts, we set a policy whereby an alert with a b_1 (completely relevant evaluation remark) less than $b_{1threshold}$ and b_2 (relevant evaluation remark) less than $b_{2threshold}$ is discarded.

4 Experiments and Analysis

In practical experiments, Snort 2.0 IDS and IDAMS were deployed on a subnet (xxx.71.75.130-xxx.71.75.180) in our lab, which connects to Internet. We collected alert data from 9:30 am to 5:30 pm and carried simulated attacks (such as Vertical Scan, CGI Vulnerability Scan and Serv-U Ftp MDTM overflow etc.) on the subnet from other different subnets in our campus network during the 8 hour of data collection.

Table 7. The experiments results of alert verification

	$b_{1threshold}$	Detection rate	False alert rate	Irrelevant alert rate
	$b_{2threshold}$			
Snort	N/A	65.6%	93.8%	86 %
I	0.5	65.6%	33.4%	15.3 %
D	0.8			
A	0.8	63.3%	26.2%	3.4 %
M	1.0			
S				

Note: $K=(0.45, 0.35, 0.2)$, $K_1=(0.6, 0.4)$
 $K_2=(0.6, 0.4)$.

The experiment results shown in Table 7 prove that the alert verification approach is effective in achieving false positive alert and irrelevant alert reduction. However, it reminds us that the detection rate is also reduced a little as the decreases of false positive alerts and irrelevant alerts although the sacrifice of detection rate is necessary. Therefore, we have to balance the detection rate and the false positive alert rate while we set thresholds.

5 Conclusions

The IDS alert verification based on multi-level fuzzy comprehensive evaluation can effectively reduce false positive alerts and irrelevant alerts and facilitate

the alerts deeply processing, such as alert correlations etc. The algorithm could deal with uncertainty well in the process of alert verification. The relevance score vector obtained in this model is very useful for further automated alert processing and the formulation of fine and flexible security policies.

References

1. Ning P., Cui Y.: An intrusion alert correlator based on prerequisites of intrusion. Technical Report TR-2002-01, Department of Computer Science, North Carolina State University (2002)
2. Qin X., Lee W.: Statistical causality of INFOSEC alert data. In: Recent Advances in Intrusion Detection. Lecture Notes in Computer Science, Vol. 2820. Springer-Verlag, Berlin Heidelberg New York (2003) 73-94
3. Gula R.: Correlating IDS Alerts with Vulnerability Information. Technical report, Tenable Network Security (2002)
4. Goldman R.P., Heimerdinger W., Haro S.A.: Information modeling for intrusion report aggregation. In: DARPA Information Survivability Conference and Exposition (DISCEX II) (2001)
5. Morin B., Mé L., Debar H., Ducassé M.: M2D2: A formal data model for IDS alert correlation. In: Recent Advances in Intrusion Detection. Lecture Notes in Computer Science, Vol. 2516. Springer-Verlag, Berlin Heidelberg New York (2002) 115-137
6. Porras P.A., Fong M.W., Valdes A.: A mission-impact-based approach to INFOSEC alarm correlation. In: 5th International Symposium on Recent Advances in Intrusion Detection (2002).
7. ICAT vulnerabilities database, available <http://icat.nist.gov/icat.cfm>
8. Xie J., Liu C.: The Methodology and Application of Fuzzy Mathematics (in Chinese). Hua Zhong University of Science and Technology Press, China (2000)

Improving the Scalability of Automatic Programming

Henrik Berg and Roland Olsson

Østfold University College, Remmen, N-1783 Halden, Norway
{henrik.berg, roland.olsson}@hiiof.no

Abstract. When automatically constructing large programs using program transformations, the number of possible transformations grows very fast. In this paper, we introduce and test a new way of combining several program transformations into one transformation, inspired by the combinatorial concept of Covering Arrays (CA).

We have equipped the ADATE automatic programming system with this new CA transformation and conducted a series of 18 experiments which show that the CA transformation is a highly useful supplement to the existing ADATE transformations.

1 Introduction

When designing the ADATE automatic programming system [6] more than one decade ago, it immediately became clear that combinations of neutral program transformations are highly useful in large scale combinatorial search of program space. Given a well written specification and sufficient of CPU time, ADATE can synthesize generally recursive functional programs to solve practically any small algorithmic problem from scratch, including the standard ones found in undergraduate textbooks on algorithm design. It is a general system for automatic functional programming with the ability to evolve recursive programs from first principles and automatically invent recursive help functions. It has been applied successfully to a number of tasks, including learning a simple human language [7], and as far as we know it is the only system capable of inventing a recursive function for algorithmic problems like permutation generation from scratch, including on-the-fly invention of recursive help functions.

However, as is always the case within the field of automatic programming, the number of potential programs to search through grows fast when the programs get larger. This is mainly due to the high number of possible combinations of program transformations. Our goal is therefore to more efficiently explore such combinations of program transformations for synthesis of large programs.

First, let us have a look at how the ADATE system works today. The core of the ADATE system is a set of programs, a *population*, of varying size. Initially, this population consists of one single program, the “empty” program, which is a program that always raises an exception when run. During an ADATE run, each program in the population is transformed numerous times, using the

program transformations described below, and all the resulting programs are tested and given a *fitness value*, using an evaluation function supplied by the user in the specification. For each program generated, ADATE compares its fitness value to those of the programs already in the population, and potentially adds it to the population, possibly removing some older, now inferior programs. The decision whether or not to add a new program to the population also depends on the program sizes, in that ADATE will prefer a smaller program over a larger program with the same fitness value. This reduces the risk of overfitting, since specially adapted programs usually are larger than more general programs.

As mentioned, ADATE creates programs from old programs using a set of *program transformations*. The most basic transformation is the *Replacement*, abbreviated R. This is, as the name suggests, simply a replacement of some expression in the old program with some new expression, possibly reusing a subexpression of the replaced expression. The number of possible expressions to insert is of course very large, and grows exponentially with the size of the expressions to use. Therefore the R transformation is combinatorially very expensive, and the inserted expressions should be kept as small as possible.

To achieve this, ADATE borrows an idea from the neutral theory of evolution [5][3]. Instead of simply applying all the possible R transformations directly on a program, it will first generate a number of new programs with the same fitness as the original, and then try to apply R transformations to these new programs. In terms of natural evolution, this roughly corresponds to having a number of individuals of a given species, one of which mutates to a new species with a higher fitness than the original. ADATE uses different neutral transformations to create these new intermediate programs, the most important one is called REQ (Replacement preserving EQuality).

Our aim in this paper is to more efficiently search the space of possible programs resulting from application of REQ transformations. The current version of ADATE systematically tries to apply all combinations of 1, 2, 3 and 4 REQ transformations. This is a very time consuming task, if the program size is N , and k different REQ transformations has been found at each position in the program, the number of combinations of t REQ transformations is $\binom{N}{t}k^t \in O(N^t k^t)$.

To reduce the number of combinations ADATE has to test, and still be able to test all combinations of a given number of REQ transformations, we use so-called *covering arrays*[9]. Covering and orthogonal arrays [4] are standard tools from combinatorial design theory that guarantee coverage of all combinations of a given number of parameters, and were one of the major inspirations when we started this work. Generally, a *covering array with N columns, k levels and strength t* is an N -column array with the property that whichever set of t columns you choose, all the k^t different combinations of “levels”, usually numbers in the range $0..(k-1)$, occur at least once in the selected columns. A simple example with $k = 2$, $N = 3$ and $t = 2$ is given in Fig. 1. A main objective is to make such an array with as few lines as possible. This is a difficult task for the cases $t \geq 4$ but the literature contains numerous results concern-

ing the cases $t \leq 3$ [2]. It is for example quite straightforward to construct a covering array with N columns, k levels and strength two with $O(k^2 \log N)$ lines [8].

If we identify the k levels with the different REQ transformations available per position in our program, and the strength t is the number of REQ transformations we wish to combine, the lines of such a covering array would correspond to a number of programs resulting from a combination of REQ transformations, such that all combinations of up to t REQ transformations are covered. This could significantly reduce the number of programs ADATE has to test. For example, when testing pairs of REQ combinations, the old version of ADATE generates one program for each of the $\binom{N}{2}k^2 \in O(N^2k^2)$ pairs of transformations, whereas using a covering array of strength two, the number of programs could be reduced to $O(k^2 \log N)$.

```
0 0 1
0 1 0
1 0 0
1 1 1
```

Fig. 1.

Unfortunately, as we shall see, it will not be that easy in practice. *Epistatic interactions* between the neutral transformations, as will be described in the next two sections, have turned out to make a complete covering of all possible pairs hard. We have therefore for the time being decided to focus on the $t = 1$ -case, that is, simply, arrays where in any one column all different values occur at least once. Such $t = 1$ -arrays are of course very easy to construct, however they may still give a significant reduction in the number of programs the system has to generate. If testing each allele one at a time, the number of programs created will be $O(Nk)$. A simple covering array of strength one could in theory reduce this to $O(k)$. So using covering arrays of strength one should also be an improvement, especially if the programs are large.

2 Syntactic Epistasis

When viewing programs as lines in an array, we have abstracted away some important structural information about the program. Of course, programs are not simple strings of consecutive expressions that can be transformed independently, the subexpressions of a program are all part of the same expression tree. So a line like “000”, telling us to replace each of the subexpressions by some new expression, will not always make sense.

For example, consider the expression $f(a, b)$, and that the REQ-replacement numbered 0 at the first position is a replacement of the $f(x, y)$ -function with some constant f_0 . If we apply this transformation to our expression, we obviously cannot at the same time apply some transformation on either of the subexpressions a or b , since they have been removed from the expression.

This technical issue, which we call “Syntactic Epistasis”, is not very hard to solve, especially for the case of strength $t = 1$ which we have so far focused our attention on. All we have to do is split the affected lines according to the syntactical structure of our program. In the above case, a line “000” would have to be split into the two lines “0--” and “-00” (where the “-”-symbols denote positions at which no transformation is to be applied).

3 Semantic Epistasis

Our goal here is to combine several neutral transformations into a few large transformations in order to reduce the number of programs ADATE has to search through. There is, however, one large problem we have to consider, which is that we have no guarantee that the programs resulting from these combined transformations are equivalent to those resulting from simple REQs. Even though we are only combining neutral transformations, the resulting transformations may still decrease the fitness of the program. In fact, it turns out that such non-neutrality resulting from neutral transformations is very often the case.

To remove such “Semantic Epistasis” from an array, we can test each of the lines in the array, and split up as soon as we discover a non-neutral line. The most straightforward option is to simply split such lines in two equal halves, and recursively test (and possibly split) each of the halves until all epistatic interactions have been removed. The result will be that the number of lines will grow due to the splitting, but all the lines will be neutral, i.e., there will be no epistatic interaction between the alleles in any line.

Another strategy, which has turned out to be better in the sense that it generates fewer partitions, is to first greedily take as many alleles as possible without destroying neutrality into one partition, take as many of the remaining alleles as possible into a second partition, and so on until all the alleles have been partitioned into a hopefully quite small number of partitions, each of which is neutral with respect to fitness.

Of course if we are working with covering arrays for strength $t > 1$, these simple partitioning strategies cannot be used. If we split a line in two equal halves, we will lose one half of the pairs, one fourth of its triples, and, generally, $\frac{1}{2^{t-1}}$ of its t -tuples, since the original line contains $\binom{N}{t} \approx \frac{N^t}{t!}$ t -tuples, while the two halves together contain only $2\binom{\frac{N}{2}}{t} \approx \frac{1}{2^{t-1}} \frac{N^t}{t!}$ t -tuples. Therefore, our simple halving algorithm cannot be used if we are working with strength $t > 1$. An alternative might be “ $t + 1$ -splitting” (of which halving is a special case): each time a line with epistasis occurs, split it into $t + 1$ (roughly) equal parts, then form $\binom{t+1}{t}$ new lines by combining t of these $t + 1$ parts in all possible ways.

In the case $t = 1$ this is the same as the “halving” strategy described above, and although not the most efficient strategy, it can at least be practically usable, creating in the worst case $O(N)$ new lines for each epistatic line in the array.

However, for higher values of t , the number of lines resulting from such a strategy grows fast. For instance, if $t = 2$, we would have to split any epistatic line into three lines, each of the lines containing $\frac{2N}{3}$ of the alleles in the original line. Then we would have to recursively check, and possibly split, each of these three lines, and so on. Denoting the number of lines resulting from running this strategy on a line with N alleles as a_N , we have in the worst case, i.e., if we have to continue recursively to split as long as possible, $a_N = 3a_{2N/3}$, which gives us $a_N \approx N^{2.7}$ [1], ruining our hopes of reducing the number of programs generated from $O(N^2k^2)$ to $O(k^2 \log N)$. Unless we find some other way of dealing with epistasis, we will therefore not be able to use covering arrays for strengths $t > 1$.

4 Experiments

In order to test our covering array algorithm, we have conducted a series of experiments. One possible way to conduct such experiments would be to do complete runs of the modified ADATE system, and compare the results to similar complete runs of the unmodified ADATE system. However, this approach is unsuitable for the following two reasons:

- Since our covering arrays combine several transformations into one, the resulting programs tend to be quite large and contain much redundancy. The ADATE system, however, tends to prefer smaller programs in order to reduce the risk of overfitting, as described in section 1. A full-scale run of ADATE modified by our new covering arrays strategy might therefore not give us a correct impression of the new strategy. We could couple it with some heuristics to reduce the size of the programs generated, but it would make it more difficult to directly measure the performance of our new CA-transformation.
- During a complete run of ADATE, thousands of programs are created, inserted into the population, transformed and discarded. The results of such a run would be difficult to analyze. It would be better to concentrate our experiments on the aspects of ADATE where our Covering Arrays would be used, in order to more easily analyze the results.

Since our work concern the combination of neutral transformations in order to more efficiently search the local neutral fitness space, we decided to let the experiments concentrate on the local search for transformations for a specific program. We ran the main part of the ADATE system, namely the local search algorithms, from a number of different test programs.

Each experiment started from one given test program, and iteratively increased the cost limit used to transform that program. Each time a new program better than the test program was found, it was noted in a logfile, together with the transformation form used (like REQ REQ R, or CA if the new covering array algorithm was used). Thereafter the system continued to search for transformations from the test program. So there is no real local search, all we did was to search the neighborhoods of the test programs with iteratively increasing cost limits and thus iteratively increasingly greater neighborhoods.

In order to be able to compare the new CA transformations with the old REQ-transformations, the time the system spent doing REQs was split into two: 50% of the CPU time was given to the old REQ algorithm, while the remaining 50% of the CPU time was given to our new CA algorithm.

The experiment was repeated 18 times, using several different specifications, and one, two or three different test programs for each specification. For each of the specifications, the best program found by a run of ADATE, that typically was given one week on a single CPU, was used as a test program. Being the best programs found by ADATE, these programs should be difficult to improve further. Additionally, some random test programs, marked with ' or " in the table, were chosen from old ADATE runs for some of the more interesting specifications. So we had a mix of difficult and average programs for our experiments.

The syntactic complexity of the start programs varied from around 150 to 660 bits, and many of them contained one or more invented help functions.

The specifications we used were:

- **ASMB** - parsing and computing simple arithmetical expressions including parentheses, addition, subtraction and multiplication, following the standard rules of precedence.
- **BOXPACK** - packing as many boxes of different sizes as possible into another box, given the sizes of the boxes as parameters.
- **CHESS1** - discovering as many legal moves as possible given a representation of a chess board as its parameter. The specification contained only training input with one king and/or one rook in the same color.
- **CHESS1b** - the same specification as **CHESS1** but with a higher number of training inputs and a more complex and difficult to improve test program.
- **CHESS2** - the same specification as **CHESS1**, but this time including training inputs where the king and the rook can reach each other, so the function has to check for collisions and disallow jumping.
- **GS** - learning the semantics of a simple language as described in [7].
- **PERMS** - generating all the possible permutations of a given input-list.
- **SAT** - The Satisfiability problem, a fundamental NP-complete problem.
- **ST** - A function that tries to maximize profit when buying and selling stock every day. The training data is based on 10 years of Swedish stock prices.
- **UNSAT** - The Unsatisfiability problem, another fundamental problem in algorithm design and likely to be even more difficult than the SAT problem.

5 The Results of the Experiments

Each of the experiments was run for about a week on a single-CPU 2.8GHz computer. Afterwards, we extracted from the logfiles the transformation forms used each time a new program was constructed that was better than any of the previously constructed programs. The results are plotted as a timeline for each experiment in Fig. 2¹. The solid lines in the timelines represent segments of time in which our new CA transformation was used in the construction of the currently best program, while the dashed lines represent segments of time in which the currently best program was constructed using one or more REQs. At the empty areas of the timelines, no program better than the test program has been found, or the currently best program was created using other transformations.

Note that the graphs do not show how many programs were actually constructed using the different forms. So for example a long solid line does not mean that a lot of good programs were constructed using our new CA transformation, all it means is that for a long period of time, the currently best program was created using CA. In many such cases the system found just one good program, after which it took a very long time before any better program was found with either method, if at all. This simplified visualization will certainly not give

¹ The experiments were given slightly varying amounts of runtime, therefore the timelines in the graph end at varying points (denoted by x's).

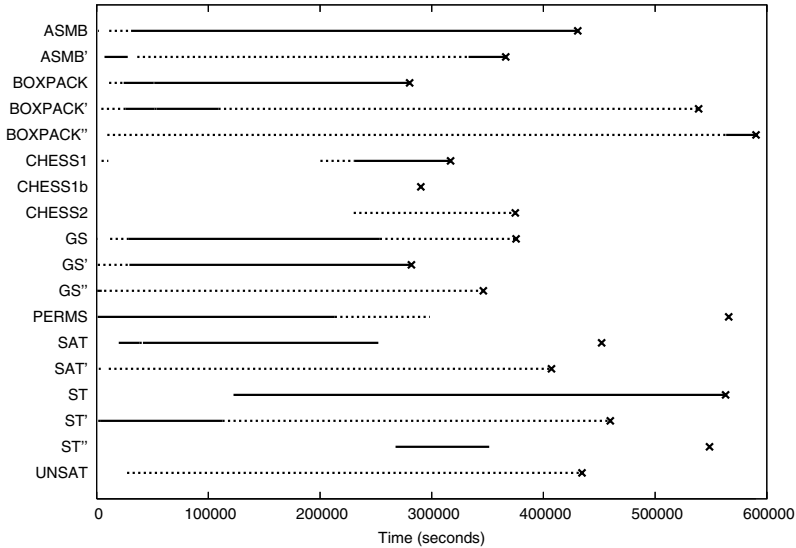


Fig. 2. Timelines of the experiments, representing which transformations that were used to produce the currently best programs at any time

us a complete picture of our results, but it will give us an impression of which one of the methods yielded the best programs at varying points of time.

In 7 of the 18 timelines, our new CA transformation had been used to construct the currently best programs as the experiments were terminated. In another 7, the old REQ transformation was the “currently best” when the experiments ended. In 6 of the 11 experiments in which CA was not involved in creating the best programs at the end of the experiments, CA was at least used to create currently best programs at some earlier points in time, but these were later surpassed by programs constructed without using CA. In the remaining 5 experiments, CA was never involved in creating the currently best programs.

The results did not seem to depend on the size of the test programs. This is rather unexpected, as combinatorially there should be more to gain from combining transformations if the programs are larger, as pointed out in section 1.

During all the experiments, a total of 214 currently best programs were reported. A single REQ transformation was part of the transformation form only in three of these cases, all the other occurrences of REQ transformations were used in conjunction with other REQ transformations. However, a testrun of six of the experiments *without* using the CA transformation showed us that a single REQ transformation appeared to be important in five of the six cases. This may indicate that single REQ transformations are far less important when the CA transformation is included, which is as we expected, since one purpose of our CA transformations is to combine single REQs in as few transformations as possible.

However, combinations of several REQ transformations were used extensively in our experiments. This seems reasonable, since we have restricted ourselves to the case of strength $t = 1$.

6 Conclusions and Future Work

We have introduced a new transformation, CA, into the ADATE system, and showed that it is superior to the old REQ transformations in a number of cases. It virtually removes the need for single REQ-transformations, and can potentially even replace the REQ-transformations altogether. This would result in a significant efficiency gain for the entire ADATE system, especially when synthesizing large programs. As we have seen, the potential gain in efficiency can be very high, for example, in the $t = 2$ case the number of programs to be constructed might be reduced from $O(N^2k^2)$ to $O(k^2 \log N)$.

With this new scalability, the ADATE system should in the near future be able to synthesize much larger programs, maybe upto thousands of lines, far more than anything possible today. With such scalability, the system may be able to create real world commercial applications, or even solve difficult programming problems not yet solved by humans, like for example efficient prime factorization.

References

1. Brassard, G. and Bratley, P. *Fundamentals of Algorithms*. Prentice-Hall, New Jersey, pp. 133 – 134.
2. Chateauneuf, M. and Kreher, D.L. On the state of strength-three covering arrays. *Journal of Combinatorial Designs*, Vol. 10, Nr. 4, pp. 217–238.
3. Geard, N., J. Wiles, J. Hallinan, B. Tonkes and B. Skellett. A comparison of neutral landscapes - NK, NKp and NKq. *Proceedings of the 2002 Congress on Evolutionary Computation*, pp. 205 – 210.
4. Hedayat, A.S., Sloane, N.J.A. and Stufken, J. *Orthogonal Arrays*. Springer, New York.
5. Kimura, M. *The Neutral Theory of Molecular Evolution*. Cambridge University Press, Cambridge.
6. Olsson, R. Inductive functional programming using incremental program transformation. *Artificial intelligence*, Vol. 74, Nr. 1, pp. 55 – 83.
7. Olsson, R. and D.M.W. Powers. Machine learning of human language through automatic programming. *International Conference on Cognitive Science*, University of New South Wales, pp. 507 – 512.
8. Sherwood, G. On the Construction of Orthogonal Arrays and Covering Arrays Using Permutation Groups. *Webpage*, <http://home.att.net/~gsherwood/cover.htm>
9. Sloane, N.J.A. Covering Arrays and Intersecting Codes. *Journal of Combinatorial Designs*, Vol. 1, Nr. 1, pp. 51–63.

Texture Segmentation by Unsupervised Learning and Histogram Analysis Using Boundary Tracing

Woobeom Lee and Wookhyun Kim

Department of Computer Engineering, Yeungnam University,
214-1 Dae-dong, Gyeongsan-si, Gyeongbuk 712-749, Republic of Korea
{beomlee, whkim}@yumail.ac.kr

Abstract. Texture analysis is an important technique in many image processing areas, such as scene segmentation, object recognition, and shape&depth perception. However, most methods are restricted to issue of computational complexity and supervised problems. Accordingly, we propose an efficient method of segmenting texture that uses unsupervised learning schemes to discover a texture cluster without a pre-knowledge. This method applies 2D Gaussian filters to the clustered region iteratively, and the thresholding value for segmenting is automatically determined by analyzing histogram of the clustered inner-region. It can be acquired by the boundary tracing in the clustered region. In order to show the performance of the proposed method, we have attempted to build a various texture images, and the segmenting quality was measured according to the goodness based on the segmented shape of region. Our experimental results showed that the performance of the proposed method is very successful.

1 Introduction

Texture analysis is an important technique in many image processing areas, such as scene segmentation, object recognition, and shape&depth perception, as textures in image which described as fine, coarse, grained, smooth, etc. are a very effective cue for image segmentation. Thus, a numerous approach has been studied in this literature during the past decades[1, 2, 3, 4].

Although such methods are effective only for segmentation, most of these methods require a fair amount of complex computation, plus are restricted to issue of supervised problems requiring a pre-knowledge. Moreover, because they cannot provide a proper information for object recognition in an image, the segmented results is improper in the detecting and retrieval systems that use a query image.

Accordingly, we propose a efficient method of segmenting texture without a pre-knowledge, along with the retrieval of an partial image corresponding to a query image. This paper focuses on reducing the constraint problems by the unsupervised learning schemes, and proposing unsupervised segmentation method using 2D Gaussian filters and boundary tracing in the clustered region. Specially, the segmenting quality is then considered as the shape similarity for a recognizing task.

The proposed method uses a self-organizing neural network for unsupervised block-based texture clustering, and applying 2D Gaussian filters to the clustered block regions for smoothing effects. Also, the thresholding value for segmentation is automatically determined by analyzing histogram of the boundary traced inner-region. Finally, the segmentation is then achieved by applying zero-crossing algorithm to the thresholding image.

2 Unsupervised Learning for Discovering Texture Cluster

The unsupervised clustering for the discovering texture cluster structure from multiple texture image used in our approach is outlined in Fig. 1, and consists of the following four stages: spatial feature extraction, block-based clustering, block-based merging and dilation.

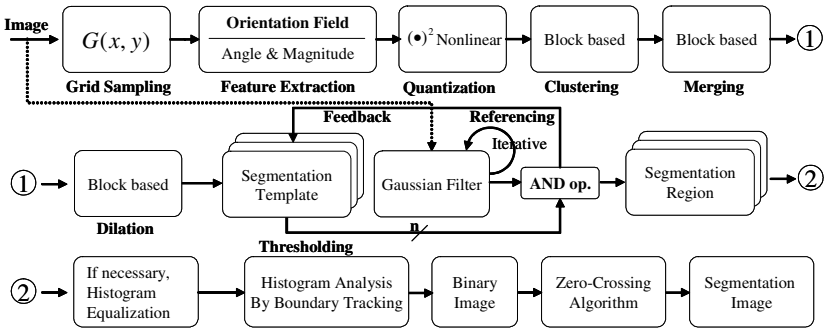


Fig. 1. Unsupervised Texture Image Segmentation processing used in our approach

2.1 Spatial Feature Extraction Based on Average Intensity

The spatial feature extraction in the orientation-field consists of a computationally efficient grid sampling scheme, which expects a higher recognition from low resolution image[5]. Thus, we define the feature function in the spatial domain as (1) respectively for a grid sampling pixel, $G(x, y)$ in an image.

$$F_{OA}(x, y) = \tan^{-1} \left(\frac{dx}{dy} \right), \quad F_{OM}(x, y) = \sqrt{(dx^2 + dy^2)} \quad (1)$$

where $F_{OA}(x, y)$ and $F_{OM}(x, y)$ are the spatial feature values of orientation angle and magnitude respectively. $dx = I(x + 1, y) - I(x - 1, y)$, $dy = I(x, y + 1) - I(x, y - 1)$ is a horizontal and vertical gradient value for any $G(x, y)$ in an image respectively, and $I(x, y)$ is a gray level intensity in $G(x, y)$. Then, if the texture orientation-feature is only considered by the gradient value of 4-neighborhood like form Eq. (1), most of the textures yields the spatial features distinguishably. However, if there is a variation of texture luminance by the light, Texture feature characterized by not so much a orientation-feature, but a brightness of each texture region. Therefore, the spatial feature extraction considering the average intensity is proposed by using the weight factors μ_h, μ_v as follows (2):

$$dx'(x, y) = \mu_h \times dx(x, y), \quad dy'(x, y) = \mu_v \times dy(x, y) \quad (2)$$

where μ_h, μ_v is a horizontal and vertical average gradient value for any $G(x, y)$ in an image respectively. This method is efficient in acquiring distinct orientation-feature for the different intensity(brightness) region where the same gradient value of 4-neighborhood is computed. Consequently it yields orientation-angle and magnitude feature matrices in orientation-field.

2.2 Unsupervised Block-Based Texture Clustering

We propose a novel approach for unsupervised block-based texture clustering that uses self-organizing neural network schemes by Kohonen[6].

To yield the input pattern vectors for self-organizing network, after both $F_{OA}(\cdot)$ and $F_{OM}(\cdot)$ is divided into equally $N \times N$ sized blocks, feature values in the same block $B_{1 \leq k, l \leq N}(x, y)$ of both matrices are quantized by the quantization function, $Q(\cdot)$ and assigned to a count array by $alloc(\cdot)$ function as follows:

$$B_{k,l}(x, y) = alloc(i, j) \quad : \text{ where, } i = Q(F_{OA}(x, y)) \text{ and } j = Q(F_{OM}(x, y)) \quad (3)$$

where k, l is the k -th column and l -th row block index respectively, and $alloc(\cdot)$ is a function that assigns a quantized level to the corresponding count array. This task makes more easy to similarity measures between the texture blocks. After the quantized level is assigned to all points of block, the input pattern vector $x_{k,l}$ for $B_{k,l}$ block may be defined as follows:

$$x_{k,l} = [P_0, P_1, \dots, P_i, \dots, P_n]^T \quad : \text{ where, } P_i = (fm_{i0}, fm_{i1}, \dots, fm_{in}) \quad (4)$$

where $fm_{i,j} = CNT(alloc(i, j))$, and $CNT(\cdot)$ is the accumulative function that corresponds to the accumulative value for the degree i orientation angle and the degree j orientation magnitude in the block $B_{k,l}$. After yielding the input pattern vectors for all block in an image, the winner neuron C_j for unsupervised learning is determined by using the minimum-distance Euclidean criterion like form (5).

$$C_j = \begin{cases} \arg \{ \min \| x_{k,l} - w_j \| \} & \text{if } C_j \in C_{set} \\ \text{assign to new cluster, and append to } C_{set} & \text{otherwise} \end{cases} \quad (5)$$

After the winner neuron C_j is selected, the block-based clustering is performed by using a competitive learning scheme like form (6).

$$w_j(n+1) = \begin{cases} w_j(n) + \alpha[x_{k,l} - w_j(n)] & \text{if } x_{k,l} \in C_j \\ w_j(n) & \text{otherwise} \end{cases} \quad (6)$$

where α is the learning-rate parameter, and each of blocks in an image is then clustered by comparing a similarity of between blocks. Consequently similar blocks are assumed to belong to the same cluster, thus unique labels are assigned

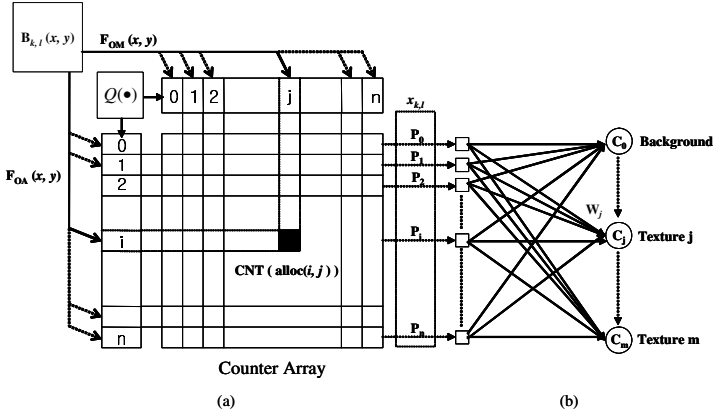


Fig. 2. (a) The input pattern vector $x_{k,l}$ production and architecture for unsupervised learning by a self-organizing neural network in any Block $B_{k,l}$

to each block. However, one textures may be split into several parts, causing a fragmentation problem despite a homogeneous texture region in image. Thus, to overcome this problem, a block-based merging procedure is proposed. If one block is identified as similar to a neighboring block, the same label is assigned and blocks are merged, yielding a number of distinct regions in the image.

3 Histogram Analysis Using Boundary Tracing Analysis from the Clustered Regions

The unsupervised segmentation for extracting the clustered region consists of the following four stages: block-based dilation, 2D Gaussian spatial filtering, block-based boundary tracing and histogram analyzing.

3.1 Block-Based Dilation

As a result of the block-based merging, a number of distinct regions can be acquired from the texture image. However, as merging result do not include enough blocks to segment a texture region of interest, it is needed that the area of blocks are extended before segmentation. To solve this problem is then the block-based dilation operation like form (7)[7].

$$A \oplus B = \{ c \in E^N \mid c = a + b \text{ for some } a \in A \text{ and } b \in B \} \quad (7)$$

where A is the merge map, B is the 4-neighbor elements, and E^N is the set space for the dilation result. If the dilation operation, as defined above, applied to the merged map, the dilation map including enough blocks to segment is yielded.

3.2 2D Gaussian Spatial Filtering

2D Gaussian filters are appropriate for uniforming image containing a very irregular intensity characteristic, as they have a smoothing and blurring effect.

The 2D Gaussian function as a spatial filter is defined as follows (8):

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\left[\frac{(x-x_0)^2}{2\sigma_x^2} + \frac{(y-y_0)^2}{2\sigma_y^2}\right]\right) \quad (8)$$

where (x_0, y_0) specify the means, (σ_x, σ_y) denote the standard deviation of the distribution, and a Gaussian function used in our approach is a circularly symmetric form with mean $(0, 0)$ and $\sigma_x = \sigma_y$.

This filter yields a weighted average of each pixel's neighborhood, with the average weighted more towards the value of the central pixels, a Gaussian distribution provides gentler smoothing and preserves the better edges. Thus, the Gaussian filter is very similar to the optimal smoothing filter for edge detection, and for this reason, it is sometimes useful to apply a Gaussian smoothing filter to an image before performing edge detection[5].

The degree of smoothing effect is determined by the standard deviation of the Gaussian distribution, which is considered as the size of filters. By using a small Gaussian filter and then applying an edge detector, there will also be a lot of unwanted edge fragments appearing due to noise and fine texture. Also, for a larger smoothing filter, there will be fewer unwanted edge fragments, but much of the detail in the edges will also be lost.

Accordingly, the n -th output $O(x, y)$ of a Gaussian filter is defined by

$$O^{n+1}(x, y) = O^n(x, y) + \sum_{\xi=0}^{m-1} \sum_{\eta=0}^{m-1} \left(g^n(\xi, \eta) \cdot O^n(x + \xi, y + \eta) \right) \quad (9)$$

where $O^0(\cdot)$ is an original image, (ξ, η) denote a area of the n -th Gaussian filtering, and m specifies a size of the n -th Gaussian filter. Our method applies the effect of smoothing with successively larger and larger Gaussian filters to an image. As a result, the filtering effect results in essentially uniform responses in the clustered regions.

3.3 Histogram Analyzing Using Block-Based Boundary Tracing

Applying a Gaussian filter successively to the original image while varying with the size of filter, each of the distinct texture region in an image has a individually uniform filtering response. Here, problem for segmenting is to know a range of the uniform response in an image. With respect to this problem, only the texture region containing the uniform response should be first extracted from an image. This task is performed by using the image AND-op. between the Gaussian filtered image and the dilation results, i.e., the segmentation templates in Fig. 4 (f).

When one of the clustered regions was acquired like Fig. 3(a), the block-based boundary tracing is required to analyze a histogram for the merely interior region

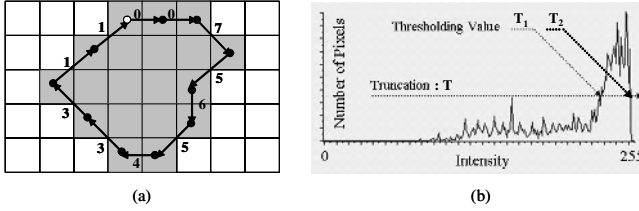


Fig. 3. (a) Block-based boundary tracing, and (b) Determining the thresholding value by analyzing histogram

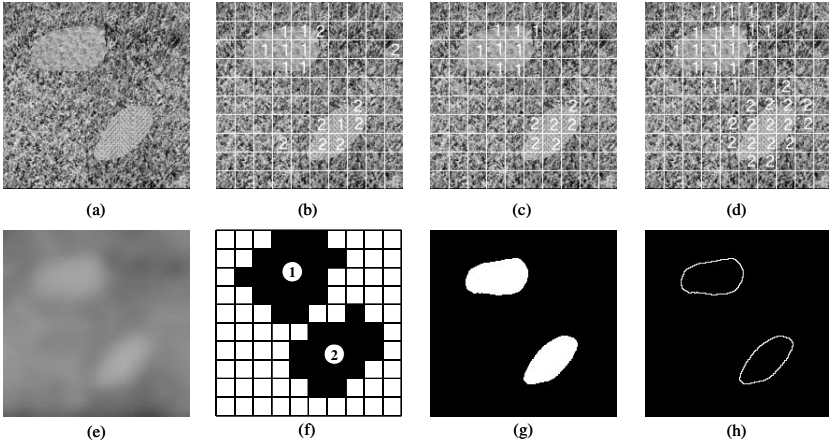


Fig. 4. Experimental Result : (a) Collage of the Brodatz textures D9(Grass : background), D92(Pigskin), D15(Herringbone weave), (b) and (c) depict the clustering and merging result by self-organizing neural network. where the size of original image is 200 x 200 pixel, and block unit is scaled to a size of 20x20 pixel, (d) depicts the dilation result. (e) shows an successive Gaussian filtering image, (f) depicts the segmentation templates of texture cluster 1 and 2 respectively. And (g) depicts the thresholding image by analyzing histogram of boundary tracing, (h) Segmentation image by zero-crossing.

of the boundary. This method is based on the 8-connectivity of the blocks, and coded by using a numbering scheme. As the boundary is traced, a boundary point is then assigned to each center pixel of the blocks, as shown in Fig. 3(a). After a completion of the block-based boundary tracing, the histogram of region inside boundary can be calculated to determine an thresholding value for segmentation.

3.4 Segmentation

As a result shown in Fig. 3(b), because the traced inner-region includes the almost segmenting region and a little of boundary region, the histogram has grouped into dominant and non-dominant range. The segmentation is then completed by extracting a dominant region. In order to segment a dominant region, it is necessary to truncate a non-dominant range. Thus, a truncation value T is defined as follows:

$$T = \omega \bullet P_t \quad (10)$$

where P_i is the total number of pixels in histogram, and as ω is a weight factor, considering as $\omega = 0.1, 0.2, \dots$. If the histogram is truncated by T , the bounding intensity value of remaining range is determined as the upper and lower bound for a thresholding processing. Then, a thresholded binary image $b(x, y)$ is defined as below:

$$b(x, y) = 1 \quad \text{if } T_1 \leq f(x, y) \leq T_2, \quad \text{otherwise } 0 \quad (11)$$

where T_1 and T_2 are the upper and lower bound, respectively, of the truncated histogram, and $f(x, y)$ is the grey level intensity of point (x, y) .

Finally, the final segmentation result is achieved by using efficient edge detection algorithm[5] applying to the thresholded binary image of Eq. (11). The approach by D. Marr is applied for the edge detection algorithm, as follow:

$$\nabla^2 g(x, y) = \frac{1}{2\pi\sigma_e^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_e^2}\right) - A \frac{1}{2\pi\sigma_i^2} \exp\left(-\frac{x^2 + y^2}{2\sigma_i^2}\right) \quad (12)$$

where σ_e and σ_i represent the space constants of excitatory and inhibitory regions, respectively, and the ratio of space constants $\sigma_e/\sigma_i = 1.6$. The ratio yields a good approximation to the ideal Laplacian operator[5]. The segmentation is accomplished by finding the zero crossing of a spatial filter, Eq. (12) applied to the binary image.

4 Experimental Results

In order to evaluate the segmentation quality of the proposed approach, we have attempted to build a various more than 100 texture images from Brodaz texture book[8]. The segmentation quality was measured by using two shape similarity. One is the ratio of the area to the perimeter of a segmented region. The other is the ratio the width of the minor axis to the length of the major axis. Therefore, the shape measure function of a segmented region, SM can be defined as follows:

$$SM = \left(\frac{4\pi A}{P^2}\right) + \left(\frac{MaAL}{MiAW}\right) \quad (13)$$

where A is the total number of pixel inside the segmenting boundary, and P is the pixel distance around the circumference of the segmenting boundary. Also, $MaAL$ is the pixel distance length between the major axis endpoints, and $MiAW$ is the pixel distance length between the minor axis endpoints. As a results, the shape similarity between the ground truth and the segmented results were measured at the over to 90% in spite of varying a ω factor.

This means that the proposed method guarantees the high quality in the shape similarity despite reducing the constraint problems. Therefore, our method has enough to detect and retrieve a partial image corresponding to a query image.

5 Conclusions

We proposed a novel approach for efficient segmenting multiple texture images that uses unsupervised learning schemes and histogram analysis using boundary

tracing. The focus of our approach is to reduce the constraint problems by the unsupervised learning schemes, to provide a proper information for retrieval of a partial image corresponding to a query image, and to improve the segmenting quality based on a shape goodness for recognition. However, some problems, such as sampling, quantization, and the size of an 2D spatial filter, are left for future works. Nonetheless, the proposed method demonstrated a very successfully results in respect of the segmented shape similarity. Therefore, our approach has enough to develop the user image query system in the future.

References

1. T. Randen and J. Husoy, Filtering for texture classification: A Comparative study, *IEEE Trans. PAMI* **21** (1999) 291-310
2. M. Idrissa and M. Acheroy, Texture classification using Gabor filters, *Pattern Recognition Letters* **23** (2002) 1095-1102
3. D. Tsai, etc., Optimal Gabor filter design for texture segmentation using stochastic optimization, *Image and Vision Computing* **19** (2001) 299-316
4. S. Grigorescu, etc., Comparasion of Texture Feature Based on Gabor Filters, *IEEE Trans. Image Precessing* **11** (2002) 1160-1167
5. D. Marr, *VisionA Computational Investigation into the Human Representation and Processing of Visual Information* W. H. Freeman & Company (1982)
6. T. Kohonen, The self-organizing map, *Proc. IEEE* **78** (1990) 1464-1480
7. M. Sonka, etc., *Image Processing, Analysis, and Machine Vision* 2th, PWS (1999)
8. P. Brodatz, *Texture A Photographic Album for Artists and Designer*, Dover (1966)

An Improved Bayesian Network Learning Algorithm Based on Dependency Analysis

Fengzhan Tian, Shengfeng Tian, Jian Yu, and Houkuan Huang

School of Computer & Information Technology,
Beijing Jiaotong University, Beijing 100044, China
{fztian, sftian, jianyu, hkhuang}@center.njtu.edu.cn

Abstract. Generally speaking, dependency analysis based Bayesian network learning algorithms are of higher efficiency. J. Cheng's algorithm is a representative of this kinds of algorithms, while its efficiency could be improved further. This paper presents an efficient Bayesian network learning algorithm, which is an improvement to J. Cheng's algorithm that uses Mutual Information (MI) and Conditional Mutual Information (CMI) as Conditional Independence (CI) tests. Through redefining the equations for calculating MI and CMI, our algorithm could decrease a large number of basic operations such as logarithms, divisions etc. and reduce the times of access to datasets to the minimum. Moreover, to efficiently calculate CMI, an efficient method for finding an approximate minimum cut-set is proposed in our algorithm. Experimental results show that under the same accuracy, our algorithm is much more efficient than J. Cheng's algorithm.

1 Introduction

A BN is a Directed Acyclic Graph (DAG) with a conditional probability distribution for each node. The DAG structures of such networks contain nodes representing domain variables, and arcs between nodes representing probabilistic dependencies. In the past decade, the hardship to elicit BNs from expert knowledge makes researchers turn to learning BNs from data.

The existing algorithms for learning Bayesian networks assumably fall into two categories: Search & Scoring based algorithms [2, 6, 7, 8, 11] and Dependency Analysis based algorithms [4, 5, 12]. The typical process of Search & Scoring based algorithms can be stated as follows. Given a training set, searches the space of network structure candidates according to some search strategy, evaluates each candidate using a scoring function with respect to the training set, finds out the network that best matches the training set (i.e., gives out highest score with respect to the training set). The two commonly used scoring functions are the BDe (Bayesian Dirichlet) metric [7] and the one based on MDL (Minimal Description Length) principle [8]. The size of the space of network structure candidates is exponential on n , the number of domain variables. Despite employing heuristic search strategies, generally speaking, the Search & Scoring based algorithms are of low efficiency due to search of the above huge space.

While the dependency analysis based algorithms evaluate dependencies between variables by Conditional Independence (CI) tests to infer BN structures. Generally speaking, they are more efficient than search & scoring algorithms owing to no need of searching the space of BN structures. It is the case especially for the sparse networks. The algorithm by J. Cheng [3, 4] is the representative of this kind of algorithms, which employs Mutual Information (MI) and Conditional Mutual Information (CMI) to do CI tests. Compared with other dependency analysis based algorithms, J. Cheng’s algorithm enjoys many advantages, which decreases the times of CI tests to $O(n^2)$.

While J. Cheng’s algorithm could be improved further in terms of efficiency. In this paper we presents an efficient Bayesian network learning algorithm, which is an improvement to J. Cheng’s algorithm. First, our algorithm redefines the equations for calculating MI and CMI to decrease the basic operations such as logarithms, divisions etc. and needs only one dataset scan to accomplish all the calculations. Furthermore, to efficiently calculate CMI, our algorithm proposes an efficient method for finding an approximate minimum cut-set. Finally, experimental analysis on Alarm network and Hailfinder network are conducted with respect to J. Cheng’s algorithm and our algorithm.

2 Backgrounds

We assume readers are familiar with the basics about BNs and we will not give unnecessary details here for the sake of space. Let us first review the concept of *d-separation* originally defined in [10], which plays a key role in dependency analysis based algorithms.

Definition 1. d-separation: *Given a DAG G , an adjacency path Ap from node X to node Y is said to be **blocked** by the node set \mathbf{C} , if there is a node $W \in Ap$ such that, either*

1. $W \in \mathbf{C}$ and arrows of Ap does not meet head to head at W , or
2. Neither $W \notin \mathbf{C}$ nor W has any descendants in \mathbf{C} , and the arrows of Ap do meet head to head at W .

*Two nodes, X and Y , are said to be **d-separated** by \mathbf{C} , if all adjacency paths between node X and Y are blocked by \mathbf{C} . \mathbf{C} is called a cut-set or a *d-separating set*.*

An adjacency path is a path between two nodes without considering the directionality of the arcs.

It can be proven that the concept of *d-separation* can reveal all the conditional independence relations that are encoded in a BN. So it can be used to infer the structures of BNs. From an information theoretic point of view, a BN can be viewed as a network system of information channels, where each node is a valve that is either active or inactive and the valves are connected by noisy information channels. The information flow can pass an active valve but not an inactive one. The amount of information flow between two nodes can be measured with MI when no nodes are instantiated, or CMI when some other nodes are instantiated.

Following the above principle, J. Cheng’s Algorithm [3, 4] first calculates MI and CMI, and on this basis, does CI tests and constructs a BN. His Algorithm has three phases: **drafting**, **thickening** and **thinning**. Given the node ordering, the process of J. Cheng’s algorithm is as follows:

Phase I: Drafting

1. Initiate a graph $G = (V, E)$, where $V = \{\text{all the nodes where each corresponds an attribute of a dataset}\}$, $E = \{\}$. Initiate an empty list L .
2. For each pair of nodes (V_i, V_j) , where $V_i, V_j \in V$, calculate MI $I(V_i, V_j)$ with equation (1). For all the pairs of nodes that have MI greater than a certain small value ε , sort them by their MI and put these pairs of nodes into list L from large to small.
3. Create a pointer P that points to the first pair of nodes in L .
4. Get the pair of nodes from L at the position of the pointer P . If there is no adjacency path between the two nodes, add the corresponding edge to E and remove this pair of nodes from L .
5. Move the pointer P to the next pair of nodes and go back to Step 4 unless P is pointing to the end of L or G contains $n - 1$ edges. (n is the number of nodes in G .)

Phase II: Thickening

6. Move the pointer P to the first pair of node in L .
7. If P is pointing to the end of L , then go to Step 10; otherwise, go to next step.
8. Get the pair of nodes (V_i, V_j) from L at the position of the pointer P . Call Procedure `find_cut_set` (G, V_i, V_j) to find a small cut-set \mathbf{C} that can d-separate V_i and V_j in current graph G . Calculate CMI $I(V_i, V_j|\mathbf{C})$ with equation (2). If the result is greater than ε , connect V_i and V_j by adding a corresponding arc to E ; otherwise, go to next step.
9. Move the pointer P to the next pair of nodes and go back to Step 7.

Phase III: Thinning

10. For each edge (V_i, V_j) in E , if there are other paths besides this edge between the two nodes, remove this edge from E temporarily and get graph G' . Call Procedure `find_cut_set` (G', V_i, V_j) to find a small cut-set \mathbf{C} that can d-separate V_i and V_j in graph G' . Calculate CMI $I(V_i, V_j|\mathbf{C})$ to check if the two nodes are conditionally independent given \mathbf{C} . If so, remove the edge permanently; otherwise add this edge back to E .

In the above process, MI and CMI between two nodes is calculated as follows:

$$I(X, Y) = \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)} \tag{1}$$

$$I(X, Y|\mathbf{C}) = \sum_{x,y,\mathbf{c}} P(x, y, \mathbf{c}) \log \frac{P(x, y|\mathbf{c})}{P(x|\mathbf{c})P(y|\mathbf{c})} \tag{2}$$

Where \mathbf{C} is a node set. When $I(X, Y)$ is smaller than a certain threshold ε , it is said that X and Y are marginally independent; When $I(X, Y|\mathbf{C})$ is smaller than ε , it is said that X and Y are conditionally independent given \mathbf{C} . The Procedure `find_cut_set` in the above process employs a heuristic strategy, which requires an exponential number of basic operations for finding the paths between two nodes.

3 An Improved BN Learning Algorithm

According to the time complexity analysis by J. Cheng [3], the computational consumption is mainly on calculations of MI and CMI. While calculations of MI and CMI include two parts: statistical operations that compute the probabilities from datasets and basic operations such as logarithms, divisions and multiplications etc. This enlightens us to improve J. Cheng’s algorithm from two aspects: optimizing the statistical operations and decreasing the basic operations. Following this idea, we redefine the equations for calculating MI and CMI as follows:

$$I(X, Y) = \alpha \sum_{x, y} N(x, y) \log \frac{N(x, y)}{N(x)N(y)} + \beta \quad (3)$$

$$I(X, Y|\mathbf{C}) = \alpha \sum_{x, y, \mathbf{c}} N(x, y, \mathbf{c}) \log \frac{N(x, y, \mathbf{c})N(\mathbf{c})}{N(x, \mathbf{c})N(y, \mathbf{c})} \quad (4)$$

Where $\alpha = 1/N(*)$ and $\beta = \log N(*)$. $N(\mathbf{x})$ is the frequency of cases with $\mathbf{X} = \mathbf{x}$ in the dataset, for example, $N(x, y, \mathbf{c})$ is the frequency of cases with $X = x$, $Y = y$ and $\mathbf{C} = \mathbf{c}$ in the dataset; $N(*)$ is the number of all the cases in the dataset.

α and β are two constants related to the size of the dataset, so their computing cost can be ignored. When we calculate MI using Equation (3) instead of Equation (1), each CI test can save $O(|X| * |Y|)$ division operations; And when we calculate CMI using Equation (4) instead of Equation (2), each CI test can save $O(|X| * |Y| * |\mathbf{C}|)$ division operations. $|X|$ and $|Y|$ represent the cardinality of variable X and Y respectively. $|\mathbf{C}|$ represents the cardinality of cut-set \mathbf{C} , which can be computed as follows. Assume $\mathbf{C} = \{C_1, C_2, \dots, C_k\}$, then $|\mathbf{C}| = \prod_{i=1}^k |C_i|$. Because many CI tests have to be done, a very large number of basic operations can be saved through the above redefinition.

Calculating MI $I(X, Y)$ needs to count $N(x)$, $N(y)$ and $N(x, y)$. Utilizing the standard SQL statements, we can accomplish all the statistical computation through one access to the dataset. For example, with the statement ”**Select** X , Y , **Count(*) From** *Table Name* **Group By** X, Y ”, we can group all the cases that have the same values for variable X and Y into main memory. And all the needed sufficient statistics can be obtained through one scan of the cases in main memory. Because the number of the cases is $|X| * |Y|$ at most, which is very small in general, the statistical computation can be done wholly in main memory, the time cost can be ignored.

However, calculation of CMI is totally different from that of MI. This is because the size of the dataset retrieved is exponential on $|\mathbf{C}|$, when $|\mathbf{C}|$ is too large,

the time cost of transferring the dataset into and handling it in main memory can not be ignored. This will result in very expensive calculation of the CI tests. Moreover, if $|\mathbf{C}|$ is too large, the CI tests can be unreliable when the database is not large enough [4]. So how to find a small cut-set is a critical problem for the dependency analysis based algorithms.

The ideal situation is to find the minimum cut-set. But to find the minimum cut-set is a combinatorial optimization problem, which is also computationally expensive especially when the network is not a sparse one. In [1], an algorithm for finding a minimum cut-set is presented, which guarantees that a minimum cut-set between two nodes X and Y can always be found. But that algorithm is quite complex. Although J. Cheng's procedure of finding a cut-set employs a heuristic strategy and is simpler than the one presented in [1], it is still quite complex because of the using of the subtle criterion: d -separation.

It is a good compromise to find an approximate minimum cut-set rather than an exact one with high efficiency. Next we will focus on how to efficiently find an approximate minimum cut-set. In [1], Acid proved the following theorem:

Theorem 1. *The problem of finding a minimum d -separating set (cut-set) for X and Y in a DAG G is equivalent to the problem of finding a minimum separating set for X and Y in the undirected graph $(G_{An(X,Y)})^m$.*

Where $(G_{An(X,Y)})^m$ represents the moral graph of $G_{An(X,Y)}$, which is a subgraph of DAG G and composed of node set $An(X, Y)$. $An(\mathbf{X})$ can be defined as follows.

$$An(\mathbf{X}) = \mathbf{X} \cup \bigcup_{X \in \mathbf{X}} an(X) \tag{5}$$

Where $an(X)$ is the set of ancestors of X . The moral graph of a DAG can be obtained from the original DAG by "marrying" parents with a common child and then dropping directions on arrows [9].

Based on **Theorem 1**, we can simplify the problem of finding a minimum cut-set in a DAG to a simpler one finding a minimum separating set in an undirected graph. Following this idea, we present a concise and efficient procedure for finding an approximate minimum cut-set shown in the following.

```

Procedure Find cut-set (Current Graph  $G$ , Node  $X$  and  $Y$ )
  Begin
  Initiate an empty cut-set  $\mathbf{C}$ .
  Calculate  $An(X, Y)$  from  $G$  by Equation (5) and construct  $G_{An(X,Y)}$ .
  Construct the undirected graph  $(G_{An(X,Y)})^m$  denoted by  $U$ .
  While there are chains between  $X$  and  $Y$  in  $U$  Do
    Begin
    Find a node in  $U$  that have maximum number of adjacent edges.
    Remove the node and its adjacent edges from  $U$ .
    Put the node into cut-set  $\mathbf{C}$ .
    End
  Return cut-set  $\mathbf{C}$ 
End

```


It first constructs the moral graph $(G_{An(X,Y)})^m$ from G . Then it employs the following heuristic search to find an approximate minimum separating set. Repeatedly find a node that can intercept the maximum number of chains between X and Y in the undirected graph U and remove it, until X and Y are disconnected. All the removed nodes in the procedure will make up the needed cut-set. Although this procedure uses greedy search, it can find a minimum cut-set in most cases. For example, in the experiments on Alarm network, all the cut-sets found by this procedure are exactly minimum.

When the moral graph becomes too complex, e.g., where there are hundreds or thousands of chains between a pair of nodes, for the sake of efficiency, we use the node set that contains the parents of Y that are on the paths between X and Y as the cut-set (suppose that Y is not an ancestor of X). Although the cut-set found in this way may be larger than the cut-set found by the above procedure, it is also acceptable as long as the dataset is large enough.

4 Experiments

For the convenience of comparison, we implement J. Cheng's algorithm and our algorithm in a Java based system. The criterion for evaluating the accuracy of these two algorithms is to make comparison between the numbers of missing edges and extra edges of the learned networks from the experimental datasets by the two algorithms contrasted with the true networks. We also compare our algorithm with J. Cheng's algorithm through experiments in terms of efficiency.

In our experiments, we consider two typical BNs: Alarm network having 37 nodes and 46 arcs and Hailfinder network having 56 nodes and 66 arcs. For each network, using Netica System (which is available at <http://www.norsys.com>), we generated random samples with 10,000, 20,000, 40,000, 60,000, 80,000 and 100,000 cases respectively. In the experiments, the threshold of CI tests in the two algorithms is set as 0.005. All the experiments were conducted on a Pentium 2.4 GHz PC with 512MB of RAM running under Windows 2000. The datasets are stored in an Oracle database that can be accessed with JDBC. The experimental outcome is the average of the results running 10 times for each condition given the variable order.

Table 1 shows the experimental results of J. Cheng's algorithm and our algorithm running on Alarm network. Where the "A+B" in the table represents there are A missing edges and B extra edges in the networks learned from the datasets by the two algorithms compared with the true network. From the table, we can see that the experimental results of two algorithms on different sample size in terms of learning accuracy are totally same, which have same missing edges and extra edges. The general trend for the two algorithms is that the learned networks by two algorithms are more and more accurate as the size of the datasets gradually enlarge. When the size of the dataset reaches 60,000, both of the two algorithms can learn exactly the true network. The reasons why the two algorithms have same accuracy are as follows. On one hand, the redefined equations for calculating MI and CMI are equivalent to the original ones. On

Table 1. Experimental results of the two algorithms on Alarm Network

Algorithms		Sample Size					
		10,000	20,000	40,000	60,000	80,000	100,000
Missing and Extra edges	Cheng's algorithm	1+0	0+1	0+1	0+0	0+0	0+0
	Our algorithm	1+0	0+1	0+1	0+0	0+0	0+0
Running time (S)	Cheng's algorithm	134	252	455	592	793	956
	Our algorithm	31	60	117	159	212	253

Table 2. Experimental results of the two algorithms on Hailfinder Network

Algorithms		Sample Size					
		10,000	20,000	40,000	60,000	80,000	100,000
Missing and Extra edges	Cheng's algorithm	3+1	3+0	2+0	1+0	1+0	0+0
	Our algorithm	3+1	3+0	2+0	1+0	1+0	0+0
Running time (S)	Cheng's algorithm	201	371	685	968	1290	1585
	Our algorithm	38	72	143	208	269	326

the other hand, although the cut-sets found by two algorithms may be different in size, the results of CI tests based on them are the same.

Table 1 also illustrates the relationship between running time of the two algorithms and different sample size. We can see from Table 1 that the running time of the two algorithms is roughly linear to the size of the datasets and the slopes of the running time of the two algorithms are generally the same. However, the running time of our algorithm is much less than that of J. Cheng's algorithm on all samples, which is about only one quarter of that of J. Cheng's algorithm. This proves that the improvement to J. Cheng's algorithm is very effective.

Table 2 shows the experimental results of the two algorithms running on Hailfinder network. Similar to the experimental results in Table 1, the two algorithms have totally same learning accuracy on all samples, roughly linear running time and about same slopes of running time. As same as in Table 1, the running time of our algorithms is much less than that of J. Cheng's algorithm on all samples, while it is only about one fifth of that of J. Cheng's algorithm. This gives us confidence that our algorithm could save more running time as learning more complex networks.

5 Conclusions

This paper presents an efficient Bayesian network learning algorithm, which is an improvement to Jie Cheng's algorithm that uses MI and CMI as CI tests. Our algorithm redefines the equations for calculating MI and CMI, which can not

only decrease a large number of basic operations, but reduce the times of access to dataset to the minimum. Moreover, to efficiently calculate CMI, an efficient method is proposed in our algorithm to find an approximate minimum cut-set. The experimental results on Alarm network and Hailfinder network show that under the same accuracy, our algorithm is much more efficient than J. Cheng's algorithm.

Acknowledgments

This work is supported by NSF of China under Grant No. 60442002 and Grant No. 60303014. Thank anonymous readers for helpful comments.

References

1. Acid S., Campos L.M.: An algorithm for finding minimum d-Separating sets in belief networks. Proceedings of the twelfth Conference of Uncertainty in Artificial Intelligence (1996)
2. Buntine, W.: A guide to the literature on learning probabilistic networks from data. IEEE Transactions on Knowledge and Data Engineering **8** (1996) 195-210
3. Cheng, J., Bell, D.A., Liu, W.: Learning belief networks from data: An information theory based approach. Proceeding of the sixth ACM International Conference on Information and Knowledge Management (1997)
4. Cheng, J., Greiner, R., Kelly, J., Bell D.A., Liu, W.: Learning Bayesian Networks from Data: an Information-Theory Based Approach. The Artificial Intelligence Journal **137** (2002) 43-90
5. Dash, D., Druzdzel, M.J.: Robust Independence Testing for Constraint-Based Learning of Causal Structure. UAI 2003 (2003) 167-174
6. Friedman, N.: The Bayesian structural EM algorithm. Fourteenth Conf. on Uncertainty in Artificial Intelligence (1998)
7. Heckerman, D., Geiger, D., Chickering, D.: Learning Bayesian networks: The combination of knowledge and statistical data. Machine Learning **20** (1995) 197-243
8. Lam, W., Bacchus, F.: Learning Bayesian belief networks: An approach based on the MDL principle. Computational Intelligence **10** (1994) 269-293
9. Madsen, A.L., Jensen, F.V.: Lazy propagation: a junction tree inference algorithm based on lazy evaluation. Artificial Intelligence **113** (1999) 203-245
10. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, Inc., San Mateo, CA (1988)
11. Tian, F., Lu, Y., Shi, C.: Learning bayesian networks with hidden variables using the combination of em and evolutionary algorithm. Proceedings of the 5th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Heidelberg: Springer-Verlag (2001) 568-574
12. Tian, F., Lu, Y., Shi, C.: Learning bayesian networks from incomplete data based on EMI method. Preceedings of ICDM 2003, Melbourne, Florida, USA (2003) 323-330

Mining Common Patterns on Graphs

Ivan Olmos¹, Jesus A. Gonzalez¹, and Mauricio Osorio²

¹Instituto Nacional de Astrofísica, Óptica y Electrónica,
Luis Enrique Erro #1, Sta. Maria Tonantzintla; Puebla, México
{iolmos, agonzalez}@inaoep.mx

²Universidad de las Américas Puebla, Sta. Catarina Mártir, Puebla, México
josorio@mail.udlap.mx

Abstract. Finding common patterns is an important problem for several computer science subfields such as Machine Learning (ML) and Data Mining (DM). When we use graph-based representations, we need the Subgraph Isomorphism (SI) operation for finding common patterns. In this research we present a new approach to find a SI using a list code based representation without candidate generation. We implement a step by step expansion model with a width-depth search. The proposed approach is suitable to work with labeled and unlabeled graphs, with directed and undirected edges. Our experiments show a promising method to be used with scalable graph matching.

1 Introduction

Graphs are a powerful and flexible knowledge representation used to model simple and complex structured domains. The representation power and flexibility is the reason of why this representation model has been adopted by researchers in different areas such as ML and DM [1, 5], where an important problem is to find similarities between objects. If we use a graph-based representation (GBR), the problem turns into finding similarities between graphs, which includes SI detection [1, 2, 3, 4 and 5]. The core of the SI problem consists on finding the association (or mapping) between the vertices and edges of the graphs to match. Since the SI Problem is known to be in NP-C [6], in the worst case the time to detect a SI is exponential, unless $P = NP$.

Several works have been developed to find these mappings, using different ideas and representations to speedup the process. For example, Subdue [1] implements a computationally-constrained beam search (the algorithm may not always find a match when it does exist). Some algorithms impose topological restrictions to the input graphs such as trees [7]. There are other algorithms such as AGM [2], FSG [5], and gSpan [3], where completeness is not sacrificed.

In this research we present a new algorithm to detect SI instances in simple directed or undirected graphs. We used a list code based representation, where each code is sorted using criteria such as degrees, statistical summaries of the codes and a label order. The sequences of codes are built with a step by step expansion without candidate's generation and pruning phases that aims to reduce the number of operations to perform. This paper extends our work presented in [11]: we added a new theoretical analysis, the capability to work with directed graphs and more experiments

that show the algorithm capability. In Section 2 we introduce some concepts and notation. In Section 3 we present our graph-based representation. In Section 4 we describe the algorithm. In Section 5 we show a theoretical analysis and in section 6 and 7 our experiments and results. In Section 7 we show our conclusions.

2 Definitions and Notations

In this work we assume that a *graph* G is a 6-tuple $G = (V, E, L_V, L_E, \alpha, \beta)$, where: $V = \{v_i \mid i = 1, \dots, n\}$, is the finite set of vertices, $V \neq \emptyset$; $E \subseteq V \times V$, is the finite set of edges, $E = \{e = \{v_i, v_j\} \mid v_i, v_j \in V\}$; L_V , is a set of vertex labels; L_E , is a set of edge labels; $\alpha: V \rightarrow L_V$ is a function assigning labels to the vertices; and $\beta: E \rightarrow L_A$ a function assigning labels to the edges.

Let G be a graph, where $G = (V, E, L_V, L_E, \alpha, \beta)$. A *subgraph* S of G , denoted by $S \subseteq G$, $S = (V^S, E^S, L_V^S, L_E^S, \alpha^S, \beta^S)$ is a graph such that $V^S \subseteq V$, $E^S \subseteq E$, $L_V^S \subseteq L_V$, $L_E^S \subseteq L_E$.

Given two graphs $G' = (V', E', L'_V, L'_E, \alpha', \beta')$, $G = (V, E, L_V, L_E, \alpha, \beta)$, G' is *isomorphic* to G , $G' \cong G$, if there exists $f: V' \rightarrow V$ and $g: E' \rightarrow E$ as bijections, where : $\forall v' \in V', \alpha'(v') = \alpha(f(v'))$ and $\forall \{v'_i, v'_j\} \in E', \beta'(\{v'_i, v'_j\}) = \beta(g(\{v'_i, v'_j\}))$.

Let G' and G be two graphs, then G' is a SI of G if $\exists S \subseteq G$ such that $G' \cong S$.

3 Subgraph Isomorphism Based on List of Codes

We developed an algorithm to detect the instances of a graph G' in a graph G (SI problem) using a list code based representation. A code represents the information of an edge label and its adjacent vertices. Each code is sorted using criteria such as vertices degrees, statistical summaries of the codes and a label order. The sequences of codes are built with a step by step expansion without candidate's generation. Also, we implement a width-depth search algorithm with a pruning phase that aims to reduce the number of operations to perform. This idea is shown in Fig. 1.

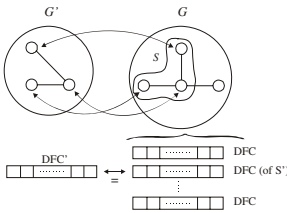


Fig. 1. Finding a Subgraph Isomorphism Based in Lists of Codes

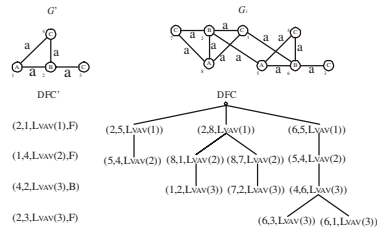


Fig. 2. Example of a Width-DFS using L_{VEV} Codes

Our method starts by building a model of G' , represented by a linear sequence of codes, called DFC' . The DFC' model is a sorted sequence $\langle dfc'_1, \dots, dfc'_n \rangle$, where each entry $dfc'_x = (i, j, c_k, t)$ and i, j , are the indexes associated to the adjacent vertices of edge $e_x \in E'$, c_k is a code that represents the label's information of the edge e_x , $c_k = L_{VEV}(e_x)$ (the L_{VEV} list represents the label's combination of L_V and L_E) and t is a mark

to classify the edges as forward or backward. We use three criteria to build the DFC' sequence: a) the labels combination; b) the most repeated label, and c) the vertices degree. These concepts are used in combination with a DFS strategy to build DFC' .

Then, we perform the mapping process constrained to find a DFC code where $|DFC'|=|DFC|$ and for every entry dfc'_x in DFC' and dfc_x in DFC , the L_{VEV} fields are the same. First, we apply a pruning phase, where $\forall v \in V: \alpha(v) \notin L'_V$ and $\forall e \in E: \beta(e) \notin L'_E$ are removed. This task can be performed with L_{VEV} , removing $\forall e \in G$ where $L_{VEV}(e) \notin L_{VEV}$. Moreover, we can remove edges in G without the minimum number of repetitions. We also use the vertices degrees and their number of repetitions in the graph to further prune G .

Once the pruning phase is performed, G is explored with a width-depth search. We use a backtracking algorithm, because this technique is fairly stable and performs well in most cases. Since DFC' is an array, the backtracking strategy can be used.

Fig. 2 shows an example based on the L_{VEV} codes and a width-depth search strategy. In this example, we can see how the partial result vectors are generated and pruned. The algorithm can find all the instances of G' in G (by keeping it running).

4 The Algorithm

The pseudocode of our concepts is shown in Fig 3 and Fig 4. First the L_{VEV} and DFC' codes are generated in steps 1 and 2, based on the concepts described in Section 3. Clearly, the cost to generate L_{VEV} is bounded by $O(\min\{|E'|, |L'_V|^2 * |L'_E|\})$ and the cost to build DFC' is related to the number of edges, so it is bounded by $O(|E'|)$. Using the L_{VEV} code G is pruned, step 3 of Fig. 3. The computational cost of this process is bounded by $O(\max\{|V|*|L'_V|, |E|(|L'_V|^2, \min\{|E'|, |L'_V|*|L'_E|\})\})$. After this phase, G might have been partitioned in s connected graphs that are compared with G' . Let $GSet$ be the set of graphs derived from this process, step 4 of Fig. 3.

The processing phase consists on finding the mapping between vertices and edges of G' and G_i , where $G_i \in GSet$ (the process is applied to each graph in $GSet$, while a DFC code has not been found in the case of a decision problem, steps 5–12 of Fig. 3).

The backtracking algorithm is a recursive function, see Fig. 4. The input arguments of the `RecursiveIteration` function are: v , the vertex to expand; dim , the dimension of the solution array; DFC' from G' ; DFC from G and L_{VEV} . DFC only stores the vertices' indices because the last two fields of DFC' should be the same as in DFC .

The process starts by finding the vertices v in G_i where $\alpha(v) = \alpha(v_i)$, $v_i \in G'$, $deg(v) \geq deg(v_i)$, and $dfc'_1 = (i, j, c_x, F)$ where $c_x = L_{VEV}(\{v_i, v_j\})$. Taking these vertices, the construction process of DFC begins, where each vertex forms a root of expansion. The vertices found are stored in S , step 7 of Fig. 3.

Based on a selected vertex $v \in G_i$, the algorithm finds all the possible mappings that can be associated to entry dfc'_i of DFC' . In a traditional DFS expansion, the rule is to expand in depth at any step. In our case we implemented a width-depth search. The entries of DFC' are divided in two groups: *forward* (F) and *backward* (B) edges. If an entry is *forward*, it means that the edge is oriented to a vertex that is being visited for the first time. We test if there is an edge adjacent to vertex v in G_i for each edge adjacent to vertex v' (vertex to expand) in G' , (v and v' are mapped) and their

```

Input: ( $G'$ ,  $G$ )
Output: A mapping (if one exists)

1. Build the  $L_{VEV}$  code of  $G'$ 
2.  $DFC' \leftarrow$  Build the Depth First Code
3.  $G'' \leftarrow$  Pre-Processing the graph  $G$ 
4.  $GSet \leftarrow$  All Connected graphs in  $G''$ 
5. while  $GSet$  is not empty
6.  $G_i \leftarrow$  select a graph from  $GSet$  and
   remove it
7.  $S \leftarrow$  vertices in  $G_i$  mapping to
    $dfc'(v_i)$ 
8. while  $S$  is not empty
9.  $v \leftarrow$  select a vertex in  $S$  and
    $S \leftarrow S - \{v\}$ 
10.  $\forall e \in G_i$ :  $mark(e) = false$ 
11.  $Re \leftarrow RecursiveIt(v, 0, DFC', DFC, L_{VEV})$ 
12. if  $R$  is true shows the mapping and
   finish
13.  $G'$  is not a subgraph isomorphic of
    $G$ 

RecursiveIt ( $v$ ,  $dim$ ,  $DFC'$ ,  $DFC$ ,  $L_{VEV}$ )
1. if  $dim = |DFC'|$  return true
2.  $dim \leftarrow dim + 1$ 
3. ( $v'_i$ ,  $v'_j$ ,  $c_x$ ,  $et$ )  $\leftarrow DFC'[dim]$ 
4. if not  $\exists$  adjacent edges to  $v$  not visited
5.  $v' \leftarrow$  vertex in  $DFC$  associated to  $v'_i$ 
6. return  $RecursiveIt(v', dim-1, DFC', DFC, L_{VEV})$ 
7. if  $et = "F"$ 
8. if  $Adj\_List(v'_i) \not\subset Adj\_List(v)$  (based on  $L_{VEV}$ )
9. return false
10.  $OpSet \leftarrow \forall (v, v_k) \in Adj\_List(v): L_{VEV}(v, v_k) = c_x$ 
   and  $mark(v, v_k) = false$ 
11. while  $OpSet$  is not empty
12. ( $v_i$ ,  $v_j$ )  $\leftarrow$  select an entry in  $OpSet$ 
13.  $OpSet \leftarrow OpSet - (v_i, v_j)$ 
14.  $DFC[dim] \leftarrow (v_i, v_j)$ 
15.  $mark(v_i, v_j) = true$ 
16. if  $RecursiveIt(v_j, dim, DFC', DFC, L_{VEV})$  is true
17. return true
18.  $mark(v_i, v_j) = false$ 
19. else
20. if  $\exists (v, v_j)$  where  $v_j$  is associated to  $v'_j$ 
21.  $DFC[dim] \leftarrow (v, v_j)$ 
22.  $mark(v, v_j) = true$ 
23. if  $RecursiveIt(v_j, dim, DFC', DFC, L_{VEV})$  is true
24. return true
25. else
26.  $mark(v, v_j) = false$ 
   return false

```

Fig. 3. The Mapping Main Routine**Fig. 4.** RecursiveIteration Routine

L_{VEV} codes are the same. Mappings that do not have the minimum number of L_{VEV} combinations are pruned and we avoid explore them, line 8 in Fig 4.

Line 10 in Fig. 4 builds the set of vertices $OpSet$ that can be mapped to vertex v'_j , based on the L_{VEV} codes. $OpSet$, is processed entry by entry in a recursive way, lines 12 – 18. If $RecursiveIt$ returns true, it means that the expansion through vertex v_j produced a valid mapping and the process finishes, line 17. If $RecursiveIt$ is false, it means that the expansion through v_j is not valid and then it is discarded (line 18), and the process starts with other possible expansion from $OpSet$. In step 8, $Adj_List(v)$ and $Adj_List(v')$ represent the adjacent lists of non visited edges of vertex v and v' and the contention test is based on the L_{VEV} values of each edge. Backward edges are processed without special considerations, because if there is a mapping between edges e' in G' and e in G_i , then both have related adjacent vertices, lines 20 – 22.

For undirected graphs an entry in DFC is $dfc'_x = (i, j, c_k, t)$ and then the direction of the edge is not stored in this code. A solution to store it consists on adding a new field for each code. For directed graphs, an entry in DFC is defined as $dfc'_x = (i, j, c_k, t, d)$ where i, j, c_k , and t mean the same that for undirected graphs. On the other hand, d is a Boolean variable, where 1 means that the direction of the edge goes from vertex i to vertex j and 0 from j to i . The DFC' construction follows the same criteria as for undirected graphs with a few differences. First, we store the edge direction in d . Second, the degree concept changes for directed graphs because now we have an input degree and an output degree. Then, the rule a) in Section 3 applies only to the output degree. Third, DFC' is built as in the case of undirected graphs (in the expansion process, the edge direction is not considered). Clearly, in the matching process, line 8 of Fig. 5, the value of d is the same.

5 Analysis

In this section we demonstrate two important properties of the proposed algorithm: soundness and completeness.

Soundness

To prove that the algorithm is sound, it is necessary to prove the following three properties.

Property 1. If $DFC' = \langle dfc'_1, \dots, dfc'_n \rangle$ and $DFC = \langle dfc_1, \dots, dfc_m \rangle$, then $n=m$ (proof omitted).

Property 2. $\forall dfc'_x \in DFC'$ and $dfc_x \in DFC$, the L_{VEV} code is equal for both of them (proof omitted).

Property 3. DFC is valid if $\exists f: DFC' \rightarrow DFC$ as bijection where $f(dfc'_x) = dfc_x$.

Proof

It is easy to see that OpSet only includes edges e where $mark(e) = false$, line 10 of Fig. 4. Therefore, each entry dfc_x that is added to DFC represents an edge that has not been visited yet and the corresponding edge e represented by dfc_x is not considered in future recursive calls (in depth). Then, dfc_x will be associated to only one entry in DFC . Note that properties 1, 2 and 3 prove that our algorithm is sound.

Completeness

To prove that the algorithm is complete, we must show that the implemented pruning phases do not discard any possible valid result.

First, we prove that the algorithm is complete without any pruning phase. It is easy to see that the main routine considers every possible root in G , where the root vertex labels are equal to the first vertex in DFC' , line 4 from Fig. 4. After this, the RecursiveIt routine expands DFC adding a new entry at each step. These new entries come from OpSet that includes all the edges adjacent to v and in the worst case the algorithm will explore all the possible combinations of codes. Now, we prove that the implemented pruning phases do not discard any possible valid result.

In the pruning phase, we first remove $\forall v \in V: \alpha(v) \notin L'_v$ and $\forall e \in E: \beta(e) \notin L'_E$. We use L_{VEV} to remove $\forall e \in G$ where $L_{VEV}(e) \notin L_{VEV}$. Note that if $e = (v_i, v_j)$ and $L_{VEV}(e) \notin L_{VEV}$, then $\alpha(v_i) \notin L'_v$ or $\beta(e) \notin L'_E$ or $\alpha(v_j) \notin L'_v$. Therefore, no valid results are affected. We also remove edges in G without the minimum number of repetitions. That is, $\forall e \in G$ is removed if the number of edges in G with the same $L_{VEV}(e)$ is k and k is smaller than the number of edges in G' for the same $L_{VEV}(e)$ code. Finally, in the pruning phase we use the degree of the vertices to remove $\forall v \in G: deg(\alpha(v)) < deg_{min}(\alpha')$ and $\alpha(v) = \alpha'$. So, if a vertex v is removed, it means that the number of edges incident to v is smaller than $deg_{min}(\alpha')$ where $\alpha(v) = \alpha'$ and then, no valid results containing v are removed. Another pruning phase is implemented inside RecursiveIt, line 8. In this step, the list of edges adjacent to v'_i (vertex in G' that is being expanded) is checked, searching that for each adjacent edge, there exists at least one edge e in G to which it can be mapped (where e is adjacent to v). This process is performed based on the L_{VEV} codes. If one edge adjacent to v'_i can not be mapped, it means that v'_i can not be mapped to v . Then, at each recursive call, only vertices with these characteristics are discarded.

Finally, to prove that the algorithm will always finish, we must show that the pruning and processing phases are sound. However, since G is finite, the pruning phases finishes when all vertices and edges are removed. On the other hand, a vertex or edge is considered only once in the mapping process (see line 10 of Fig. 4), and then this process stops when there are not vertices and edges without association. With the above-mentioned, we prove that the algorithm terminates.

6 Experimental Results

The performance of the proposed algorithm was examined using graphs with different dimension and topologies. These experiments were divided in four sections.

First, we tested directed graphs with different labels for each vertex and edge. Second, graphs having the same label for all their vertices and edges. Third, tests when a SI does not exist and fourth, experiments oriented to solve the Hamiltonian problem in dense graphs and to solve the 2-SAT and 3-SAT problems. All the experiments were done with a 2.8 Ghz Intel PC with 1 GB of main memory running Red Hat Linux 9.0. All the reported times are expressed in seconds. We compared the performance of our algorithm with an implementation available in the Web as part of the Subdue system (<http://ailab.uta.edu/subdue/>), called *sgiso*, where an expansion with candidate generation is employed. On the other hand, algorithms like AGM, FSG, SEuS, gSpan are not comparable directly, because these find common patterns in a set of graphs (moreover, some of them are not available and other versions have restrictions over the input graphs). Moreover, the traditional Ullman algorithm is not comparable directly because it works with unlabeled graphs.

For the case in which all the labels are different, we know that the SI Problem can be solved in P. According to our results (shown in Table 1), the response time of our algorithm suffers a proportional increase according to the dimension of the directed graphs. We registered the time taken to find S (S.I.D.) and the total time taken to find S including the preprocessing and the time to write the output. Note that if the graph dimension of G' and G is small, *sgiso* reports similar runtimes (in some cases, better). This result can be explained because the structure of the graphs helps *sgiso* to find the right mapping. However, if the graph dimension is larger, our representation and the pruning phase show their effectiveness with shorter runtimes.

Table 2 shows the results of the experiments where all the labels of the vertices and edges are the same. For this case, *sgiso* was not able to find a solution in less than 2 hours (represented by “-”). This behaviour is a consequence of the input data because the number of combinations increases a lot, on the other hand, our approach has a better response time. This is a consequence of our concepts because not all the vertices have the same L_{VEV} combination and degree.

Table 3 shows some results where there does not exist any instance S in G (all labels are different). These experiments show that our algorithm found a solution very quickly, because our pre-processing phase prunes a high number of vertices and edges of the graph (in our experiments most of the G graphs were decomposed in subgraphs with only one vertex). On the other hand, *sgiso* had to do a larger number of operations to discover that there did not exist any mapping.

Table 1. Tests where all Labels are Different

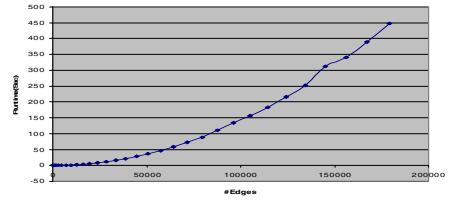
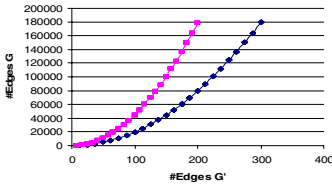
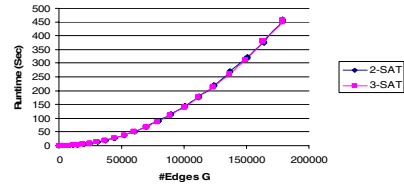
G'		G		Sgiso Time	S.I.D. Time	Total Time.
V'	E'	V	E			
705	4290	1000	8047	1.78	0.01	1.44
750	4558	1000	8047	1.99	0.01	1.83
800	6431	1000	8047	3.09	0.03	4.47
851	6841	1000	8047	3.19	0.04	5.84
909	7303	1001	8047	3.52	0.05	6.97
952	7655	1000	8047	3.69	0.06	7.89
1000	10039	2000	20078	15.22	0.09	15.55
1000	10039	4000	40156	29.36	0.09	15.50
1000	10039	8000	80312	36.58	1	15.55

Table 2. Tests where all Labels are the same

G'		G		Sgiso Time	S.I.D. Time	Total Time.
V'	E'	V	E			
120	177	601	900	-	0	0.03
240	357	600	900	-	0	0.03
360	537	609	900	-	0	0.03
159	474	805	2400	-	0	0.17
321	954	803	2400	-	0.01	0.18
480	1434	800	2400	-	0.01	0.21
204	877	1003	4500	-	0.01	0.56
400	1777	1007	4500	-	0.01	0.59
609	2676	1002	4500	-	0.04	0.70

Table 3. Tests where not Mapping Exists

G'		G		Sgiso Time	S.I.D. Time	Total Time.
V'	E'	V	E			
120	1482	605	35943	7.39	0	0.13
120	1445	595	35946	7.38	0	0.12
120	1471	600	95942	7.47	0	0.12
160	2644	806	63923	24.26	0	0.34
160	2639	809	63921	24.29	0	0.38
160	2547	800	63922	24.50	0	0.26
200	2022	1020	64824	59.47	0	0.69
200	2062	1011	83441	60.01	0	0.74
200	2011	1015	83475	60.40	0	0.68

**Fig. 5.** Response Time for the Hamiltonian Circuit Problem**Fig. 6.** Dimension of G' with Respect of G **Fig. 7.** Response Time for SAT Problems

The SI problem can be used to solve the Hamiltonian problem, using a graph G' where $\forall v \in V: deg(v) = 2$ and for each pair of vertices v_i, v_{i+1} , there exists an edge between them (G' is a ring). If we consider any instance (G', G) of the SI problem where G' has a ring topology, then, determining whether there exists a graph S , such that $S \subseteq G$ and $S \cong G'$ is known to be NP-C. Nevertheless, if $|V'| = |V|$, $|L_{V'}| = |L_E| = 1$ and G is dense [10], then the solution to this decision problem is in P. Based on this property, we tested our algorithm to solve the Hamiltonian problem over undirected dense graphs, where all the labels are the same. The graphs were generated with different dimensions, from 20 vertices and 173 edges to 600 vertices and 179103 edges. As we can see in Fig. 5, the response time again increases according to the dimension of the input graphs.

We also performed experiments for the 2-SAT and 3-SAT problems using a reduction. In these experiments, we generate instances starting from 12 to 300 clauses, where the number of variables is randomly chosen from #Clauses to $(2 \text{ or } 3) * \#Clauses$ (2 for the 2-SAT and 3 for the 3-SAT problem). As we know, 2-SAT is a P problem and 3-SAT is a NP-C problem. In Fig. 6 we compare the dimension of the G' and G graphs and Fig. 7 shows the response time to solve the problems. It is

interesting to see how this graphic grows, because it was very similar for both problems and it did not suffer an exponential behavior.

7 Conclusion

In this work we presented a new approach to solve the SI Problem. We implemented concepts like a code-based list representation, a step by step expansion model without candidate generation, a width-depth search and a pruning phase. In our experiments, we used several types of graphs with the aims to get a good measure of the proposed algorithm performance. Our experiments showed that our approach prunes a high number of paths and finds the mappings very quickly. Moreover, our algorithm was capable to find a solution in polynomial time with special cases such as graphs where all the labels are different, the Hamiltonian Circuit over dense graphs and the 2-SAT problem. With the 3-SAT problem, which is a NP-C problem, our response time was very attractive. We will continue our research comparing our approach with other algorithms focused to solve the problem. We will also do experiments with real domains, such as the chemical toxicity, biological and Web logs domains. We are also implementing a version to be used with the Subdue system, where it is necessary to evaluate when a graph is a SI of another graph. Finally, we are developing research focused to extend our algorithm to solve problems such as graph based data mining.

References

1. Cook, D. J.; Holder, L. B. Substructure discovery using minimum description length and background knowledge. *Journal of Artificial Intelligence Research*, 1:231-255, 1994.
2. Inokuchi, A. Washio, T. Motoda, H. Complete Mining of Frequent Patterns from Graphs: Mining Graph Data. *Machine Learning*, Kluwer Academic Publishers. (2003). 321 – 354.
3. Xifeng, Y. Jiawei H. gSpan: Graph – Based Substructure Pattern Mining. Technical Report. University of Illinois. (2002).
4. Shayan G. Sudarshan S.C. “SEuS: Structure Extraction using Summaries”. Technical Report. C. S. Department, University of Maryland, (2002).
5. Kuramochi, M.; Karypis, G. An Efficient Algorithm for discovering Frequent Subgraphs. Tech. Report. Dept. of Computing Science, University of Minnesota. June (2002).
6. Garey R. Michael. Johnson S. David. *Computers and Intractability, A Guide to the theory of NP-Completeness*. W. H. Freeman and Company (2003).
7. Zaki, M. J. Efficiently Mining Trees in a Forest. Tech. Report 01-7, Computer Science Department, Rensselaer Polytechnic Institute, (2001).
8. Diestel, Reinhard. *Graph Theory*. Springer – Verlag. Second Edition (2000).
9. Thulasiraman, K. Swamy, M.N.S. *Graphs: Theory and Algorithms*. Wiley – Interscience. 460 pp (1992).
10. C. L. Liu. *Elements of Discrete Mathematics*. McGraw Hill. Second Edition (1985).
11. Olmos, Ivan. Gonzalez, Jesus A., Osorio, Mauricio. Subgraph Isomorphism Detection using a Code Based Representation. Proceedings of the 18th FLAIRS Conference, 2005.

Moderated Innovations in Self-poised Ensemble Learning

Ricardo Nanculef¹, Carlos Valle¹, Héctor Allende¹, and Claudio Moraga²

¹ Universidad Técnica Federico Santa María,
Departamento de Informática, CP 110-V Valparaíso, Chile

{jnancu, cvalle, hallende}@inf.utfsm.cl

² Dortmund University, 44221 Dortmund, Germany
claudio.moraga@udo.edu

Abstract. Self-poised ensemble learning is based on the idea of introducing an artificial innovation to the map to be predicted by each machine in the ensemble such that it compensates the error incurred by the previous one. We will show that this approach is equivalent to regularize the loss function used to train each machine with a penalty term which measures decorrelation with previous machines. Although the algorithm is competitive in practice, it is also observed that the innovations tend to generate an increasingly bad behavior of individual learners in time, damaging the ensemble performance. To avoid this, we propose to incorporate smoothing parameters which control the introduced level of innovation and can be characterized to avoid an explosive behavior of the algorithm. Our experimental results report the behavior of neural networks ensembles trained with the proposed algorithm in two real and well-known data sets.

1 Introduction

Ensemble learning algorithms [11] work under the principle of combining a set of simple learners to solve the problem at hand, instead of using a single complex learner. That is, to infer a function $f : X \rightarrow Y$ we first choose a set of hypotheses $F = \{f_1, f_2, \dots, f_t\}$, $f_i : X \rightarrow Y$ from a base hypothesis space H , and then we consider as an approximation to $f(x)$ a composition $\bar{f}^t(x) = \bigoplus F$ of this set which for regression is typically a weighted average $\bar{f}^t(x) = \sum_i^t w_i f_i(x)$. What is the key in the selection of the individual hypothesis, to get real advantages of combining predictors, is *diversity* of their members. This property has been widely studied in the literature [5] [7] [6] both for regression and classification ensembles. In training with the quadratic loss function, diversity is easily understood as the second term in the so called *ambiguity decomposition* [1]:

$$(\bar{f}_t - y)^2 = \sum_i w_i (f_i - y)^2 - \sum_i w_i (f_i - \bar{f}_t)^2 \quad (1)$$

It states that the quadratic loss of the ensemble is the weighted average of individual errors minus the weighted average of individual deviations with respect

to the ensemble. The latter term (also called “ambiguity”) guarantees that the ensemble error is always less than the average error of members which makes the ensemble a reasonable gamble.

Self-poised ensemble learning [9] sequentially creates an ensemble for function approximation, modifying the training targets at each time t with an artificial innovation which is exactly the opposite of the error of previous learner. This approach ensures diversity and furthermore stochastic convergence, however it can be observed that usually the introduced innovations cause that the map to be estimated by each machine be increasingly irregular and so, the local errors are explosive in time. In such cases self-poised learning behaves like bagging or worse. To avoid this problem, we propose to incorporate smoothing parameters which control the introduced level of innovation. We also show that both the original algorithm and the smoothed one are equivalent to modify the loss function used by each machine with a penalty term which measures the level of decorrelation with previous machines.

2 Innovations in Self-poised Learning

Self-poised ensemble learning sequentially generates a set of learning machines to combine using a uniformly weighted average $\bar{f}_t = \sum_{i=1}^t f_i/t$, which will be used as an approximation to an unknown function $f : X \rightarrow Y$, $X \subset \mathbb{R}^{dx}$, $Y \subset \mathbb{R}^{dy}$, $dx, dy \in \mathbb{N}$. The first machine is trained using the original training data set and subsequent machines are generated to approximate the following “innovated targets”

$$\tilde{y}_{t+1} = y + \epsilon_t \quad (2)$$

where ϵ_t (the innovation) corresponds to the error of the machine at time t . So, the ensemble at time t can be expressed in terms of the local errors as:

$$\bar{f}_t = y - \frac{\epsilon_t}{t} \quad (3)$$

The last result is particularly suitable to prove convergence. In fact, suppose the space $X \times Y$ is equipped with a probability measure P . So, the probability of the event $A_t = \{(x, y) \mid |y - \bar{f}_t(x)| > \xi\}$ can be bounded using the Chebyshev inequality as

$$P(A_t) < \frac{E[|y - \bar{f}_t|]}{\xi} = \frac{E[|\epsilon_t|]}{t\xi} \quad (4)$$

If $E[|\epsilon_t|]$ can (almost surely) be bounded by an a constant K , convergence in probability is straight-forward from the Borel-Cantelli lemma. However if the sequence of local errors $(\epsilon_t)_t$ cannot be bounded for all t , for example the sequence has an increasing tendency, convergence is not guaranteed. Table (1) shows a typical run of self-poised ensemble learning with 10 neural networks machines trained with standard backpropagation. It shows that the performance of the members (in its own data sets) is highly damaged by the introduced innovations and then, the ensemble error are not monotonically decreased. Monotonic

Table 1. *Machine Error* and *Ensemble Error* are respectively the error of the t st machine and the t st ensemble in the prediction of its training patterns. *Diversity* and *Average Error* are the ambiguity component of the ensemble loss and the average error of machines respectively, as defined in equation (1).

Time t	Machine Error	Ensemble Error	Diversity	Average Error	Condition Satisfied?
2	35,696	8,924	2,024	10,998	yes
4	139,906	8,744	3,971	12,715	yes
6	299,855	8,329	6,435	14,765	no
8	546,633	8,541	7,724	16,265	yes
10	868,209	8,682	11,780	20,462	no

growth of diversity (second term in (1)) shows that the idea of introducing an innovation in the opposite direction to the error of the last machine works well in practice. However, from the fact that local errors (first term in (1)) damage the performance of the ensemble seems reasonable to think that the magnitude of the innovations is not the appropriate to get the desired behavior.

3 Smoothing Self-poised Ensemble Learning

In this section we propose to modify the magnitude of the introduced innovations so that the explosive behavior exposed in the previous section can be avoided. So, at time $t \geq 1$ we will consider a modified innovation $\eta_t \epsilon_t$, such that the map to be estimated by the $(t + 1)$ -machine be

$$\tilde{y}_{t+1} = y + \eta_t \epsilon_t \quad (5)$$

where the parameter η_t is named “the smoothing parameter”. The quadratic loss of this machine in its training patterns is then given by

$$\begin{aligned} L_{t+1} &= \left| \tilde{y}_{t+1} - \tilde{f}_{t+1} \right|^2 = \left| y + \eta_t \epsilon_t - \tilde{f}_{t+1} \right|^2 \\ &= \left| y + \eta_t y + \eta_t \eta_{t-1} \epsilon_{t-1} - \eta_t f_t - f_{t+1} \right|^2 = \dots = \left| \sum_{i=1}^{t+1} \pi_i^{t+1} (y - f_i) \right|^2 \end{aligned}$$

where

$$\pi_t^t = 1, \pi_i^t = \prod_{j=i}^{t-1} \eta_j \quad (6)$$

We can write equivalently

$$\begin{aligned} L_{t+1} &= \sum_{i=1}^{t+1} \sum_{j=1}^{t+1} \pi_i^{t+1} (y - f_i) \pi_j^{t+1} (y - f_j) \\ &= 2 \sum_{j=1}^t \pi_j^{t+1} (y - f_{t+1})(y - f_j) + (y - f_{t+1})^2 + \delta_{t+1} \end{aligned}$$

where δ_{t+1} does not depend on f_{t+1} . That is, training the $t + 1$ st machine with the innovated target \tilde{y}_t it is equivalent to consider the “regularized” loss function:

$$E_{t+1} = (y - f_{t+1})^2 + 2 \sum_{j=1}^t \pi_j^{t+1} (y - f_{t+1})(y - f_j) \quad (7)$$

This equation says that the $t + 1$ st machine is trained to get a compromise between the error of the hypothesis in the target data and the decorrelation of this error with the errors of previous machines. The parameters π_j^{t+1} measure the relative influence of the j -th machine in the training of the actual learner and have the objective of changing the optimization landscape to lead a smoothed one. When all the smoothing parameters are $\eta_i = 1$, that is, no smoothing is being applied, we have that $\pi_j^{t+1} = 1 \forall j, \forall t$, and then the local errors $(y - f_j)$ have equal importance in the selection of the new hypothesis. (7) also links the algorithm with other ensemble learning approaches [12], [5] [10] which encourage decorrelation between learners to get diversity. In [9] it has been shown that positive diversity can be achieved by negative correlation between the errors of the members of the ensemble. If the local errors ϵ_t have null expectation, $E[f_j] = y$ and it becomes clear that the function (7) encourages negative covariance. If, furthermore, we have uncorrelated local errors covariance can be reduced to

$$\begin{aligned} \text{covar}(\bar{f}_t) &= \sum_{i=2}^t E[(f_i - y)(f_1 - y)] + \sum_{i=3}^t \sum_{j=2}^{i-1} E[(f_i - y)(f_j - y)] \\ &= - \sum_{i=1}^{t-1} -\eta_i E[\epsilon_i^2] \end{aligned}$$

which proves that the introduced correlation is negative when positive η_i are considered. The same conclusion over the positivity of the smoothing parameters can be obtained if we want to ensure that the gradient of (7) is in the same direction with the gradient of the ensemble error at time $t + 1$. Since η_i must be positive, what is problematic with the innovations are not their direction but their magnitude. Noting that $\epsilon_t = \tilde{y}_t - f_t$ ($\tilde{y}_1 = y$) we have that

$$tE[(f_t - y)] = \sum_{i=1}^{t-1} (\eta_t - 1)\epsilon_t - \epsilon_t$$

Then, if the local errors can be bounded by k (probably after the introduction of smoothing) the convergence in probability of the method can be guaranteed when we have that

$$\begin{aligned} \lim_{t \rightarrow \infty} P(|y - \bar{f}_t(x)| > \xi) &\leq \lim_{t \rightarrow \infty} \frac{E[|y - \bar{f}_t|]}{\xi} \\ &= \lim_{t \rightarrow \infty} \frac{E[|\sum_{i=1}^{t-1} (\eta_t - 1)\epsilon_t - \epsilon_t|]}{t\xi} < \lim_{t \rightarrow \infty} \frac{kE[|\sum_{i=1}^{t-1} (\eta_t - 1)\epsilon_t|]}{t\xi} \end{aligned}$$

is null, that is when $\sum_{i=1}^{t-1} \eta_t$ grows slower than t . It can be achieved when $\eta_t \in [0, 1]$. Now, from (6) we can observe that

$$\pi_i^t = \pi_{i+1}^t \eta_i$$

That is, $\eta_i \in [0, 1]$ leads to a monotonically increasing sequence of parameters π_i^t . The effect of this smoothing strategy over the loss function used by the t -machine, is to limit its memory about the local ensemble errors, increasing the importance of the last machines with respect to the earlier ones. A simple but interesting smoothing function with this properties (named ‘‘constant’’) could be

$$\pi_i^t = k^{t-i} \tag{8}$$

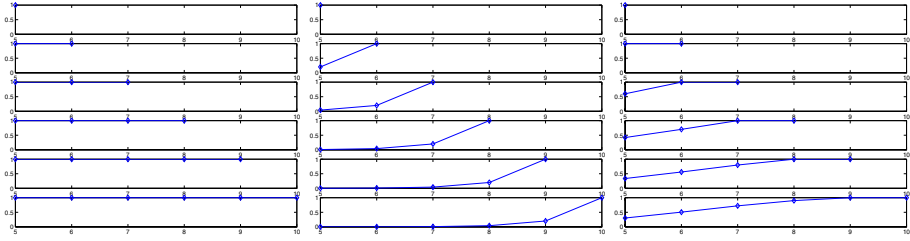


Fig. 1. In the horizontal axis, the machine indexes for $i = 5, \dots, 10$ and in the vertical one, the value of π_i^t , with $t = 10$. From left to right: No smoothing, i.e., $\pi_i^t = 1$; a constant strategy with $k = 0.2$; the heuristic increasing strategy.

Algorithm 1. Smoothed Self-Poised Learning

- 1: Let be M the required number of learners
- 2: Let be $z^1 = \{(x_1, y_1), \dots, (x_n, y_n)\}$ a training set.
- 3: Generate an initial predictor f_1 training a learner with z^1 and the quadratic loss.
- 4: **for** $t = 1$ to $M - 1$ **do**
- 5: Choose a set of parameters π_i^t based on some smoothing strategy.
- 6: Generate a predictor f_t training a learner with the data z^1 and the loss function

$$E_t = (f_t - y)^2 + \sum_{i=1}^{t-1} \pi_i^t (f_t - y)(f_i - y)$$

7: **end for**

8: Set the ensemble to be $\bar{f}_t = 1/M \sum_{i=1}^M f_i(x)$

which corresponds to choose $\eta_i = k$, where k is a constant in $[0, 1]$ measuring how much one needs to relax the level of innovations (and so the optimization landscape) to avoid an explosive growth of local errors. Lower values of k give a loss function (7) which quickly forgets the errors of previous machines. Finally, in our experiments we will analyze the effect of a decreasing set of smoothing parameters selected heuristically to begin with a moderated level of innovation in the first machines and a total level of innovation in the last machine. This approach is based in the idea that strongly moderating the innovations at the beginning of training allows ensemble error to decrease regularly in time. The function, named “decreasing” is defined as

$$\eta_i^t = t/T \tag{9}$$

which is in $[0, 1]$ and leads to $\pi_i^t = \frac{t!}{i!T^{i-t}}$. Figure 1 depicts the behavior of some smoothing strategies. Note that the effect on the memory of the algorithm about the past errors can be modelled more or less aggressive depending of the strategy. Algorithm 2 is the final proposed algorithm with the π_i^t parameters.

4 Experimental Results

Our experimental analysis will be made with respect to two real and well-known data sets, *Boston* and *Building2*¹. Each reported experiment, that is, the algorithm and parameters combination, was repeated a minimum of 50 times with the *Boston* data, randomly reordering this at each time. With the *Building2* data, only 20 trials were ran because in this case we have a pre-established order, and repetitions only reflect the variability of the initialization of the learning machines. As learning machines we select neural networks with one hidden layer trained with the quadratic loss, using standard gradient descent with learning parameter $\alpha = 0.2$. We follow [8] to select a training set equivalent to 75% of the total number of examples in the data and 25% for testing. The data was scaled to the unitary hypercube centered in the origin, as usual in neural network literature. Reported errors were however computed in the original scale.

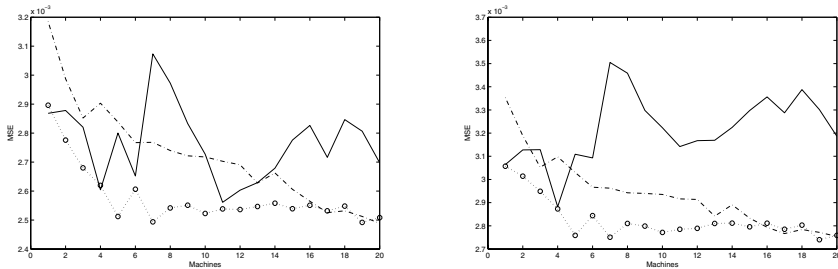


Fig. 2. In the horizontal axis the number of machines. In the vertical axis the quadratic loss in the training (left) and testing (right) set of the *Building2* data. The solid curve corresponds to a non-smoothed strategy, the circle-dotted line to a constant smoothing strategy and the dashed line to an increasing smoothed strategy.

Table 2 shows the results of different smoothing strategies and implementations in the *Boston* and *Building2* data. In both we include for comparison purposes, an increasing smoothing strategy implemented working directly with perturbations and other implemented working with the modification of the loss function. Algorithms are also compared with bagging and boosting as defined in [3] and [4] respectively. Figure 2 shows the behavior of the algorithm when learning the *Building2* data. For this comparison, the median of the ensemble error was selected and then the complete training “history” was recovered. It should be observed that smoothing can lead to an almost monotonic reduction of the ensemble error in contrast with the non-smoothed algorithm.

¹ A detailed description of both data sets can be obtained from the *UCI Machine Learning Repository* [2].

Table 2. Performance results with the Boston and Building Data. M is the number of machines and N the number of hidden neurons. **Incr.1** corresponds to an increasing smoothing strategy implemented directly innovating the targets. **Incr.2** is the same strategy but implemented modifying the loss function. With this implementation we also show the algorithm without a smoothing strategy (**Pure**) and with a constant smoothing strategy (**Const.**). We also present the results for Boosting and Bagging.

M	N	Incr.1	Pure	Const.	Incr.2	Bagging	Boosting
Mean in the Building2 Training Set							
5	5	2,504E-3	2,616E-3	2,600E-3	2,735E-3	2,657E-3	2,743E-3
5	10	2,327E-3	2,579E-3	2,563E-3	2,530E-3	2,336E-3	2,394E-3
10	5	2,968E-3	2,636E-3	2,526E-3	2,490E-3	2,562E-3	2,828E-3
10	10	2,395E-3	2,541E-3	2,385E-3	2,370E-3	2,264E-3	2,480E-3
20	5	2,423E-3	2,307E-3	2,325E-3	2,307E-3	2,529E-3	2,735E-3
20	10	2,596E-3	2,419E-3	2,296E-3	2,263E-3	2,235E-3	2,360E-3
Mean in the Building2 Testing Set							
5	5	2,588E-3	2,963E-3	2,829E-3	2,988E-3	2,762E-3	2,918E-3
5	10	2,593E-3	3,034E-3	2,807E-3	2,812E-3	2,539E-3	2,698E-3
10	5	3,145E-3	3,077E-3	2,786E-3	2,778E-3	2,687E-3	2,970E-3
10	10	2,817E-3	2,999E-3	2,771E-3	2,755E-3	2,467E-3	2,751E-3
20	5	2,767E-3	2,876E-3	2,724E-3	2,673E-3	2,629E-3	2,951E-3
20	10	2,654E-3	2,963E-3	2,699E-3	2,658E-3	2,431E-3	2,690E-3
Standard Deviation in the Building2 Testing Set							
5	5	4,211E-5	1,228E-4	5,886E-5	1,617E-4	1,586E-4	1,162E-4
5	10	2,516E-5	1,740E-4	8,197E-5	6,081E-5	2,539E-4	1,112E-4
10	5	6,897E-5	1,721E-4	3,932E-5	4,866E-5	1,381E-4	1,693E-4
10	10	3,585E-5	5,187E-5	5,467E-5	5,507E-5	1,111E-4	1,303E-4
20	5	4,244E-5	2,036E-5	7,578E-5	6,600E-5	1,284E-4	1,006E-4
20	10	3,377E-5	1,263E-4	4,756E-5	3,662E-5	1,365E-4	0,851E-4
Mean in the Boston Training Set							
5	5	9,5324	7,2955	7,5734	8,2415	11,0582	10,6459
5	10	8,8845	6,4907	7,8874	7,4423	10,8725	9,8158
10	5	9,5444	6,0566	6,6973	7,6766	11,5588	8,9136
10	10	7,9763	5,7170	7,2921	6,9138	9,7304	7,2538
20	5	9,8657	5,2734	7,3518	7,4988	10,5024	8,4366
20	10	8,7424	5,0133	6,9058	6,7218	9,8524	6,7876
Mean in the Boston Testing Set							
5	5	17,1740	16,7563	14,6694	16,2701	17,5992	18,0869
5	10	15,7029	15,9738	14,6410	16,5934	17,2186	17,0666
10	5	14,8049	15,9692	13,9670	16,0237	16,0329	16,9364
10	10	15,6729	15,9699	14,3401	15,4006	15,9964	16,6778
20	5	17,2894	15,9430	13,8284	16,0650	16,2457	16,0660
20	10	15,9519	16,3352	14,1371	15,7663	14,2401	14,7374
Standard Deviation in the Boston Testing Set							
5	5	5,0867	6,0437	4,5304	4,3309	5,3182	6,0985
5	10	4,9682	5,8055	5,8409	5,2537	6,0785	4,4294
10	5	4,3724	5,5499	4,2611	5,3826	4,1598	5,0401
10	10	5,1772	5,8035	5,8489	4,7920	5,6799	4,4665
20	5	6,4968	6,0056	4,4595	6,4307	4,2255	3,7182
20	10	5,4697	5,8201	5,5853	5,1754	4,3362	3,7456

5 Conclusions

In this paper we have analyzed practical and theoretical problems that can arise when training an ensemble with self-poised ensemble learning. Experiments with neural networks ensembles in two real data sets show that by using smoothing strategies, a better performance may be achieved than without using them. Furthermore, a clear reduction of the variance is obtained. This allows to obtain algorithms which can be better than bagging or boosting. Another important result when using smoothing strategies is the improvement of the generalization rate -(quotient between training error and testing error)- leading to lower testing errors even in cases of eventually larger training errors. Future work has to study how to automatically determinate the appropriate smoothing strategy and has also to explain the better generalization performance of smoothing strategies.

References

1. J. Vedelsby A. Krogh, *Neural network ensembles, cross-validation and active learning*, Neural Information Processing Systems **7** (1995), 231–238.
2. C.L. Blake and C.J. Merz, *UCI repository of machine learning databases*, 1998.
3. L. Breiman, *Bagging predictors*, Machine Learning **24** (1996), no. 2, 123–140.
4. H. Drucker, *Improving regressors using boosting techniques*, Fourteenth International Conference on Machine Learning, 1997, pp. 107–115.
5. R. Harris G. Brown, J. Wyatt and X. Yao, *Diversity creation methods: A survey and categorisation*, Information Fusion Journal **6** (2004), no. 1, 5–20.
6. L.I. Kuncheva and R.K. Kountchev, *Generating classifier outputs of fixed accuracy and diversity*, Pattern Recognition Letters (2002), no. 23, 593–600.
7. C. Whitaker L. Kuncheva, *Measures of diversity in classifier ensembles*, Machine Learning **51** (2003), 181–207.
8. L. Prechelt, *Proben1 - a set of benchmarks and benchmarking rules for neural training algorithms*, Tech. Report 21/94, Universitat Karlsruhe, 1994.
9. H. Allende C. Moraga R. Nanculef, C. Valle, *Self-poised ensemble learning*, Lecture Notes in Computer Science, vol. 3646, Springer, 2005, pp. 272–282.
10. B. Rosen, *Ensemble learning used decorrelated neural networks*, Connection Science **8** (1999), no. 3-4, 373–384.
11. R. Schapire, *The strength of weak learnability*, Machine Learning **5** (1990), 197–227.
12. X. Yao Y.Lui, *Ensemble learning via negative correlation*, Neural Networks **12** (1999), no. 10, 1399–1404.

An Adaptive Framework for Solving Multiple Hard Problems Under Time Constraints

Sandip Aine, Rajeev Kumar, and P.P. Chakrabarti

Department of Computer Science and Engineering,
Indian Institute of Technology Kharagpur,
Kharagpur, WB 721302, India
{sandip, rkumar, ppchak}@cse.iitkgp.ernet.in

Abstract. We address the problem of building an integrated meta-level framework for time deliberation and parameter control for a system solving a set of hard problems. The trade-off is between the solution qualities achieved for individual problems and the global outcome under the given time-quality constraints. Each problem is modeled as an *anytime* optimization algorithm whose quality-time performance varies with different control parameter settings. We use the proposed meta-level strategy for generating a deliberation schedule and adaptive cooling mechanism for anytime simulated annealing (ASA) solving *hard* task sets. Results on task sets comprising of the traveling salesman problem (TSP) instances demonstrate the efficacy of the proposed control strategies.

1 Introduction

The development of problem solving frameworks that can adequately handle quality-time trade-offs while solving constrained optimization problems has been a major area of interest in automated reasoning. Typically, we are given a set of problem instances that have to be solved within a limited time-frame in such a way that the quality of solutions obtained is maximized. The quality measure is usually some form of an aggregate of the individual solution qualities.

Several approaches have been suggested in the past to handle the quality-time trade-off of optimization algorithms. The proposed frameworks have two major characteristics. The first is the use of an algorithm, which is interruptible and can provide some (probably sub-optimal) solution at almost every stage of their execution (anytime algorithm) [1, 2, 3], to solve the individual problems at hand. Solution quality is expected to improve with allocation of additional time, providing opportunity to reason about the quality-time trade-off. The second characteristic of frameworks that handle problem solving under quality-time constraints is to use past data, based on some sort of learning, to take informed decisions on time allocation to problem instances [4]. A popular approach is to develop a *utility* function that combines quality with time [2, 3] as the trade-off model, and take a deliberation decision, based on the past data, to maximize the utility.

This work is motivated by the fact that when a system is working with a set of problems to be solved under a global time constraint, there is no obvious utility function that can be used to obtain an optimal deliberation schedule. Also, time allocation is not the only parameter that decides the rate of quality improvement of an optimization algorithm. Anytime algorithmic techniques often use a number of other parameters to control the convergence. Some typical control parameter examples are, the inflation rate for anytime A* [5], cooling/annealing schedule in Simulated Annealing (SA) [6] or the parameters for evolutionary algorithms [7]. While adaptive control of such parameters has been a topic of active research, the combination of time deliberation and parameter control has not been formalized to achieve their combined benefits.

We present a generalized meta-reasoning framework that unifies time and parameter control for solving multiple problems under time constraints without the use of utility functions. The meta-level strategies can be static or dynamic. In case of a static strategy, the time allocation and parameter value of each individual instance is decided prior to execution and no revision is done at runtime. For the dynamic case, the control decision is refined at some selected intervals when the current state of the system is monitored. Dynamic strategies can be non-preemptive (where a task once stopped can not be resumed later) or preemptive (any task can be executed at any stage). We propose optimal strategies for all static and dynamic configurations mentioned above. We also present experimental results to support the efficacy of the formulation with task sets comprising of traveling salesman problems. We use an Anytime Simulated Annealing (ASA) algorithm as our base algorithm for the experiments. The proposed monitoring based framework is used to find adaptive cooling schedules along with optimal time allocations for solving multiple problem instances under quality-time constraints.

The rest of the paper is organized as follows. Section 2 includes the definitions and the problem-formulation. Here we discuss the static and dynamic strategies for deliberation and control. We present the experimental results in Section 3. We conclude in Section 4.

2 Problem Formulation and Definitions

When a system is given a number of *hard* problems to be completed within a given deadline, the basic objectives are to maximize the *global* aggregate quality and to minimize the variations among the solutions obtained. In our formulation, the number of tasks and the global deadline is known *a priori*. We consider a controller working on a set of n tasks whose relative weights are defined by an n -element weight vector W ; the tasks should be completed within a global deadline D . Weight indicates the relative importance of an individual task. The strategy is to schedule tasks and set the control parameters such that,

$$\sum_i t_i \leq D \text{ and } \sum_i w_i * q^i_{obtained} \text{ is maximized} \quad (1)$$

Here i denotes the index of the problem in the task queue, t_i is the time allocated to the i^{th} problem and $q^i_{obtained}$ is quality obtained for the i^{th} task. The objective

is to maximize the overall quality of the task set within the constraint of a given global deadline. We use the term, *approximation ratio* [4], throughout the rest of the paper, to refer to *quality* of an available (or intermediate) solution. For cost minimization problems, *approximation ratio* is given by,

$$Appx. Ratio = Cost(Appx. Soln.) / Cost(Opt. or Ref. Soln.) \quad (2)$$

For maximization, the inverse ratio is used. The approximation ratio is a unit-less metric and can be used in a generic way as it does not depend on the exact cost of a solution, which can vary for different problem instances.

In the static meta-reasoning model, time allocation and control parameter values are decided at the start of execution and the decision is not revised during runtime. The static model works well when there is little or no uncertainty about the quality-parameter (including time) relation. For environments where the quality improvement is not that predictable and different parameter settings and schedules may be suitable depending on the current state of system, a dynamic monitoring based strategy is preferred. The decision can be revised, in the dynamic case, during runtime by monitoring the local (non-preemptive) or the global (preemptive) state at pre-decided intervals.

2.1 Static Schedule

The meta-level control framework requires a model to measure the improvement of the solution quality with time for different control parameter-settings. Dean & Boddy [2] introduced the term *performance profile* which represents the expected solution quality as a function of the allocated time for an anytime algorithm. In this work, we extend the basic concept of performance profile to get a *parameterized performance profile* of an anytime algorithm as the model for the algorithm's quality-time relation depending on the control parameter settings.

Definition 1. [Parameterized Performance Profile (PPP)] *Parameterized Performance Profile of an anytime algorithm given by $P(q_i|t, c)$ is the probability of getting a solution with quality q_i , where t is the time allocated and c is the value of the control parameter. (In multi-parameter case, c is a vector of control parameters.)*

The 3-tuple Parameterized Performance Profile ($\langle P, t, c \rangle$) can be represented using a look-up table. All the components of PPP, namely, the solution quality, the computational time and the control parameter values are discretized to have a finite number of choices. Next, with a given PPP for each class of the problems in the problem set, the static meta-level controller has to decide the amount of time allocation and selection of the parameter values so that the aggregate of expected quality is maximized.

Strategy 1. [Fixed Control Strategy] *Given an anytime task set and a strict deadline, an optimal fixed strategy for time allocation and control parameter value is obtained by solving the following equation for Static Value Function V_S :*

$$V_S(i, T_{left}) = (\max t_i, c_i) w_i * \sum_j P_i(q_j|t_i, c_i) * q_j + V_S(i - 1, T_{left} - t_i) \quad (3)$$

where i is the index of the current problem in the queue (also the queue length at this point) and T_{left} is the time left before deadline. $P_i(q_j|t, c)$ denotes the probability of task i to reach a quality level q_j if the allocated t_i and the control parameter is set to c_i . Solving $V_S(n, D)$ we can get the optimal control strategy for time allocation t_i and control parameter value c_i for each problem in the set.

2.2 Dynamic Scheduling Using Run Time Monitoring

If there is uncertainty about the progress of an algorithm then monitoring the algorithm's performance and revising the control strategy (when progress is faster or slower than expected) can significantly improve the utility. Hansen and Zilberstein [4] extended the idea of 'performance profile' including the current quality state in the definition, naming it 'dynamic performance profile'. We adapt the dynamic definition with inclusion of control parameters to produce Dynamic Parameterized Performance Profile (DPPP) for the monitoring based framework. DPPP predicts the quality improvement with time based on the quality of the currently available solution and the control parameter values used.

Definition 2. [Dynamic Parameterized Performance Profile (DPPP)] *The Dynamic Parameterized Performance Profile of an anytime algorithm $P(q_j|q_i, \Delta t, c)$ is defined as the probability of reaching a solution quality q_j if an additional time Δt is given and control parameter value is set to c when the current solution quality is q_i (As indicated earlier, in the multi-parameter case, c would be replaced by a vector of control parameters).*

The model assumes that the improvement of quality is dependent on the current solution quality only and not on the path to reach that solution (Markov property). For other cases, we can strengthen the prediction mechanism by defining a solution state using quality of the solution and the cumulative time to reach the quality and use it to generate the DPPP.

In runtime monitoring based strategy, the controller monitors the algorithm at certain points and decides on a sequence of actions depending on the current state of the system. Monitoring can be continuous assuming the time taken to monitor is negligible or we can incorporate the cost of monitoring within the strategy to decide on an optimal monitoring schedule. We include the cost of monitoring in the framework and generate an appropriate monitoring schedule.

Non-preemptive Dynamic Strategy: In the non-preemptive case, as the name suggests, a task once stopped can not be resumed later. The control policy should take a combined decision based on the following parameters – (i) additional time to run the current task (Δt), (ii) value of the controllable parameter for this period (c), and (iii) a decision whether to preempt the current task or to monitor the system again after the time slot (d).

Strategy 2. [Cost Sensitive Dynamic Non Preemptive Strategy] *The cost sensitive adaptive control strategy $D_N(i, q_j, T_{left})$ is a mapping from a state (i, q_j, T_{left}) where i is the index of the current problem, q_j is the current quality level of the i^{th} problem, and T_{left} is the time left, to a strategy decision*

$(\Delta t, c, d)$, where Δt is the additional time allocated for the current task (indexed by i), c is the control parameter value chosen for this interval, and d is a binary variable which decides whether to monitor or to preempt the current task after Δt . The cost sensitive adaptive control strategy can be obtained optimizing the non-preemptive value function $(V_N(i, q_j, T_{left}))$, defined as follows:

$$V_N(i, q_j, T_{left}) = \max_{\Delta t, c, d} \begin{cases} w_i * \sum_k P_i(q_k | q_j, \Delta t, c) * q_k + \\ V_N(i-1, q_{init}, T_{left} - \Delta t) & \text{if } d = 0 \\ \sum_k P_i(q_k | q_j, \Delta t, c) * V_N(i, q_k, T_{left} - \Delta t) - M_p & \text{if } d = 1 \end{cases} \quad (4)$$

if $d = 0$ the current task is preempted after the allocated interval (Δt) and the next task is started, otherwise the system is again monitored after the chosen interval without any preemption. $P_i(q_k | q_j, \Delta t, c)$ is obtained from the DPPP for the i^{th} problem. M_p is the penalty assigned for each monitoring step.

We use dynamic programming approach to solve the above equation optimally.

Theorem 1. *The cost sensitive dynamic non-preemptive control strategy which maximizes the above value function, represents an optimal control strategy for a set of anytime tasks and a given fixed deadline, when quality improvement is taken to be Markov.*

Proof. The proof follows directly by the application of dynamic programming under the Markov assumption [8]. Intermediate states are described using the quality of the currently available solution. The dynamic programming formulation ensures non-myopic optimality.

Preemptive Dynamic Strategy: When we consider the preemptive case, any task that has been interrupted earlier can again be run from the state where it was interrupted. Here the controlling decisions are slightly different from the non-preemptive case. Now the control strategy needs to take a combined decision based on the following – (i) selection of the task (i) to run next, (ii) duration of the task (Δt) to run, and (iii) value of the control parameter for this period (c). Based on this formulation, we formally define the dynamic preemptive strategy below.

Strategy 3. [Cost Sensitive Dynamic Preemptive Strategy] *The cost sensitive adaptive control strategy $D_P(q^1, q^2, \dots, q^n, T_{left})$ is a mapping from a state $(q^1, q^2, \dots, q^n, T_{left})$ where q^j is the current quality level of the j^{th} task and T_{left} is the time left before the deadline, to a strategy decision $(i, \Delta t, c)$, where i is the index of task chosen for next interval, Δt is the additional time allocated for chosen task, and c is the control parameter value chosen for this interval. The optimal control policy for preemptive tasks can be obtained optimizing the preemptive value function $(V_P(q^1, q^2, \dots, q^n, T_{left}))$, given as follows:*

$$V_P(q^1, q^2, \dots, q^n, T_{left}) = \max_{i, \Delta t, c} \begin{cases} w_i * \sum_j P_i(q_j^i | q^i, T_{left}, c) * q_j^i + \\ \sum_{k(k \neq i)} w_k * q^k \\ \sum_j P_i(q_j^i | q^i, \Delta t, c) * V_P(q^1, q^2, \dots, q_j^i, \dots, q^n, \\ T_{left} - \Delta t) - M_p \end{cases} \quad (5)$$

$P_i(q_j^i | q^i, T_{left}, c)$ is obtained from the DPPP for the i^{th} task. If $\Delta t = T_{left}$ i.e., the first row of the equation is chosen then no monitoring is done otherwise the system is monitored again after Δt . M_p is the penalty for each monitoring step.

The monitoring schedule is inherently generated with this formulation because if the task chosen is given the entire residual time then there will be no monitoring. Otherwise, the system is monitored after the chosen interval and the decision is revised accordingly. Optimality of the decision making can be proved similar to the non-preemptive case.

Maximizing the value functions (using dynamic programming), an off-line optimal policy for time allocation, control parameter value and the monitoring schedule can be computed for each solution state (both for preemptive and non-preemptive cases). At runtime, the controller monitors the progress of the algorithms and decide on parameter-settings using a simple table look up.

Complexity of Preemptive and Non-preemptive Scheduling: In case of preemptive scheduling, the meta-reasoning framework has more control over the global situation as the system state includes all the currently available qualities, and any task can be chosen for execution at anytime. However, the preemptive policy has more overheads in respect of both storage and time.

Considering the storage issue, storage needed for DPPP is equal for both the cases as it is independent of the strategy. But in preemptive scheduling, the number of possible states are of exponential order (more precisely $O(|q|^n |T|)$). Therefore, the table storing decisions for each state has exponential entries. On the other hand in non-preemptive case, number of states are of order $O(|n| |q| |T|)$, so the decision storage is less. Considering the time complexity of the off-line algorithm, if n is the number of tasks, T is the total number of time units before deadline, q is the maximum number of discrete quality levels, and C is cardinality of control parameter set; the complexity of dynamic programming strategies are — (i) for non-preemptive case :: $O(|n| |q|^2 |T|^2 |C|)$, and (ii) for preemptive case :: $O(|q|^n |n| |q| |C| |T|^2)$.

Thus, we see the preemptive control is more time consuming and requires more storage. Also the context switching overheads are to be considered in case of preemptive scheduling. Still, if the number of problems to be solved are not many, then the preemptive strategy calculation and storage may not be a problem as long as the quality improvement over other strategies is significant.

3 Experimental Results

We perform experiments with task sets comprising of the traveling salesman problem (TSP). As stated earlier, we chose Anytime Simulated Annealing (anytime version of the traditional simulated annealing algorithm) as our base technique. The algorithm starts with an initial solution and then iteratively improves it by swapping edges (2-Opt) [6]. Rate of cooling is chosen as the control parameter. An adaptive cooling schedule is generated by selecting cooling rates dynamically from a given set. The approximation ratio is computed using

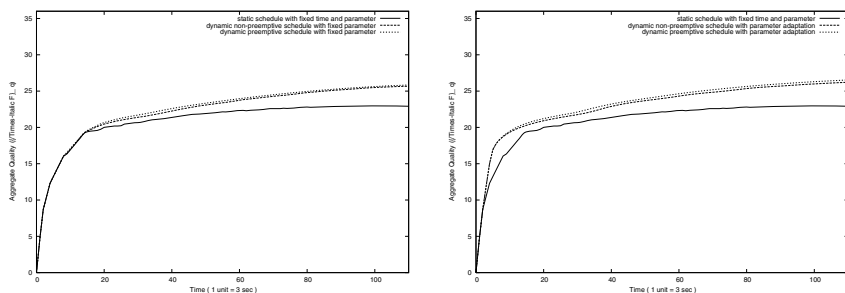
Table 1. Quality specification for TSP

Quality Level (q)	Approximation Ratio
6	1.005 - 1.000
5	1.010 - 1.005
4	1.015 - 1.010
3	1.030 - 1.015
2	1.050 - 1.030
1	1.100 - 1.050
0	∞ - 1.100

Eqn. 2. Quality discretization is taken to be uniform for all classes and shown in Table 1. Random instances of 20, 50, 100, 150 and 250 city problems are used for generating the DPPP. The time is discretized using steps of 3 seconds. We take minimal cooling rate to be 0.90, and then in the increments of 0.01 making a set of ten cooling rates. Results are generated for five problems each taken from a different class. Figure 1(a) shows the expected aggregate qualities of different time deliberation strategies (fixed cooling rate). In Figure 1(a) we include the results for strategies with parameter adaptation.

From the results, we observe that using dynamic time deliberation the expected aggregate quality is improved significantly. The average improvement is around 9.1% over time allocation ranging 15 – 110 units. The improvement with parameter adaptation is on an average 5 – 7% over fixed parameter time deliberation strategies. The difference of expected aggregate qualities using preemptive and non-preemptive is 2 – 6% depending on the time allocation. However, we did not consider the context switching requirements for preemptive scheduling which may nullify the improvement over the non-preemptive case.

From the experimental results, we conclude that dynamic time deliberation strategies improve the aggregate quality compared to the static deliberation. The quality can be further enhanced by controlling the base algorithm adaptively. Thus, we can infer that a combined strategy on deliberation and control for time-critical problems is expected to give better performance than a static strategy.



(a) Strategies with fixed parameter values (b) Strategies with adaptive parameter values

Fig. 1. Comparison of control strategies for traveling salesman problem set

4 Conclusions

In this paper, we have attempted to address the quality-time trade-off problem when a set of *hard* problems are to be solved within a given deadline. We have presented a model that uses runtime monitoring to generate a combined strategy for parameter control and time allocation. The strategies depict how dynamic adjustment of the time allocation of the individual problems (local adjustment) can improve the overall global quality. Although we have applied the technique with anytime simulated annealing algorithm, proposed framework can be used as a generic model for dynamic parameter adaptation through profile information for various classes of algorithms.

References

1. Boddy, M., Dean, T.: Deliberation scheduling for problem solving in time-constrained environments. *Artificial Intelligence* **67** (1994) 245–285
2. Dean, T., Boddy, M.: An analysis of time-dependent planning. In: *Proceedings of 6th National Conference on Artificial Intelligence (AAAI 88)*, St. Paul, MN, AAAI Press (1988) 49–54
3. Horvitz, E.J.: *Computation and action under bounded resources*. PhD thesis, Stanford University (1990)
4. Hansen, E.A., Zilberstein, S.: Monitoring and control of anytime algorithms: A dynamic programming approach. *Artificial Intelligence* **126** (2001) 139–157
5. Likhachev, M., Gordon, G.J., Thrun, S.: *Ara*: Anytime A* with provable bounds on sub-optimality*. In: *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA (2004)
6. van Laarhoven, P., Aarts, E.: *Simulated Annealing: Theory and Applications*. Kluwer, Dordrecht (1992)
7. Eiben, A.E., Hinterding, R., Michalewicz, Z.: Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation* **3** (1999) 124–141
8. Bertsekas, D.P.: *Dynamic Programming: Deterministic and Stochastic Models*. Prentice-Hall, Englewood Cliffs, NJ (1987)

An RLS-Based Natural Actor-Critic Algorithm for Locomotion of a Two-Linked Robot Arm

Jooyoung Park, Jongho Kim, and Daesung Kang

Department of Control and Instrumentation Engineering, Korea University,
Jochiwon, Chungnam 339-700, Korea
{parkj, oyeasw, mpkds}@korea.ac.kr

Abstract. Recently, actor-critic methods have drawn much interests in the area of reinforcement learning, and several algorithms have been studied along the line of the actor-critic strategy. This paper studies an actor-critic type algorithm utilizing the RLS(recursive least-squares) method, which is one of the most efficient techniques for adaptive signal processing, together with natural policy gradient. In the actor part of the studied algorithm, we follow the strategy of performing parameter update via the natural gradient method, while in its update for the critic part, the recursive least-squares method is employed in order to make the parameter estimation for the value functions more efficient. The studied algorithm was applied to locomotion of a two-linked robot arm, and showed better performance compared to the conventional stochastic gradient ascent algorithm.

1 Introduction

Recently, actor-critic methods have drawn much interests in the areas of reinforcement learning, and several algorithms have been studied along the line of the actor-critic strategy. In the actor-critic methods, a separate memory structure is used to explicitly represent the policy independent of the value functions, and the policy structure is known as the actor, because it is used to select actions, while the part handling estimated value functions is called the critic, because it criticizes the actions made by the actor [1]. Among the various implementations of the actor-critic algorithms, particularly pertinent to this paper is the results on the natural actor-critic algorithm [2], which show that the actor-critic algorithm using the natural policy gradient is significantly better than other algorithms belonging to greedy methods or vanilla policy gradient methods, and can be a promising route to develop a reinforcement learning method for truly high-dimensional state-action systems. In this paper, we address the problem of modifying the natural actor-critic algorithm of [2] toward the use of the RLS(recursive least-squares) method, which is one of the most efficient techniques for adaptive signal processing, to estimate the critic parameters recursively, and also apply the resulting algorithm, which will be called the RLS-based natural actor-critic algorithm in this paper, to locomotion of a two-linked robot arm. A previous work on the use of the RLS method for reinforcement learning is

the so-called RLS-TD(λ) [3], which shows how to convert existing reinforcement learning algorithms to recursively updated versions utilizing the RLS method.

The remaining parts of this paper are organized as follows: In Section 2, after briefly describing about the actor part of the natural actor-critic algorithm, we report on how the recursive least-squares method can be employed for the estimation of the critic parameters. Section 3 shows the applicability of the RLS-based natural actor-critic algorithm via an example dealing with locomotion of a two-linked robot arm. Finally, in Section 4, concluding remarks are given.

2 RLS-Based Natural Actor-Critic Method

In this paper, we consider a discounted reward reinforcement learning problem [1] with states $s \in \mathcal{S}$, actions $a \in \mathcal{A}$, rewards $r \in \mathfrak{R}$, and time steps $t \in \{0, 1, 2, \dots\}$, in which a learning agent interacts with an MDP(Markov decision process). As usual, the environment's dynamics are characterized by state transition probabilities $p(s'|s, a) \triangleq Pr\{s_{t+1} = s' | s_t = s, a_t = a\}$, and expected rewards $r(s, a) \triangleq E\{r_t | s_t = s, a_t = a\}$. The objective of the learning agent is to pursue a policy that can maximize the discounted sum of rewards $J(\pi) \triangleq E\{\sum_{k=0}^{\infty} \gamma^k r_k | s_0, \pi\}$, where $\gamma \in (0, 1)$ is the discount rate, r_k is the immediate reward observed after the state transition from state s_k to s_{k+1} , s_0 is a designated start state, and π denotes the policy from which actions are chosen. In general, the policy is described as a conditional probability $\pi(a|s) \triangleq Pr\{a_t = a | s_t = s\}$. Note that by introducing the state value function $V^\pi(s) \triangleq E\{\sum_{k=0}^{\infty} \gamma^k r_{t+k} | s_t = s, \pi\}$, the action value function $Q^\pi(s, a) \triangleq E\{\sum_{k=0}^{\infty} \gamma^k r_{t+k} | s_t = s, a_t = a, \pi\}$, together with the so-called discounted state distribution [4] $d^\pi(s) \triangleq \sum_{k=0}^{\infty} \gamma^k Pr\{s_k = s | s_0, \pi\}$, we can rewrite the objective function in the following form¹:

$$J(\pi) = V^\pi(s_0) = \sum_a \pi(a|s_0) Q^\pi(s_0, a) = \sum_s d^\pi(s) \sum_a \pi(a|s) r(s, a).$$

The real essence of the actor-critic methods is in using separate parametrized families for the policy distribution $\pi(a|s)$ and approximators for the value functions. In the following, we describe the actor part updated via the natural gradient method [2], and then address on how the recursive least-squares method can be employed for the estimation of its critic part parameters. The resultant algorithm will be applied to the locomotion of a two-linked robot arm later in this paper.

2.1 Actor

The main role of the actor is to generate actions via a parametrized family. At each state $s \in \mathcal{S}$, an action $a \in \mathcal{A}$ is drawn in accordance with the conditional distribution $\pi_\theta(a|s)$, where θ is the parameter vector characterizing the

¹ Dealing with continuous states and actions requires the corresponding summations changed into integral representation. Throughout this paper, the summation representation is used for notational simplicity.

distribution. Thus, the objective we seek to maximize can be written as follows: $J(\pi) = J(\theta) = \sum_s d^{\pi_\theta}(s) \sum_a \pi_\theta(a|s)r(s, a)$. One of the convenient strategies for seeking the best distribution parameter θ is to utilize the direction of $\nabla_\theta J(\theta)$, which is often called the policy gradient. Utilizing the famous policy gradient theorem [4], [5] along with the equality $\sum_a \nabla_\theta \pi_\theta(s, a) = 0$ for $\forall s \in \mathcal{S}$, the policy gradient can be written as follows:

$$\begin{aligned} \nabla_\theta J(\theta) &= \sum_s d^{\pi_\theta}(s) \sum_a \nabla_\theta \pi_\theta(a|s) Q^{\pi_\theta}(s, a) \\ &= \sum_s d^{\pi_\theta}(s) \sum_a \nabla_\theta \pi_\theta(a|s) (Q^{\pi_\theta}(s, a) - V^{\pi_\theta}(s)) \\ &= \sum_s d^{\pi_\theta}(s) \sum_a \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s) A^{\pi_\theta}(s, a), \end{aligned} \quad (1)$$

where $A^{\pi_\theta}(s, a) \triangleq Q^{\pi_\theta}(s, a) - V^{\pi_\theta}(s)$ is the advantage value function which gives the advantage of action a over the average value in a state s . According to a remarkable observation introduced in [4] and [5], the advantage value function $A^{\pi_\theta}(s, a)$ can be replaced by the so-called compatible approximator²

$$\tilde{A}_w(s, a) \triangleq \nabla_\theta \log \pi_\theta(a|s)^T w \quad (2)$$

without affecting the unbiasedness of the gradient estimate. Note that the compatible approximator $\tilde{A}_w(s, a)$ is linear with respect to the parameter vector w . Based on equations (1) and (2), we see that a desirable function form for an estimate of the policy gradient can be given as follows:

$$\begin{aligned} \nabla_\theta J(\theta) &= \sum_s d^{\pi_\theta}(s) \sum_a \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s) A^{\pi_\theta}(s, a) \\ &\approx \sum_s d^{\pi_\theta}(s) \sum_a \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s) \tilde{A}_w(s, a) = F(\theta)w, \end{aligned}$$

where $F(\theta) \triangleq \sum_s d^{\pi_\theta}(s) \sum_a \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s) \nabla_\theta \log \pi_\theta(a|s)^T$. Among a variety of powerful gradient based algorithms, one of the most efficient tools for updating the parameter vector θ of $\pi_\theta(a|s)$ would be the natural gradient method [6]. When an objective function is of the form $L(\theta) = \sum_s p(s)l(s, \theta)$, where $p(s)$ is a probability mass function, and its gradient $\nabla_\theta L(\theta)$ is given, the natural gradient of L , which is denoted by $\tilde{\nabla} L(\theta)$, equals the steepest ascent in a Riemannian space with respect to the Fisher information metric $G(\theta) \triangleq \sum_s p(s) \nabla_\theta \log p(s) \nabla_\theta \log p(s)^T$, *i.e.*, $\tilde{\nabla}_\theta L(\theta) = G^{-1}(\theta) \nabla_\theta L(\theta)$. In a recent remarkable paper of Peters *et al.* [2], it was shown that for the reinforcement learning problem, matrix $F(\theta)$ is exactly the same with the Fisher information matrix, thus the natural gradient of $J(\theta)$ can be estimated using the following: $\tilde{\nabla}_\theta J(\theta) = G^{-1}(\theta) \nabla J_\theta(\theta) \approx G^{-1}(\theta) F(\theta)w = w$. This result enables us to update the actor parameter θ via the following simple rule:

$$\theta \leftarrow \theta + \alpha \tilde{\nabla}_\theta J(\theta) \approx \theta + \alpha w, \quad (3)$$

where $\alpha > 0$ is the learning rate. Note that the actor part described above is that of the natural actor-critic algorithm in [2]. Also, note that in practical applications of the update rule (3), the use of upper bound for the magnitude of each actor parameter may be often desirable.

² In this paper, we assume that the policy distribution π_θ is such that the gradient $\nabla_\theta \log \pi_\theta(a|s)^T$ is well-defined.

2.2 Critic

As mentioned before, the essence of the actor-critic methods is in using separate parametrized families for the actor part which is represented by the policy distribution $\pi_\theta(a|s)$, and the critic part which is represented by value functions. For the parametrized families for the critic part, this paper employs the linear functions $\tilde{A}_w(s, a) \triangleq \nabla_\theta \log \pi_\theta(a|s)^T w$ and $\tilde{V}_v(s) \triangleq \phi^T(s)v$, which approximate the advantage value function $A^{\pi_\theta}(s, a)$ and the state value function $V^{\pi_\theta}(s)$ for action-generating policy π_θ , respectively. From the Bellman equation [1] $Q^{\pi_\theta}(s, a) = r(s, a) + \gamma \sum_{s'} p(s'|s, a)V^{\pi_\theta}(s')$, we see that through a sampled trajectory, $Q^{\pi_\theta k}(s_k, a_k)$ can be approximated by $r_k + \gamma V^{\pi_\theta k}(s_{k+1})$; thus $r_k + \gamma \tilde{V}_v(s_{k+1})$ is a valid estimate for the $Q^{\pi_\theta k}(s_k, a_k)$. Also from $Q^{\pi_\theta}(s, a) = A^{\pi_\theta}(s, a) + V^{\pi_\theta}(s)$ and the usual strategy using the eligibility trace [1], we see that in order for the approximators $\tilde{A}_w(s, a)$ and $\tilde{V}_v(s)$ to be useful in the t -th time step, it is desirable to minimize the following:

$$\begin{aligned} \tilde{\Psi}_t(v, w) \triangleq & \left\| \sum_{k=0}^t z_k [(\tilde{V}_v(s_k) + \tilde{A}_w(s_k, a_k)) - (r_k + \gamma \tilde{V}_v(s_{k+1}))] \right\|^2 = \\ & \left\| \sum_{k=0}^t z_k [\phi^T(s_k) - \gamma \phi^T(s_{k+1}), \nabla_\theta \log \pi(a_k|s_k)^T] \begin{bmatrix} v \\ w \end{bmatrix} - \sum_{k=0}^t z_k r_k \right\|^2, \end{aligned} \quad (4)$$

where z_k is the eligibility trace vector defined via

$$\begin{aligned} z_k &= \gamma \lambda z_{k-1} + [\phi^T(s_k), \nabla_\theta \log \pi(a_k|s_k)^T]^T \text{ for } k \geq 1, \\ z_0 &= [\phi^T(s_0), \nabla_\theta \log \pi(a_0|s_0)^T]^T, \end{aligned}$$

and $\lambda \in [0, 1]$ is the trace-decay parameter. Note that minimizing (4) is simply a least-squares problem utilizing the entire history of agent-environment interactions up to the t -th time step. When there is a need to put more emphasis on recent observations, the use of the so-called forgetting factor $\beta \in (0, 1)$ is desirable.

In this case, the following needs to be used instead of (4): $\tilde{\Psi}_t(v, w) \triangleq \|M_t \begin{bmatrix} v \\ w \end{bmatrix} - b_t\|^2$, where $M_t \triangleq \sum_{k=0}^t \beta^{t-k} z_k [\phi^T(s_k) - \gamma \phi^T(s_{k+1}), \nabla_\theta \log \pi(a_k|s_k)^T]$, and $b_t \triangleq \sum_{k=0}^t \beta^{t-k} z_k r_k$. Note that for $t \geq 1$, the above M_t and b_t can be written in the following recursive form: $M_t = \beta M_{t-1} + z_t [\phi^T(s_t) - \gamma \phi^T(s_{t+1}), \nabla_\theta \log \pi(a_t|s_t)^T]$, $b_t = \beta b_{t-1} + z_t r_t$. Also, note that when M_t is invertible, the optimal solution to the problem of minimizing $\tilde{\Psi}_t(v, w)$ is obviously $\begin{bmatrix} v \\ w \end{bmatrix} = M_t^{-1} b_t$. However, M_t is usually not invertible until a sufficient number of samples have been included in its summation. A common strategy used in the recursive least-squares method for ensuring the invertibility of M_t is to use δI for its initialization [3]. Employing the strategy, we use $M_0 = \delta I + z_0 [\phi^T(s_0) - \gamma \phi^T(s_1), \nabla_\theta \log \pi(a_0|s_0)^T]$, where δ is a positive number, instead of $M_0 = z_0 [\phi^T(s_0) - \gamma \phi^T(s_1), \nabla_\theta \log \pi(a_0|s_0)^T]$. Now, by applying the matrix inversion formula [7] $(A + XY)^{-1} = A^{-1} - A^{-1}X(I + YA^{-1}X)^{-1}YA^{-1}$ to the equation for M_t , we can obtain a recursive update rule for M_t , and also the following procedure for an approximate solution to minimizing $\tilde{\Psi}_t(v, w)$: Let

$$\begin{aligned}
z_0 &\triangleq [\phi^T(s_0), \nabla_\theta \log \pi(a_0|s_0)^T]^T, \\
M_0 &\triangleq \delta I + z_0[\phi^T(s_0) - \gamma\phi^T(s_1), \nabla_\theta \log \pi(a_0|s_0)^T], \\
M_t &\triangleq \beta M_{t-1} + z_t[\phi^T(s_t) - \gamma\phi^T(s_{t+1}), \nabla_\theta \log \pi(a_t|s_t)^T] \text{ for } t \geq 1, \\
P_t &\triangleq M_t^{-1} \text{ for } t \geq 0, \text{ and } K_t \triangleq P_t z_t \text{ for } t \geq 0.
\end{aligned}$$

Then with the update rules

$$\begin{aligned}
z_t &= \gamma\lambda z_{t-1} + [\phi^T(s_t), \nabla_\theta \log \pi(a_t|s_t)^T]^T, \\
P_t &= \frac{1}{\beta} \left(P_{t-1} - \frac{P_{t-1} z_t [\phi^T(s_t) - \gamma\phi^T(s_{t+1}), \nabla_\theta \log \pi(a_t|s_t)^T] P_{t-1}}{\beta + [\phi^T(s_t) - \gamma\phi^T(s_{t+1}), \nabla_\theta \log \pi(a_t|s_t)^T] P_{t-1} z_t} \right), \\
K_t &= \frac{P_{t-1} z_t}{\beta + [\phi^T(s_t) - \gamma\phi^T(s_{t+1}), \nabla_\theta \log \pi(a_t|s_t)^T] P_{t-1} z_t},
\end{aligned} \tag{5}$$

the solution for the critic parameters at time t can be obtained by the following recursive equation:

$$\begin{aligned}
\begin{bmatrix} v_t \\ w_t \end{bmatrix} &= \begin{bmatrix} v_{t-1} \\ w_{t-1} \end{bmatrix} \\
&+ K_t (r_t - [\phi^T(s_t) - \gamma\phi^T(s_{t+1}), \nabla_\theta \log \pi(a_t|s_t)^T] \begin{bmatrix} v_{t-1} \\ w_{t-1} \end{bmatrix}).
\end{aligned} \tag{6}$$

Note that the resultant recursive least-squares solution w_t will be used in the update process for the actor part via (3).

2.3 Algorithm

The algorithm considered in this paper repeats two tasks: An agent-environment interaction task in which the agent interacts with its environment with an action generated by the current policy and observes the consequence of the interaction, and a task for the estimation and improvement in which the agent optimizes its policy by updating the actor and critic parameters on the basis of the natural gradient and the recursive least-squares method. More precisely, the RLS-based natural actor-critic algorithm can be summarized as follows:

Given:

- Initial state s_0
- Parametrized policy $\pi_\theta(a|s)$ with its initial parameter vector $\theta = \theta_0$, and derivative $\nabla_\theta \log \pi_\theta(a|s)$
- Basis functions $\phi(s) \triangleq [\phi_1(s) \cdots \phi_K(s)]^T$ in use for $\tilde{V}_v(s) \triangleq \phi(s)^T v$, which approximates the state value function
- Learning rate $\alpha > 0$ for updating the actor parameter θ
- Forgetting factor $\beta \in (0, 1)$
- Discount rate $\gamma \in (0, 1)$
- Trace-decay parameter $\lambda \in [0, 1]$
- Constant $\delta > 0$
- Bound M for limiting the magnitude of each actor parameter

Goal:

- Parameter vectors θ_t which tend to that of a locally optimal policy distribution $\pi_{\theta}(a|s)$
- Parameter vectors v_t and w_t which yield good approximation for \tilde{V}_{v_t} and \tilde{A}_{w_t}

Algorithm:

for $t := 0, 1, 2, \dots$ **do**

Draw a control action a_t from the distribution $\pi_{\theta_t}(\cdot|s_t)$.

Perform a_t , and observe the reward r_t and the next state s_{t+1} .

Use the recursive least-squares rules (5) and (6) to find w_t and v_t .

Adjust the policy distribution parameters via $\theta_{t+1} = \theta_t + \alpha w_t$.

if any entry of θ_{t+1} is of magnitude bigger than M ,

then reduce it to M .

end

end

3 Locomotion of a Two-Linked Robot Arm

In this section, we address the application of the RLS-based natural actor-critic algorithm to an example of [8], which dealt with locomotion of a robot arm. This example considers a planar two-linked manipulator in a gravitational environment. The mission assigned to the robot is to move forward as fast as possible, without knowing the environment in advance. So, the agent needs to find out

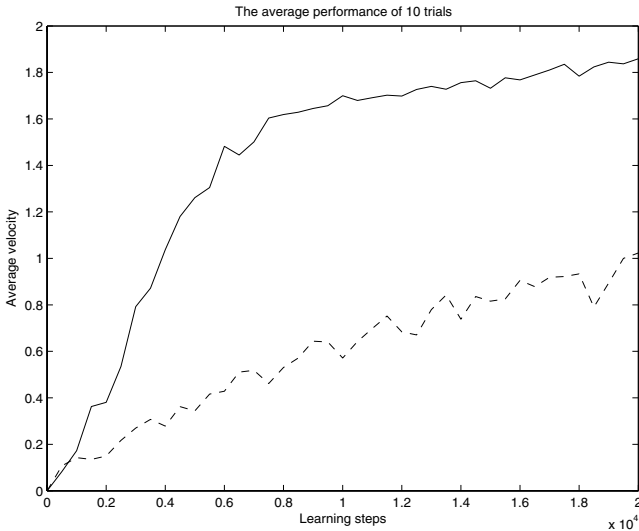


Fig. 1. The average performance of 10 trials. RLS-based natural actor-critic approach (solid), and the SGA approach of [8] (dashed).

an efficient policy based on the observed agent-environment interactions. The immediate reward for this problem is defined as the distance that the body of the robot moved forward in the current step. At the t -th time step, the agent reads the normalized joint angles, $x_1(t)$ and $x_2(t)$, and outputs a binary action value (+1 or -1) for each joint, which indicates turning direction of the joint motor, according to its stochastic policy. Here, the policy is represented by the logistic distribution function, i.e., $\pi_{\theta_t}(a_t = 1|s_t) = 1/(1 + \exp(-\theta_t^T s_t))$, where s_t is the state vector at the t -th time step defined by $s_t \triangleq [x_1(t), x_2(t), 1]^T$. The considered robot has the same specifications with [8], thus it satisfies the following:

- The upper arm length is 34 [cm].
- The fore arm length is 20 [cm].
- The joint of the body and the arm is located on the height = 18 [cm], the horizontal distance = 32 [cm] from the body's bottom left corner.
- The angle from the horizontal of the first joint connected to the upper arm is constrained such that $-4^\circ \leq \angle(Joint1) \leq 35^\circ$.
- The angle of the second joint from the axis of the upper arm to the fore arm is constrained such that $-120^\circ \leq \angle(Joint2) \leq 10^\circ$.
- The motor of each joint moves the arm to $12^\circ + \epsilon$ in the commanded direction, where ϵ is a random variable uniformly distributed on the interval $[-4^\circ, +4^\circ]$.
- When the arm is touching the ground, the arm does not slip while the body slips.

The proposed method was applied to the robot for 20,000 time steps with the following parameters:

- Initial state $s_0 = [0 \ 0 \ 1]^T$
- Initial policy distribution vector $\theta_0 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$
- Basis functions $\phi(s) \triangleq [x_1 \ x_2 \ 1]^T$
- Learning rate $\alpha = 0.0003$
- Forgetting factor $\beta = 0.99$
- Discount rate $\gamma = 0.99$
- Trace-decay parameter $\lambda = 0.5$
- Constant $\delta = 10$
- Bound for each actor parameter $M = 100$

The solid curve of Fig. 1 shows the average velocity of 10 trials resulting from the RLS-based natural actor-critic algorithm applied to the robot example. For comparison, we also performed simulations for the SGA(stochastic gradient ascent) method of [8]. Shown in the dashed curve of Fig. 1 is the average performance of 10 trials for the SGA method. Comparing these curves in the figure, we see that the RLS-based natural actor-critic algorithm gave significantly better results.

4 Concluding Remarks

In this paper, we studied on the RLS-based natural actor-critic algorithm, and applied it to locomotion of a two-linked robot arm. The natural policy gradient

and the recursive least-squares method are two key ingredients of the considered algorithm, and they are used in the process of updating the actor and critic parameters, respectively. Simulation results for the example dealing with locomotion of a two-linked robot arm showed that the algorithm considered in this paper yields better performance compared to the conventional SGA method. Further investigations yet to be done include extensive comparative studies which can reveal the strength and weakness of the considered method.

References

1. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA (1998)
2. Peters, J., Vijayakumar, S., Schaal, S.: Reinforcement learning for humanoid robotics. In: Proceedings of the Third IEEE-RAS International Conference on Humanoid Robots (2003)
3. Xu, X., He, H., Hu, D.: Efficient reinforcement learning using recursive least-squares methods. *Journal of Artificial Intelligent Research* **16** (2002) 259–292
4. Sutton, R.S., McAllester, D., Singh, S., Mansour, Y.: Policy gradient methods for reinforcement learning with function approximation. *Advances in Neural Information Processing Systems* **12** (2000) 1057–1063
5. Konda, V., Tsitsiklis, J.N.: Actor-Critic Algorithms. *SIAM Journal on Control and Optimization* **42(4)** (2003) 1143–1166
6. Amari, S.: Natural gradient works efficiently in learning. *Neural Computation* **10(2)** (1998) 251–276
7. Moon, T.K., Stirling, W.C.: *Mathematical Methods and Algorithm for Signal Processing*. Prentice Hall, Upper Saddle River, NJ (2000)
8. Kimura, H., Miyazaki, K., Kobayashi, S.: Reinforcement learning in POMDPs with function approximation. In: Fisher, D.H. (ed.): *Proceedings of the Fourteenth International Conference on Machine Learning* (1997) 152–160

Dynamic Clustering Using Multi-objective Evolutionary Algorithm

Enhong Chen and Feng Wang

Department of Computer Science and Technology,
University of Science and Technology of China, Hefei 230027, China
cheneh@ustc.edu.cn, fwang83@mail.ustc.edu.cn

Abstract. A new dynamic clustering method using multi-objective evolutionary algorithm is proposed. As opposed to the traditional static clustering algorithms, our method implements variable length chromosome which allows the algorithm to search for both optimal cluster center positions and cluster number. Thus the cluster number is optimized during run time dynamically instead of being pre-specified as a parameter. We also introduce two complementary objective functions--compactness and connectedness instead of one single objective. To optimize the two measures simultaneously, the NSGA-II, a highly efficient multi-objective evolutionary algorithm, is adapted for the clustering problem. The simultaneous optimization of these objectives improves the quality of the resulting clustering of problems with different data properties. At last, we apply our algorithm on several real data sets from the UCI machine learning repository and obtain good results.

1 Introduction

Usually, it is difficult for a clustering algorithm to get a good result in practice. The problem of cluster optimization is twofold: optimization of (1) cluster centers and (2) number of clusters. The latter aspect has often been neglected as most approaches typically specify the number of clusters as a parameter. Since the structure of the data is completely unknown, it takes much time to fix the appropriate number. Furthermore, for many data sets, no unambiguous partition exists. Even if it is possible to fix an appropriate cluster number, algorithm may also fail. Because most existing clustering algorithms estimate the quality of a partition by means of just one internal evaluation function, such as the spatial separation between clusters or the compactness of clusters. But this may not be reliable for certain data sets.

This paper attempts to present a dynamic clustering algorithm using multi-objective evolutionary algorithm (MOEA). To overcome the shortcoming of pre-specifying the number of clusters as a parameter, we implements variable length chromosomes which allows the algorithm to effectively search for both optimal cluster centers and cluster number. Since the cluster number is optimized during run time instead of pre-specified, such clustering is referred to as *dynamic*. The use of multi-objective optimization may help us to overcome some of the disadvantages of current clustering algorithms. By using several complementary objective functions, the simultaneous optimization of these objectives may improve the clustering quality of data with different properties.

The rest of the paper is organized as follows. Section 2 briefly summarizes the related work on clustering and evolutionary algorithms. This is followed by a description of a MOEA--NSGA-II in section 3. In section 4 we will describe the detail of our proposed method. Then we will show experiment results on several real data sets from the UCI machine learning repository in section 5. Finally, in section 6 some conclusion and future work are presented.

2 Related Work

Here we discuss a different category of clustering algorithms based on the clustering criterion. In this way clustering algorithms can be classified into three groups. The first category tends to keep intra-cluster variation small to get compact clusters, which are effective for spherical and well-separated clusters but may fail to more complicated cluster structures. This category includes k-means, average-link agglomerative algorithm. The second category is based on the concept of connectedness. Neighboring elements should share the same cluster. Single-link agglomerative and density-based algorithms are of this type. These algorithms are well-suited to detect clusters of arbitrary shapes but they lack robustness when there is little spatial separation between clusters. The third category is based on the criteria of spatial separation. However, it gives little guidance during the clustering process and can easily lead to trivial solutions.

Previous methods have been limited to the single objective case. Generally, criteria based on cluster compactness have been employed because they provide smooth incremental guidance in the search space, which is much better than objectives based on spatial separation between clusters. It may fail to find very obvious cluster structure in some data sets. Because variance is only an approximation for a fuzzy concept of true cluster structure and most clustering algorithms are not robust to variations in certain cluster shape, size and dimensionality. So combination of different clustering results by means of ensemble methods [1] has been studied. In these methods, different clustering results are retrieved by repeatedly running the same algorithm or several complementary algorithms. Clustering ensembles are more robust and yield higher quality results, but they are limited to the solutions returned by the individual algorithms and can not explore the trade-off solutions.

On the other hand, traditional approaches implement fixed length coding schemes that require the pre-specification of the number of the clusters. They must be repeated for cluster numbers ranging from a very small number to the number of data points, which is often an extremely time consuming process. As opposed to the static clustering, several dynamic clustering methods have been presented. A. Ghozeil [2] introduces a dynamic clustering algorithm based on EP. Its Results are promising but the number of clusters is relatively small (≤ 5). R.Gorunescu [3] also presents a dynamic evolutionary clustering method based on the incremental clustering. But the method may be not efficient enough when the data set is large.

Our proposed approach optimizes different objectives explicitly in one clustering algorithm, enabling different tradeoffs to be explored during the clustering process. It also uses variable length chromosomes to search for both optimal cluster centers and cluster number. Thus it clusters the data set dynamically.

3 NSGA-II

In recent years, a number of multi-objective evolutionary algorithms have been proposed [4]. Among these, the NSGA-II is an effective algorithm with low computational complexity and good convergence. We will briefly describe it as follows:

In the fast non-dominated sorting procedure, each solution p is assigned two entities: (1) n_p , the number of solutions which dominate p ; (2) S_p , a set of solutions that p dominates. All solutions in the first non-dominated front will have their $n_p = 0$. For each p with $n_p = 0$, we visit each member q of S_p and reduce n_q by 1. For any q , if n_q becomes 0, then q belongs to the second non-dominated front. The above procedure is continued until all fronts are identified. It also presents a density-estimation metric and a crowded-comparison operator. It sorts the population according to each objective value in ascending order. For each objective function, the density of a solution p is defined as the distance between the two adjacent solutions of p . The overall crowding-distance value is calculated as the sum of individual distance values corresponding to each objective.

Initially a random parent population P_0 is generated. Q_0 is its offspring population. $R_t = P_t \cup Q_t$ is formed. Then R_t is sorted according to non-domination and we chose solutions from the first non-dominated front and the second, third and so on, until the size of solutions selected is up to N to form P_{t+1} . Here we note that if m is the last non-dominated front selected and there are already M solutions chosen for P_{t+1} . Then we should sort the solutions of the m th front using the crowded-comparison operation in descending order and choose the best $N-M$ solutions. Further details on NSGA-II can be referred to [5].

4 Dynamic Clustering Algorithm Using MOEA

To apply NSGA-II to the clustering problem, we need only to choose a suitable genetic encoding scheme, some genetic variation operators, and two or more objective functions. Now we will describe the detail of our method below.

4.1 Coding Scheme

A variety of different EA representations for clustering solutions have been explored in the literature, such as the straightforward encoding and a more complex one--Falkenauer's grouping EA [6]. But these direct encoding methods fail to significantly reduce the size of the clustering search space. Here we use an indirect coding scheme based on the extended chromosome representation introduced in [7]. It codes for cluster center only. We implements variable length chromosomes in our algorithm. Thus, cluster number gives rise to a variable complexity search.

Since our method is developed to solve numerical data clustering problems and each gene represents a cluster center, a gene can be denoted as (x_1, x_2, \dots, x_m) for an m -dimensional data point. In the chromosome, the genes are ordered by ascending x_1 value (the first dimension of the data point), which is convenient for the crossover operator described below. Then a chromosome with three genes can be denoted as $[(x_{11}, x_{12}, \dots, x_{1m}), (x_{21}, x_{22}, \dots, x_{2m}), (x_{31}, x_{32}, \dots, x_{3m})]$ with $x_{11} \leq x_{21} \leq x_{31}$.

4.2 Initialization

The initial population will be made up of chromosomes with randomly-selected length. For each population member, we randomly choose the chromosome length n uniformly from a range $[a, b]$. Then we randomly select n points for cluster centers within the data bounds. While the range of cluster numbers is specified a priori, there are no restrictions on the evolved number of clusters since the search operators, to be discussed subsequently, are able to adapt cluster number. A conservative estimate of the cluster number range has been chosen in this case.

4.3 Objective Functions

We are interested in selecting optimization criteria that reflect fundamentally different aspects of a good clustering solution. Here we select two types of complementary objectives. One is based on compactness, and the other one is based on connectedness of clusters. We refrain to use a third objective based on spatial separation, as the concept of spatial separation is opposite to that of connectedness of clusters.

In order to express the cluster compactness we compute the overall deviation of a partitioning. This is simply computed as the overall summed distances between data points and their corresponding cluster centers and should be minimized:

$$Deviation(C) = \sum_{i=1}^n \sum_{j=1}^{m_i} d(c_i, x_j^i), \tag{1}$$

where n is the number of clusters, c_i is the i th cluster center, m_i is the number of data points belonging to the i th cluster, x_j^i is the j th data point belonging to the i th cluster, and $d(a, b)$ is the Euclidian distance between points a and b .

To reflect cluster connectedness, we use a measure called connectivity, which is conceptually similar to the criterion of nearest-neighbor consistency introduced by Ding et al. [8], to evaluate the degree to which neighboring data points have been placed in the same cluster. It is computed as follows:

$$Connectivity(C) = \frac{1}{N} \sum_{i=1}^N \left(\frac{\sum_{j=1}^h x_{i,nn_i(j)}}{h} \right) \text{ with } x_{m,n} = \begin{cases} 1, & \text{if } \exists C_k : m, n \in C_k \\ 0, & \text{otherwise} \end{cases}, \tag{2}$$

where $nn_i(j)$ is the j th nearest neighbor of data point i ; h is a parameter determining the number of neighbors that contribute to the connectivity measure. The value of

connectivity ranges from 0 to 1 and should be maximized. Connectivity is insensitive to the shape of the data set. The nearest neighbor list can be pre-computed.

A further important consideration in the choice of these two objective functions is their potential to balance each other's tendency to increase or decrease the number of clusters. While the objective value associated with overall deviation necessarily improves with an increasing number of clusters, the opposite is the case for connectivity.

4.4 Selection and Variation Operators

We will use two selection strategies. The first one is the ordinary binary tournament selection. It is used to select parents from P_t for crossover and mutation to form the Q_t as described in section 3.2. The other one is the selection strategy based on NSGA-II.

The crossover operator is used as the length changing operator. In canonical two point crossover, a range of genes is selected over which to swap. This remains the same here except that the range is given by a range of x_1 value (the first dimension of the data point) over which to swap. If two parents have a different number of genes within the range, both chromosomes will exhibit length changes after crossover. When only a single gene is swapped between chromosomes, a simultaneous insertion and deletion occurs.

The standard Gaussian mutation with zero mean and standard deviation $\sigma_1, \sigma_2, \dots, \sigma_m$ is chosen as our mutation operator. And $\sigma_1, \sigma_2, \dots, \sigma_m$ are the self-adaptive mutation parameters which can be adjusted as in [9].

4.5 Selection Strategy Based on NSGA-II

After selecting parents from P_t and implementing crossover and mutation operators on them to form the Q_t , we will use the selection strategy of NSGA-II described in section 3.

4.6 Termination Criterion

The evolutionary algorithm is terminated when the number of iterations exceeds 200. In most case, after 200 generations, the self-adaptive mutation parameters are so small that significant improvements in the objective functions will no longer occur.

5 Experiment and Analysis

We conducted our experiments on four real data sets from the UCI machine learning repository [10], which are described in Table 1. We chose these data because they exhibit a great diversity in dimensionality, cluster number and shape, etc, which may demonstrate the conclusions that we draw from our algorithm.

Table 1. Summary of our data sets

Name	Total	Number of each cluster	Dimension	Clusters
Iris	150	50,50,50	4	2
Wisconsin	699	458,241	9	3
Yeast	1484	463,429,244,163,51,44,37,30,20,5	8	10
Wine	178	59,71,48	13	3

We will compare the result of our algorithm with two objectives against the classical k-means algorithm, and also against with two single-objective versions of our algorithm. The single-objective version uses the compactness and connectedness separately, preserves the genetic operator and throws off the NSGA-II algorithm.

In our experiments, the population size is set to 100. The crossover rate P_c is 0.7, and the mutation rate P_m is 0.05. The range of $[a, b]$ in initialization is $[2, 15]$, and we set h to 10 for the connectivity objective function. We evaluate solutions of different algorithms using the F-measure. It is an external evaluation function that requires knowledge of the correct class labels and compares a generated clustering solution to the known, real class memberships. This measure is limited to the range $[0, 1]$, where 1 reflects a perfect agreement with the correct partitioning. For further details on F-measure, the reader is referred to [11]. We run each algorithm several times to get the distribution of the results and represent them as columniations in Figure 1.

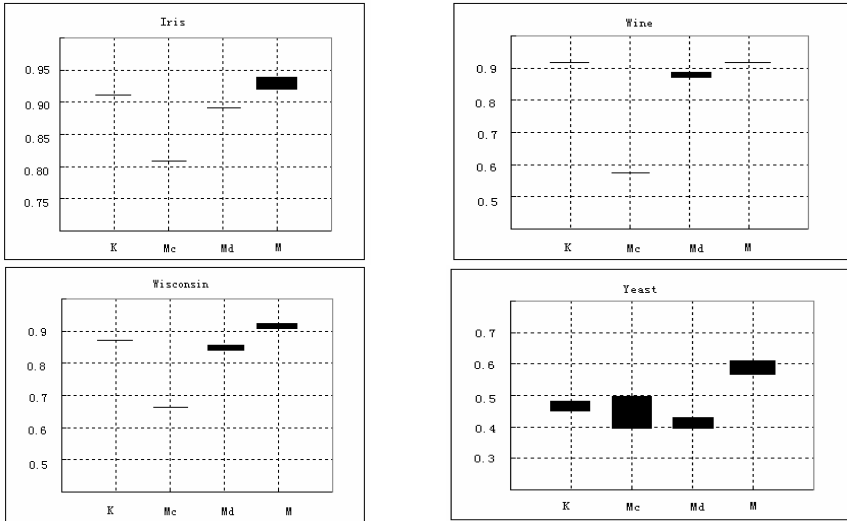


Fig. 1. Values of F-measure of different algorithms. K= k-means algorithm, Mc= the single-objective version of connectedness, Md= the single-objective version of and deviation. M=our multi-objective algorithm.

We can see from Figure 1 that the result of our proposed method is better than others. All of the other three algorithms may perform badly on particular data sets since their objective functions may not be suitable for the properties of these data sets. While our method exhibits a far more robust performance because of its use of multi-objective functions.

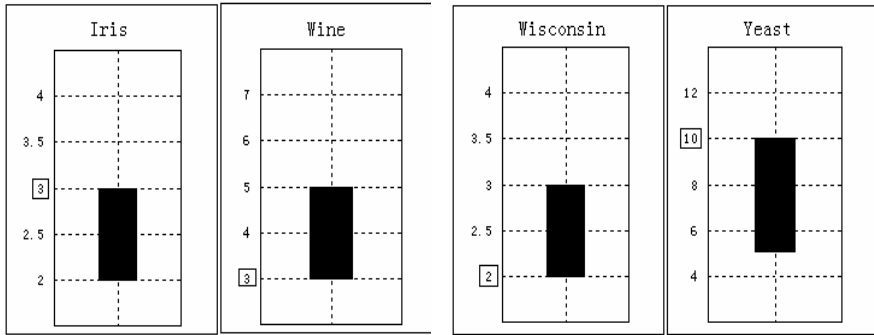


Fig. 2. The number of clusters obtained through our algorithm

We also show the number of result clusters of our algorithm. Its distribution is shown in Figure 2. Since in our method, the number of cluster is not pre-specified but automatically determined, the results are satisfying.

6 Conclusions and Future Work

In this paper, we propose a new method to cluster the data sets dynamically using multi-objective functions. In future work, we will make much effort to search for a more suitable objective function of compactness instead of the overall deviation for clustering problems having different number of clusters. Since if the overall deviation is considered as a function of cluster number, it can always be decreased by adding a data point as a cluster center; and fitness is a monotonically decreasing function of cluster number.

Acknowledgements

This work is supported by the Nature Science Foundation of Anhui Province (No.050420305), National Natural Science Foundation of China (No.60005004) and 973 Program of China (No. 2003CB317002).

References

1. M.H.C Law: Multi-objective data clustering. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC (2004) 424-430
2. A. Ghozeil and D.B. Fogel: Discovering patterns in spatial data using evolutionary programming. In Genetic Programming 1996: Proceedings of the first annual conference, Cambridge, MA (1996) 512-520
3. R.Gorunescu and D.Dumitrescu: Evolutionary clustering using an incremental technique. Studia Univ. Babeş-Bolyai, Informatica, Volume XLVIII (2003)
4. K. Deb: Multi-objective evolutionary algorithms: Introducing bias among Pareto-optimal solutions. Springer-Verlag, London, UK (2003) 263-292

5. Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan: A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, VOL 6, NO. 2 (2002) 182-197
6. E. Falkenauer: *Genetic Algorithms and Grouping Problems*. John Wiley & Son Ltd, New York, NY (1998)
7. C.-Y. Lee and E.K. Antonsson: Variable length genomes for evolutionary algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference*. Las Vegas, NV, Morgan Kaufmann (2000) 806
8. Ch. Ding and X. He: K-nearest-neighbor consistency in data clustering: incorporating local information into global optimization. In *Proceedings of the 2004 ACM Symposium on Applied Computing*. New York, NY (2004) 584-589
9. H.-P. Schwefel: *Evolution and optimum seeking*. John Wiley, New York (1995)
10. C. Blake and C. Merz: *UCI repository of machine learning database*. Technical report, Department of Information and Computer Science, University of California, Irvine (1998) <http://www.ics.uci.edu/~mlern/MLRepository.html>
11. C. van Rijsbergen: *Information Retrieval*, 2nd edition. Butterworths, London, UK (1979)

Multimodal FeedForward Self-organizing Maps

Andrew P. Papliński¹ and Lennart Gustafsson²

¹ Clayton School of Information Technology,
Monash University, Victoria 3800, Australia
`app@csse.monash.edu.au`

² Computer Science and Electrical Engineering,
Luleå University of Technology, S-971 87 Luleå, Sweden
`Lennart.Gustafsson@luth.se`

Abstract. We introduce a novel system of interconnected Self-Organizing Maps that can be used to build feedforward and recurrent networks of maps. Prime application of interconnected maps is in modelling systems that operate with multimodal data as for example in visual and auditory cortices and multimodal association areas in cortex. A detailed example of animal categorization in which the feedforward network of self-organizing maps is employed is presented. In the example we operate with 18-dimensional data projected up on the 19-dimensional hyper-sphere so that the “dot-product” learning law can be used. One potential benefit of the multimodal map is that it allows a rich structure of parallel unimodal processing with many maps involved, followed by convergence into multimodal maps. More complex stimuli can therefore be processed without a growing map size.

1 Introduction and Motivation

We present a generalization of Kohonen self-organizing maps [1] that allows us to build feedforward and feedback structures consisting of interconnected self-organizing maps. This work has originated from our attempt to model deficits in learning, caused by attention abnormalities in autism [2, 3, 4, 5, 6, 7], therefore is inclined towards modelling functions of brain. However, the results presented are more general and can be applied to interpretation of any multimodal data.

Self-organizing maps (SOMs) and their applications are very popular topic to study. The search of *IEEEexplore* shows that since 2000 75 papers has been published in IEEE periodicals and another 528 in IEEE conference proceedings. A significant number of papers have been also published outside IEEE. Most of the publications are devoted to a variety of applications using a variant of the basic Kohonen algorithm for a single SOM. In [1] Kohonen discussed possible variations of SOMs that include: maps of varying topology that has been studied, for example, in [8, 9], and tree-structured SOMs to improve the winner search procedure, e.g. [10]. The systems of interconnected SOMs that our paper is preoccupied with have not been to our knowledge studied yet and is designated “for future research” in [1] p.194.

There are also self-organizing systems of four hierarchically organized maps, *VisNet* and *VisNet2* [11, 12]. These systems, designed to capture some essential characteristics of the visual system, achieve learning through variants of Hebb's learning rule and ordering of the maps by lateral connections. These systems, while in important aspects more biological than the Kohonen maps presented in this paper, are not multimodal.

1.1 An Introduction to the Modular Architecture of Mammalian Sensory Processing

Organization of the mammalian neocortex is modular and hierarchical. A columnar structure, consisting of minicolumns and macrocolumns, comprising in the order of fifty to a hundred and a number of thousand neurons respectively, is ubiquitous [13]. On a higher aggregational scale sensory processing of different modalities is done in visual cortex, auditory cortex etc. with a number of areas, hierarchically arranged, in each (the human visual cortex has upwards of thirty areas), see e.g. [14]. These areas are primary areas (visual area V1, auditory area A1, etc.) and several levels of higher unimodal association areas. The primary areas are organized retinotopically, tonotopically etc., each neuron having a small receptive field. Higher association areas process information of a more specific kind, e.g. face recognition, each neuron having a large receptive field. For an introduction to the organization of neocortex see [15].

The stimuli presented to our nervous system are not restricted to one modality however, they are generally bimodal or multimodal. We see and hear, we see and feel, we smell and taste. If we hear an automobile crash it will catch our attention and guide our vision, through a saccade or a head movement or both, to the scene of the accident. While lower level processing of stimuli is largely unimodal, multimodal percepts are formed when the results of the unimodal processing are combined in multimodal areas in, e.g. the orbitofrontal cortex, see e.g. [16]. There are different kinds of connections between and within areas. Between areas in a hierarchy there are bottom up connections but also top down connections. There are lateral connections between areas on the same level in a hierarchy and within areas there are recurrent feedback connections. The areas may be seen as parallel processors, interchanging processing results.

The modular architecture of cortex may be formed from such biological restrictions as upper limits of a neuron's connectivity. It may also offer a processing speed advantageous to process sensory stimuli in a highly modular way with processing in all modules running in parallel. The hierarchical structure of cortex may even offer a specific advantage to humans. There is evidence that synaptogenesis is heterochronous in human neocortex as opposed to synaptogenesis in rhesus monkeys ([17, 18]). In humans the maximum density of synapses is reached earlier in sensory cortices (earlier in visual cortex than in auditory cortex) than in prefrontal cortex, in contrast to the development in rhesus monkeys where synaptogenesis is concurrent in all neocortical areas. Successive organization of areas as they appear in the sensory processing hierarchy may well be advantageous.

2 The Structure of the Feedforward Self-organizing Maps

Neural networks have been inspired by the possibility of achieving information processing in ways that resemble those of biological neural systems. Even though some neural network architectures and learning rules, such as the multilevel perceptrons and the backpropagation error correcting learning algorithms have developed into independent signal processing systems, others such as pattern associators based on Hebbian learning [19] and self-organizing networks [1] show considerable similarities with biological neural systems. It has been shown that pattern associators may well simulate the multimodal sensory processing in cortex [16].

The purpose of this paper is to show that the Kohonen networks, contrary to popular belief, lend themselves very well to building networks in a modular and multimodal way. Simulation of such modular networks consisting of interconnected Kohonen networks are suitable for execution on parallel processors. Also, a large number of interconnected Kohonen networks may serve as a simulation model for biological sensory processing.

Kohonen Self-Organizing Maps (SOMs) [1] are well-recognized and efficient tools for mapping multidimensional stimuli onto a low dimensionality (typically 2) neuronal lattice. Example of such a map is given in Fig. 1.

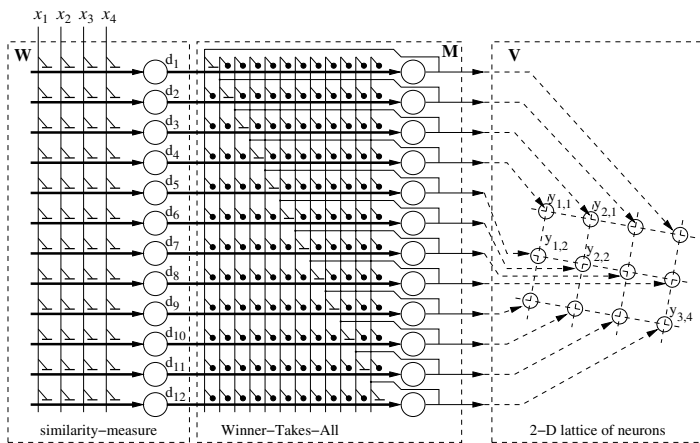


Fig. 1. A SOM mapping a 4-dimensional stimuli onto a 2-dimensional 3×4 neuronal lattice

For our purposes we can identify a similarity layer, a competitive layer based on lateral inhibition and local self-excitatory feedback and a mapping mechanism to the output layer. We can say that for a given matrix of neuronal weights, W , the n th stimulus $\mathbf{x}(n)$ is mapped into a position of the **winner** $\mathbf{v}(n)$, that is, the neuron with the weight vector most similar to the stimulus.

$$\mathbf{v}(n) = g(\mathbf{x}(n); W) ; \mathbf{v} \in \mathbb{R}^l \tag{1}$$

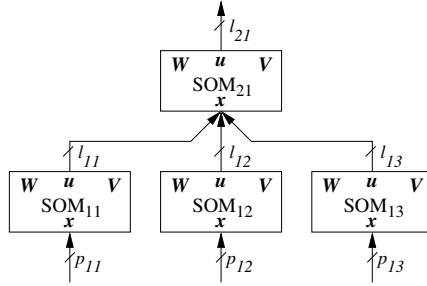


Fig. 2. A simple feedforward structure of self-organizing maps built from self-organizing blocks.

where l is a dimensionality of the neuronal grid. The self-organizing building block can now be used to form feedforward and feedback structures. In this paper we concentrate on the feedforward networks as in example in Fig. 2.

The bottom level maps, $SOM_{1,k}$, received sensory stimuli $\mathbf{x}_{1,k}$ of different dimensionality $p_{1,k}$ representing different modalities of afferent signals. Dimensionality of each map is, in general, equal to $l_{1,k}$, but is, most typically, equal to 2 for the ease of visualisation. The higher level map, SOM_{21} , received the re-coded stimuli from each lower level map of the total dimensionality $p_{21} = \sum_{k=1}^K l_{1,k}$, e.g. 6 for the example of Fig. 2. Each individual map, e.g. the one from Fig. 1, consists of $M = m_1 \times m_2 \dots \times m_l$ neurons, where m_i is the map size in the i th dimension. In a feedforward structure as in Fig. 2 learning takes place concurrently for each SOM, according to the well-known Kohonen learning law.

In the practical example presented below we work with normalised stimuli and activity data, therefore we use the simple “dot-product” learning law [1, 5]. In this case the update of the weight vector for the j th neuron is described by the following expression:

$$\Delta \mathbf{w}_j = \eta \cdot \Lambda_j \cdot (\mathbf{x}^T - d_j \cdot \mathbf{w}_j) ; \quad d_j = \mathbf{w}_j \cdot \mathbf{x} \quad (2)$$

where Λ_j is a neighbourhood function, Gaussian in our case, centred on the position of the winning neuron, and d_j is the post-synaptic activity of the j th neuron. It is easy to show that for the above learning law, if the stimuli \mathbf{x} are on a unity hyper-sphere, the resulting weight vectors \mathbf{w} are located on, or close to such a sphere.

The higher level map learns from the combined centres of activity produced by the lower level maps. During testing, a multimodal stimulus is applied and for each i, k map on every level, we can record the position of the winning neuron, $\mathbf{v}_{i,k}$.

3 Example of Mapping Multi-modal Stimuli

As an illustrative example we consider a well-known problem of categorization of animals. Our animal kingdom that we have also used in our work on modelling

autism [5, 4, 3] consists of 32 animals each characterized by 18-dimensional data. Half of the animals are variety of cats. The animals are listed below sorted according to their weight:

Grey whale, Hippopotamus, White rhinoceros, Kodiak bear, Polar bear, Grevy's zebra, Przewalski's horse, Tiger, Lion, Anaconda, Jaguar, Puma or cougar, Panther, Leopard, Snow Leopard, Canis lupus (Wolf), Atlantic salmon, Cheetah, Grey western kangaroo, Eurasian lynx, Rainbow trout, Dingo, Swamp wallaby, Serval, Ocelot, Fishing cat, Mute (white) swan, Black swan, Domestic cat (even coloured), Domestic cat (striped), Domestic cat (black), Siamese cat.

The features chosen to characterize these animals are as follows: $x_1 = \log(\text{weight})$; $x_2 \in \{1, 2, 3\}$ – food (herbivores, omnivores, carnivores); x_3, x_4, x_5, x_6 (binary) – locomotion (fins, wings, two legs, four legs); x_7 (binary) – equipped with hooves (perissodactyls) or cloven hooves (artiodactyls); x_8 (binary) – equipped with claws; x_9 (binary) – equipped with other feet; $x_{10} \in \{1, 2, 3\}$ – cover (fur, feathers and scales); x_{11} (binary) – colour black; x_{12} (binary) – colour white; x_{13} (binary) – even coloured; x_{14} (binary) – spotted; $x_{15} \in \{0, 1/4\}$ (binary) – striped; $x_{16} \in \{2, 4\}$ – facial feature (short faced, long faced); $x_{17} \in \{1, 2, 3\}$ – aquatic; $x_{18} \in \{1, 2, 3\}$ – social behaviour (single living, pair living, group living). In addition, for testing of the generalization of the multi-map structure we use two unusual animals, one being a domestic cat weighing two tonnes ('catWhale'), the other, *Andrusarchus mongoliensis*, is an extinct cloven-hoofed 1-tonne carnivour.

The features by which we describe our animals are of course multimodal and some, like behaviour, represent semantic memory data rather than immediate sensory stimuli. This example does not imply that a biological neural system would treat the total data in three primary unimodal and one multimodal map. Rather the example illustrates the functioning of a very small multiple map system. The 18-dimensional vector representing an animal is split into three sub-vectors, representing three modalities: $\mathbf{x}_{1,1} = x_1$ (weight), $\mathbf{x}_{1,2} = (x_2 \dots x_9)$ (food and locomotion), $\mathbf{x}_{1,3} = (x_{10} \dots x_{18})$ (coloration, facial features, behaviour). Single modality stimuli of dimensionality $p_{11} = 1, p_{12} = 8, p_{13} = 9$, excite three first-level maps as shown in Fig. 2. In the example below we use 16 neurons in each sensory map organized on a 1×16 grid for the (1,1) map and 4×4 grids for (1,2) and (2,3) maps. The second level map has 36 neurons organized on a 6×6 grid. The result of learning can look like in Fig. 3. The numbers that follows the animal's name is a relative distance between the animal and the closest neuron. In well-developed maps most of these distances are expected to be zero.

The organization of the three unimodal maps justifies some comments. The **first modality** map, representing weights, is a one-dimensional map with the animals simply arranged in descending weight order from left to right. The neuron which is the best fit for a particular animal has an oval shape with the name of the animal written in. Since there are thirty-two animals and the map was provided with only sixteen neurons, in some cases several animals have to share one neuron for its best fit. The **second modality** map represents food and locomotion. Therefore it is natural that salmon, trout and whale share one neuron, whereas almost all land predators share one neuron. The other animals can be

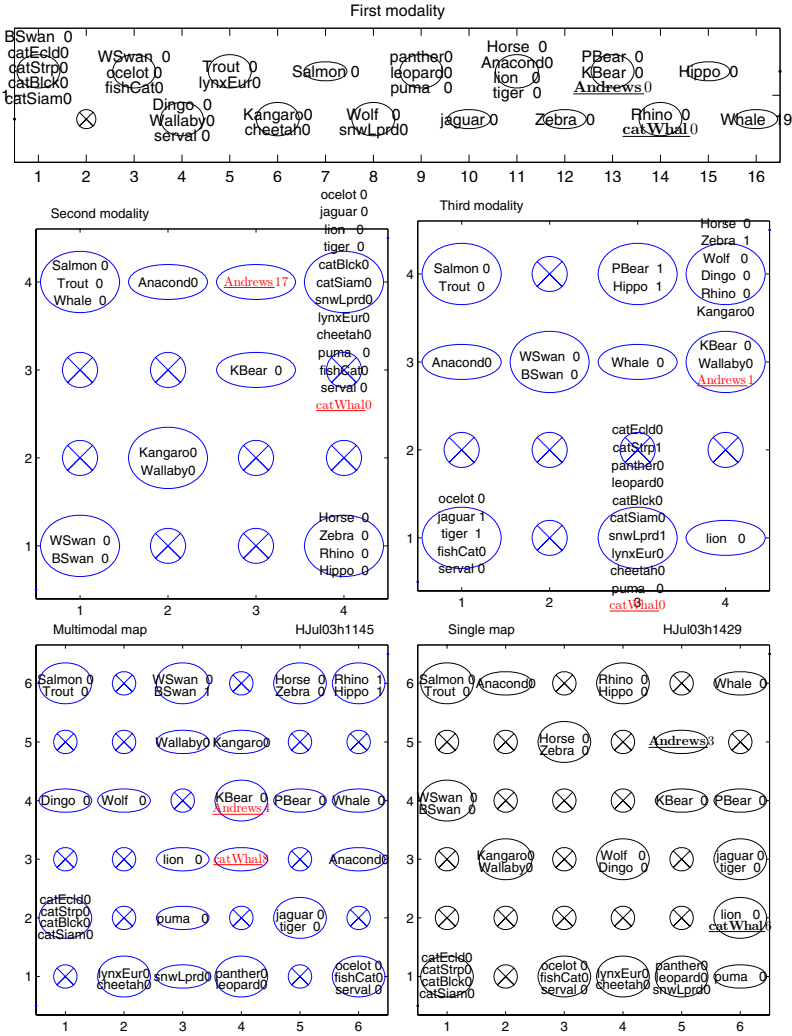


Fig. 3. Categorization of animals with the feedforward multimodal maps and a conventional single map

commented in similar ways. The **third modality** map represents coloration, facial features and behaviour. This map results in more discrimination than the second modality map. The cats which forms a tight knit group in the second modality are here split onto three neurons, cats which appreciate water, cats which do not and the lion which lives in larger groups. The Kodiak bear and the wallaby might seem an odd pair, but they both have very elongated faces, they are both even colored and they both are solitary animals. The kangaroo lives in groups and is therefore grouped with zebras, wolves and other social animals. The clustering of animals seen in the third modality map may at first seem absurd but it quite correctly reflects the features that were chosen for this map.

In the multimodal map we see that the animals are represented in a way which we find natural. Some animals like the whale and the anaconda are solitary in the multimodal map in all experiments with these data, simply because no other animals in our animal kingdom resemble them. Other animals form pairs and then they are very similar, as described by the features we chose. The dingo and the wolf are solitary in this map while in other experiments they appear as a pair. The same is true for the wallaby and the kangaroo, and the Kodiak bear and the polar bear. The cats are more often than not grouped with other cats, usually distinguished from other groups of cats mainly by weight. The four house cats which is one species with different coloration are grouped together here and this is the case for all experiments with maps of this size. The lion often has its own neuron, in some experiments it shares one neuron with the tiger and the jaguar. There are quite a number of neurons which are not best fits for any animal. This is in the nature of Kohonen maps. The two animals that were presented to the map without any prior learning, the fictitious catwhale and the extinct *Andrewsarchus* are represented in a reasonable way. The catwhale is close to the lion which is a good compromise for an animal with the conflicting features of a house cat except for the weight of a rhino. The *Andrewsarchus* is grouped with the Kodiak bear, which is reasonable, given the features by which it was described.

For comparison, the “single map” in figure 3 has been generated using a traditional unimodal Kohonen algorithm. This map has the same characteristics as the multimodal map. The detailed results are different but different experiments yield slightly different results and it is not possible to tell by the result if the map was generated as a multimodal map or as a single map.

4 Conclusion

The multimodal maps discussed in the paper have many potential benefits over their single map counterparts. One potential benefit of the multimodal map is that it allows a rich structure of parallel unimodal processing with many maps involved, followed by convergence into multimodal maps. More complex stimuli can therefore be processed without a growing map size. A second potential benefit to be explored is that this modular multimodal structure may be used to simulate biological neural systems.

Acknowledgment. This work has been supported by Monash University Small Grant, Faculty of Information Technology (Australia) and by the Department of Computer Science and Electrical Engineering, Luleå University of Technology (Sweden).

References

1. Kohonen, T.: *Self-Organising Maps*. 3rd edn. Springer-Verlag, Berlin (2001)
2. Gustafsson, L., Papliński, A.P.: Self-organization of an artificial neural network subjected to attention shift impairments and novelty avoidance: Implications for the development of autism. *J. Autism and Dev. Disorder* **34** (2004) 189–198

3. Papliński, A.P., Gustafsson, L.: An attempt in modelling early intervention in autism using neural networks. In: Proc. Int. Joint Conf. Neural Networks, Budapest, Hungary (2004) 101–108
4. Gustafsson, L., Papliński, A.P.: Neural network modelling of autism. In Casanova, M.F., ed.: Recent developments in autism research. Nova Science Publishers, Inc., Hauppauge, New York (2005) 100–134 in press.
5. Papliński, A.P., Gustafsson, L.: Detailed learning in narrow fields – towards a neural network model of autism. In Kaynak, O., Alpaydin, E., Xu, L., eds.: Lect. Notes in Comp. Sci. Volume 2714., Springer (2003) 830–838
6. Gustafsson, L., Papliński, A.P.: Preoccupation with a restricted pattern of interest in modelling autistic learning. In Pallade, V., Howlett, R.J., Jain, L., eds.: Lecture Notes in Artificial Intelligence. Volume 2774, Part II., Springer (2003) 1122–1129
7. Papliński, A.P., Gustafsson, L.: An attempt in modelling autism using self-organizing maps. In: Proc. 9th Intern. Conf. Neural Information Processing, Singapore (2002) 301–304
8. Alahakoon, D., Halgamuge, S.K., Srinivasan, B.: Dynamic self-organizing maps with controlled growth for knowledge discovery. *IEEE Trans. Neural Networks* **11** (2000) 601–614
9. Milano, M., Koumoutsakos, P., Schmidhuber, J.: Self-organizing nets for optimization. *IEEE Trans. Neural Networks* **15** (2004) 758–765
10. Xu, P., Chang, C.H., Papliński, A.: Self-organizing topological tree for on-line vector quantization and data clustering. *IEEE Tran. System, Man and Cybernetics, Part B: Cybernetics* **35** (2005) 515–526
11. Wallis, G., Rolls, E.: Invariant face and object recognition in the visual system. *Progress in Neurobiology* **51** (1997) 167–194
12. Rolls, E., Milward, T.: A model of invariant object recognition in the visual system: learning rules, activation function, lateral inhibition, and information-based performance measures. *Neural Computation* **51** (2000) 2547–2572
13. Mountcastle, V.B.: The columnar organization of the neocortex. *Brain* **120** (1997) 701–722
14. Felleman, D.J., van Essen, D.C.: Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex* **1** (1991) 1–47
15. Kandel, E.R., Schwartz, J.H., Jessel, T.M., eds.: *Principles of Neural Science*. 4th edn. McGraw Hill, New York (2000)
16. Rolls, E.T.: Multisensory neuronal convergence of taste, somatosensory, visual, and auditory inputs. In Calvert, G., Spencer, C., Stein, B.E., eds.: *The Handbook of multisensory processes*. MIT Press (2004) 311–331
17. Huttenlocher, P.R., Dabholkar, A.S.: Regional differences in synaptogenesis in human cerebral cortex. *J. Comparative Neurology* **387** (1997) 161–178
18. Rakic, P., Bourgeois, J.P., Goldman-Rakic, P.S.: Synaptic development of the cerebral cortex: implications for learning, memory, and mental illness. In van Pelt, J., Comer, M.A., Uylings, H.B.M., Lopes da Silva, F.H., eds.: *Prog. Brain Research*. Elsevier Sci. (1994) 227–243
19. Hopfield, J.: Neural networks and physical systems with emergent collective computational properties. *Proc. Nat. Academy of Sci. USA* **79** (1982) 2554–2588

Decision Fusion Based Unsupervised Texture Image Segmentation

Hua Zhong and Licheng Jiao

Institute of Intelligent Information Processing, Xidian University,
Xi'an 710071, China
hzhong@mail.xidian.edu.cn

Abstract. A decision fusion based method is proposed to improve unsupervised image segmentation. After the step of cluster label adjustment, each kind of texture is fixed with the same label. Then three simple fusion operators are applied according to the knowledge of multi-classifier fusion. Compared with feature fusion, decision fusion can combine the advantages of different features more intuitively and heuristically. Experimental results on textures and synthetic aperture radar (SAR) image demonstrate its superiority over feature fusion on removing the impact of noise feature and preserving the detail.

1 Introduction

Texture segmentation commonly consists of two steps: feature extraction and feature classification or clustering. However, it is known that no individual texture feature extraction method that is able to consistently and accurately segment textured images [1]. Therefore, considering the fusion of texture features is necessary.

Some papers fuse by simple concatenation of different feature sets [2][3][4][5], which often causes the problem of “curse of dimensionality”. A few papers focus on finding the complementary of texture features derived from different methods [1][6]. It is still a challenge to analyze theoretically the relationship between different feature sets whose complementary is hard to describe. Feature selection is another important issue [7], which improves fusion results in supervised learning, but received comparatively very little attention in unsupervised clustering. One important reason is that it is not at all clear how to assess the relevance of a subset of features without resorting to class labels.

In this paper, we aim at improving unsupervised image segmentation from another point of view, i.e. decision fusion. We first adjust the label of each cluster by a simple way so that the same texture will be assigned the same label in each run of clustering algorithm. Then, knowledge of multi-classifier fusion can be directly applied to fuse the cluster results.

This paper is organized as follows: section 2 briefly describes the methods of feature extraction and segmentation. Section 3 presents the algorithm of cluster label adjustment, and three simple fusion operators. Results and discussion are shown in section 4. Finally, the conclusions are drawn in section 5.

2 Feature Extraction and Segmentation

Only four GLCP statistics are used in this paper [8] (energy, entropy, homogeneity, dissimilarity). The parameters of GLCP feature are setup as follows: the quantized grey level $N_g = 16$, the displacement $d = 1$, and four orientation $\theta = 0^\circ, 45^\circ, 90^\circ, 135^\circ$. Thus each pixel is represented by a 16-D feature vector. The spatial window size W is chosen to be 5,9,17,33,65, separately. Each dimension of the entire feature set will be normalized individually to avoid certain features dominating the clustering results.

Standard Fuzzy c-mean algorithm (FCM) is used here. Compared with other supervised classification approaches, FCM needs no training of samples of known classes. However, different results may arrive in each run of FCM, and the label assigned to the same cluster is random. Due to this fact, application of many multi-classifier fusion schemes to clustering is restricted. In the next section, we will show how to overcome this limitation.

3 Fusion Method

3.1 Cluster Label Adjustment

Assume the number of clusters is c , pixels (i, j) belong to the same cluster are assigned a label $L_k(i, j)$, $k = 1, 2, \dots, c$, where $L_k(i, j)$ denotes the true segmentation. For each run of clustering algorithm, the pixel label may changed, denoted as $L'_k(i, j)$. What we will do is to find the optimal label $L_k^*(i, j)$ to replace $L'_k(i, j)$ so that most of the pixels in the same cluster will always get the same label, which is similar to a classifier. Under the restriction that pixels clustered into the same class will be assigned the same label, we use the following rule:

$$\{L_k^*\} = \min_{\{\hat{L}_k\}} \left\{ \sum_i \sum_j \text{sgn}(L_k(i, j) - \hat{L}_k(i, j)) \right\}. \quad (1)$$

where

$$\text{sgn}(L_k - L_k^*) = \begin{cases} 0, & \text{if } L_k = L_k^* \\ 1, & \text{if } L_k \neq L_k^* \end{cases}. \quad (2)$$

An example is shown in Fig. 1. We can see that the optimal label set is $\{L_1^*, L_2^*\} = \{L_2, L_1\}$, by which the difference between Fig. 1. (a) and Fig. 1. (b) is the smallest.

Since the true segmentation $L_k(i, j)$ is unknown, we can use the segmentation result from a comparatively more robust method instead of $L_k(i, j)$. Note that using any label set $\{L'_1, L'_2, \dots, L'_c\}$ has no impact for the clustering itself. However, improper label set will lead to serious error in decision fusion.

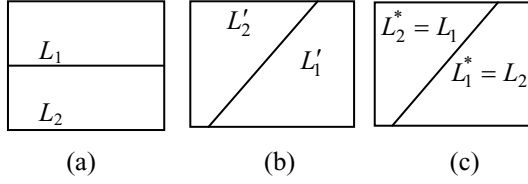


Fig. 1. Example of cluster label adjustment ($c = 2$). Label set $\{L'_1, L'_2\}$ is replaced with the optimal label set $\{L^*_1, L^*_2\}$. (a) True segmentation (b) Clustering result (c) Result after adjustment

3.2 Fusion Operators

With FCM, we can get the vector $\mathbf{U}(i, j) = [u_{l,k}(i, j), k = 1, 2, \dots, c]^T$ for each pixel, where $u_{l,k}(i, j)$ denotes the membership value of pixel (i, j) belonging to the cluster k under the l th feature set. Then the three operators can be described as follows.

1) Average rule:

$$L(i, j) = L_n, \text{ if } \frac{1}{L} \sum_{l=1}^L u_{l,n}(i, j) = \max_{k=1 \dots c} \left(\frac{1}{L} \sum_{l=1}^L u_{l,k}(i, j) \right). \quad (3)$$

2) Product rule:

$$L(i, j) = L_n, \text{ if } \prod_{l=1}^L u_{l,n}(i, j) = \max_{k=1 \dots c} \left(\prod_{l=1}^L u_{l,k}(i, j) \right). \quad (4)$$

3) Max-max rule:

$$L(i, j) = L_n, \text{ if } \prod_{l=1}^L u_{l,n}(i, j) = \max_{k=1 \dots c} \left(\prod_{l=1}^L u_{l,k}(i, j) \right). \quad (5)$$

4 Results and Discussion

It is known that window size W determines the ability to capture texture features. Small windows can lead to poor local estimates and large windows increase the risk of multiple textures appearing in the same window. Therefore, the segmentations with different window sizes are fused to get more precise results. For comparison, feature fusion by concatenation of different features is also carried out. Experiments are carried on 256×256 Brodatz mosaic images and synthetic aperture radar (SAR) images.

4.1 Mosaic of Textures

Table 1 gives the segmentation ratio of different fusion methods on a mosaic of textures (see Fig. 2. (a)) as the function of the window size W . The method “no fusion”

Table 1. Average segmentation ratio on the mosaic of textures with different value of the window size. The method “no fusion” means the segmentation ratio using the single feature set, and the other methods refer to the pairwise fusion using feature fusion and three decision operators, separately.

W	method	5	9	17	33	65
5	No fusion	57.0	—	—	—	—
	Feature fusion	—	67.6	71.3	71.2	70.1
	Average	—	68.9	72.3	72.3	66.4
	Product	—	68.1	71.7	71.1	66.4
	Max-max	—	70.0	73.2	73.0	66.3
9	No fusion	—	71.5	—	—	—
	Feature fusion	—	—	73.8	73.8	69.9
	Average	—	—	74.3	74.8	71.8
	Product	—	—	73.8	74.3	71.0
	Max-max	—	—	75.1	75.5	72.6
17	No fusion	—	—	72.3	—	—
	Feature fusion	—	—	—	73.0	68.7
	Average	—	—	—	73.2	71.5
	Product	—	—	—	72.6	69.6
	Max-max	—	—	—	73.8	73.2
33	No fusion	—	—	—	70.0	—
	Feature fusion	—	—	—	—	65.5
	Average	—	—	—	—	69.4
	Product	—	—	—	—	67.0
65	Max-max	—	—	—	—	71.0
	No fusion	—	—	—	—	61.2

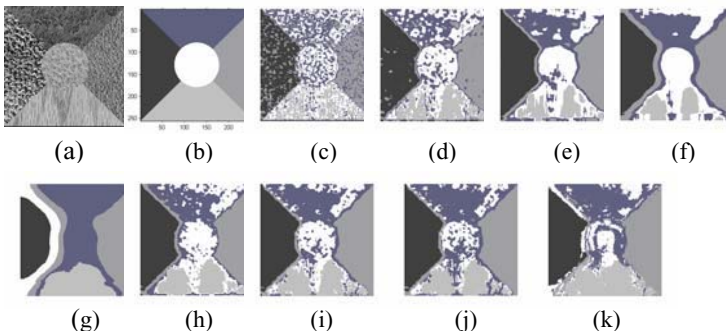


Fig. 2. (a) Original and (b) True segmentation. (c)-(g) displays the segmentations with different window size. (c) $W=5$ (d) $W=9$ (e) $W=17$ (f) $W=33$ (g) $W=65$. (h) Feature fusion with the combination $W=\{5,9,17,33,65\}$. (i)-(k) is the results of decision fusion with the same combination. (i) Average rule (j) Product rule (k) Max-max rule.

means the segmentation ratio of the single feature set, and the others refer to the pairwise fusion using feature fusion and three decision fusion operators, separately. The average of 25 FCM runs is reported for each test. The true segmentation, shown in Fig. 2. (b), is used for cluster label adjustment in order to get a correct conclusion.

In the case of no fusion, we can see that it is not satisfied for a window size W of a small value 5 or a large value 65. The segmentation maps of one run are shown in Fig. 2, which further illustrates the reasons: Fig. 2. (c) looks much noisy with $W=5$, while Fig. 2. (g) is too smooth around the edges with $W=65$. Then we make an analysis on the fusion results. Four points can be noticed: 1) Improved results are obtained for most cases; 2) Fusion the results at two ends, i.e. with the pair $W=\{5,65\}$, get slightest improvement; 3) For two neighboring values of W , for example, $\{5,9\}$, $\{17,33\}$, little improvement is made; 4) Results of decision fusion methods are always better than that of feature fusion, and Max-max rule performs the best.

Some positive fusion results using other combinations of the window size values are also obtained. One example is shown in Fig. 2. (h)-(k) using the combination of $W = \{5,9,17,33,65\}$. Results of decision fusion using the same combination is shown in, respectively. We can see that decision fusion results in Fig. 2. (i)-(k) perform better than feature fusion on removing the noise caused by small window sizes.

4.2 SAR Image

A SAR image is shown in Fig. 3 (a). Since no true segmentation for a natural image, we can verify the effectiveness of cluster label adjustment. The SAR image used here can be visually divided into two clusters: mountains and plain. But we set the cluster number of $c = 3$ in order to well describe the edge area between mountains and plain. The segmentation with $W=33$ is used to adjust cluster labels because of its robustness and compromise in window size. Fig. 3 (b)-(f) present the segmentations with different window size from 5 to 65. Similar conclusion can be drawn about the window size. The cluster colors in each segmentation are the same, which demonstrates the cluster label adjustment is effective.

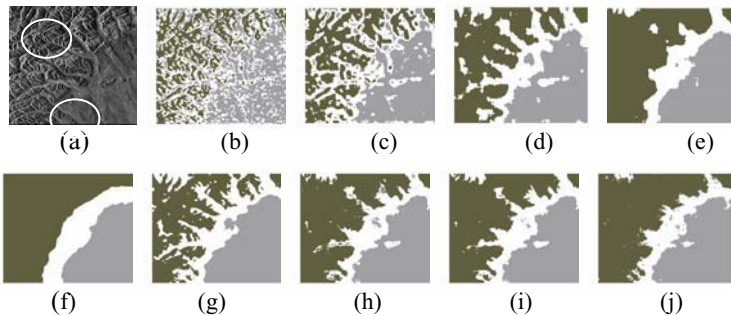


Fig. 3. (a) Original (b)-(f) displays the segmentations with different window size. (b) $W=5$ (c) $W=9$ (d) $W=17$ (e) $W=33$ (f) $W=65$. (g) Feature fusion with the combination $\{5,9,17,33,65\}$. (h)-(j) are the results of decision fusion with the same combination. (h) Average rule (i) Product rule (j) Max-max rule.

Restricted by the paper length, we only give the fusion results with the combination of $W = \{5,9,17,33,65\}$. Notice that feature fusion, shown in Fig. 3 (g), removes the area in the circle, while decision fusion preserves this detail well, as shown in

Fig. 3 (h)-(j). Also, some small details on the left area of the image appear noisy, and are smoothed by decision fusion, which is consistent with the results in section 4.1.

5 Conclusion

This paper attempts to use decision fusion instead of commonly used methods of feature fusion for unsupervised image segmentation. Before the fusion of each segmentation, a step of cluster label adjustment is performed to fix one label to the same kind of texture in each run. In each test on either textures or SAR image, preliminary results demonstrate that simple fusion operators always outperform feature fusion. Decision fusion can remove the impact of noise feature and preserve the detail more effectively.

References

1. Claudi, D.A., Deng, H.: Design-based Texture Feature Fusion Using Gabor Filters and Co-Occurrence Probabilities. *IEEE Trans. Image Processing.* 7(2005) 925–936
2. Solberg, A.H.S., Jain, A.K.: Texture Fusion and Feature Selection Applied to SAR Imagery. *IEEE Trans. Geosci. Remote Sens.* 2(1997) 475–479
3. Bashar, K.M., Ohnishi, N.: Fusing Cortex Transform and Intensity Based Features for Image Texture Classification. *Proc. 5th Int. Conf. Information Fusion.* 1(2002) 1463–1469
4. Chindaro, S., Sirlantzis, K., Deravi, F.: Color space fusion for Texture Recognition. *Proc. 4th EURASIP Conf. Video/Image Processing and Multimedia Communications.* 1(2003) 181–186
5. Trianni, G., Tosi, M., Dell’Acqua, F., Gamba, P., Lisini, G.: Fusion of Texture Measures for Urban Area Characterization. *Proc. 7th Int. Conf. Information Fusion.* 1(2004) 991–998
6. Claudi, D.A.: Comparison and Fusion of Co-occurrence, Gabor and MRF Texture Features for Classification of SAR Sea-ice Imagery. *Atmos.-Ocean.* 3(2000) 183–194
7. Law, M.H.C., Figueiredo, M.A.T., Jain, A.K.: Simultaneous Feature Selection and Clustering Using Mixture Models. *IEEE Trans. on PAMI.* 9(2004) 1154–1166
8. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural Features for Image Classification. *IEEE Trans. Syst. Man. Cybern.* 6(1973) 610–621

Speaker Adaptation Techniques for Speech Recognition with a Speaker-Independent Phonetic Recognizer

Weon-Goo Kim¹ and MinSeok Jang²

¹ School of Electronic and Information Eng., Kunsan National Univ.,
Kunsan, Chonbuk 573-701, Korea
Biometrics Engineering Research Center (BERC)
wgkim@kunsan.ac.kr

² Dept. of Computer Information Science, Kunsan National Univ.,
Kunsan, Chonbuk 573-701, Korea
msjang@kunsan.ac.kr

Abstract. We present a new method that improves the performance of the speech recognition system with the speaker-independent (SI) phonetic recognizer. The performance of the speech recognition system with the SI phonetic recognizer is worse than that of the speaker dependent system due to the mismatch between the training utterances and a set of SI models. A new training method that iteratively estimates the phonetic templates and transformation vectors is presented to reduce the mismatch using speaker adaptation techniques. The stochastic matching methods are used to estimate the transformation vectors for speaker adaptation. The experiment performed over actual telephone line shows that a reduction of about 40% in the error rates could be achieved as compared to the conventional method.

1 Introduction

As the popularity of speaker-dependent (SD) system increases, the requirements of data storage and access for performing speech recognition become crucial. To overcome this problem, methods that use the speaker-specific phonetic templates with a set of speaker-independent (SI) models were presented [1–4]. The advantage of these approaches are that the only information that need to be stored for each speaker is the phonetic string associated with each speaker's words or phrase from speech in terms of SI sub-word acoustic units (phones), resulting in substantial data reduction. However, this method has two drawbacks. The first one is that a large amount of phoneme recognition errors are generated by using SI phoneme HMMs. The other is that the performance of the system is worse than that of the SD system due to the mismatch between the training utterances and a set of SI models.

In this paper, a new training method that jointly estimates the phonetic templates and the transformation functions for the speaker adaptation from training utterances is presented to improve the performance of the personal

voice dialing system using SI phoneme HMMs. In training process, the phonetic templates and the transformation functions are estimated iteratively from the training utterances of each user with maximum likelihood stochastic matching methods based on the phonetic class or codebook using SI phoneme models. In recognition process, after SI phoneme models are transformed using the speaker specific transformation functions, input sentence is recognized.

2 Speaker Adaptation for Speech Recognition with SI Phoneme HMMs

A method that jointly estimates the phonetic template (transcription) and the transformation function (bias) for the speaker adaptation from training utterances is presented to improve the performance of the personal voice dialing system using SI phoneme HMMs. There are two kinds of sessions, enrollment and test. In an enrollment session, a user need to repeat a name for two or three times, then input a telephone number associated with the name. The joint estimation stage consists of two pass, speech recognition and adaptation. The first pass is standard decoding which produce the phoneme transcription as well as the state segmentation necessary for the estimation of SI phoneme model biases. During the second pass, model biases are estimated using stochastic matching method to generate the new set of HMMs. This process is iterated several times until achieving convergence and then final transcriptions and biases of each user are saved into the database. In a test session, the user just needs to utter the name. For telephone, the identity can be obtained from caller ID or from user's input. After SI phoneme HMMs are adapted using bias vectors, decoder recognize the input sentence with the transcriptions and adapted SI phoneme HMMs.

The maximum likelihood stochastic matching method to decrease the mismatch between training utterance and SI phoneme HMMs can be applied as follows [5]. Consider a sequence of feature vector $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_T\}$ and a set of SI phoneme HMMs Λ_X . The transformation G_η with parameters η maps Λ_X into the transformed model Λ_Y so that $\Lambda_Y = G_\eta(\Lambda_X)$.

One Approach to decreasing the mismatch between \mathbf{Y} and Λ_X is to find parameters η and the phoneme sequence \mathbf{W} that maximize the joint likelihood of \mathbf{Y} and \mathbf{W} . Thus, we need to find η' such that

$$(\eta', \mathbf{W}') = \arg \max_{(\eta, \mathbf{W})} p(\mathbf{Y}, \mathbf{W} | \eta, \Lambda_X) = \arg \max_{(\eta, \mathbf{W})} p(\mathbf{Y} | \mathbf{W}, \eta, \Lambda_X) P(\mathbf{W}) . \quad (1)$$

This joint maximization over the variables η and \mathbf{W} in (2) can be done iteratively by keeping η fixed and maximizing over \mathbf{W} , and then keeping \mathbf{W} fixed and maximizing over η . Under the assumptions, the structure of Λ_Y remains the same as that of Λ_X . In this study, two kinds of stochastic matching methods, phonetic class based and codebook-based, are used to estimate the biases. In the phonetic class based stochastic matching, multiple biases are used according to the phonetic classes. The number of bias is selected according to the acoustic phonetic class. In the codebook-based stochastic matching, the concept

of "tying" among model parameters is used in determining the number of biases. Different degrees of tying control the size of the codebook and consequently the number of biases used. The codebook could range in size from one entry, which applies a global bias to entire SI HMMs, to as many entries as the number of mixture components available. The codebook is constructed by clustering the set of mean vectors of mixture components for SI HMMs using the Lloyd algorithm with a Euclidean distance.

Let $\Omega_1, \dots, \Omega_K$ be K classes. In phonetic class based or codebook-based stochastic matching, K is the number of phonetic class or codeword each. The bias $\boldsymbol{\mu}_{b_k} = \{\mu_{k,1}, \mu_{k,2}, \dots, \mu_{k,D}\}$ can be obtained by maximum likelihood estimation, such that

$$\mu_{b_{k,i}} = \frac{\sum_{t=1}^T \sum_{(n,m) \in \Omega_k} \gamma_t(n,m) \frac{y_{t,i} - \mu_{n,m,i}}{\sigma_{n,m,i}^2}}{\sum_{t=1}^T \sum_{(n,m) \in \Omega_k} \frac{\gamma_t(n,m)}{\sigma_{n,m,i}^2}}, \quad i = 1, \dots, D, \quad k = 1, \dots, K \quad (2)$$

and

$$\gamma_t(n,m) = \begin{cases} \frac{w_{n,m} N[y_t; \mu_{n,m}, C_{n,m}]}{\sum_{j=1}^M w_{n,m} N[y_t; \mu_{n,m}, C_{n,m}]} , & \text{if } s = n \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where D is the dimension of the feature vector, $\mu_{n,m}$, $C_{n,m}$ are the mean and variance vector corresponding to mixture m in state n , $w_{n,m}$ is the probability of mixture m in state n , N is the normal distribution and s is the state sequence corresponding to the input utterance. If $\mu_{n,m}$ is clustered to the k -th class or codeword, then $(n,m) \in \Omega_k$.

The mean $\hat{\mu}_{n,m}$ of each mixture component in adapted SI HMMs are derived by adding the bias to the mean $\mu_{n,m}$ of the corresponding mixture components in SI HMMs, such that

$$\hat{\mu}_{n,m} = \mu_{n,m} + \sum_{k=1}^K \mu_{b_k} \mathbf{I}_{\Omega_k}(n,m), \quad \forall (n,m). \quad (4)$$

where $\mathbf{I}_{\Omega_k}(\cdot)$ is the indicator function for the set Ω_k .

The proposed method in enrollment session consists of three steps:

1. Multiple transcription hypothesis of the input utterance and state segmentations are obtained using SI phoneme HMMs.
2. The transformation vectors are estimated using stochastic matching method.
3. SI phoneme models are adapted with the transformation vectors.
4. Step 2 and 3 are iterated until adaptation becomes sufficiently precise and then multiple transcriptions and bias vector are saved into database.

3 Experimental Results

3.1 Database and System Setup

The experimental database consists of 10 speakers, 5 males and 5 females speaking 15 name entries repeated 13 times by each speaker over a period of several

weeks [6]. The database evaluation is on a worst-case situation where all the names are "Call ", e.g. "Call office", "Call home", etc. This database was collected over the telephone network using digital telephone interface. The input speech was sampled at 6.67kHz and saved as 8bit μ -law PCM format. In training, three utterances of each name recorded in one session were used to train a name model. In testing, ten utterances from each speaker collected from 5 different sessions were used to evaluate the recognition performance.

The sampled input speech was pre-emphasized using a first-order filter with a coefficient of 0.97, and the analysis frames were 30-ms width with 20-ms overlap. A vector of 39 features was extracted based on tenth order LPC analysis. The feature corresponds to a 12 cepstrum, 12 delta cepstrum, 12 delta-delta cepstrum, a normalized log energy, a delta log energy and a delta-delta log energy.

The SI phoneme models consisting of 41 phoneme HMMs and a silence HMM were trained using the database collected over the telephone network. The models are left-to-right HMMs and each HMMs have 3 or 5 states consisting of 10 continuous Gaussian mixture components and a silence model has one state consisting of 256 continuous Gaussian mixture components.

3.2 Performance of Baseline System

The base line system is SD voice dialing system using speaker-specific phonetic templates and SI phoneme HMMs. Although this kind of system can in substantially reduce storage space, a large amount of phoneme recognition errors are generated, especially at the front and end of the input utterance. One way to reduce the phoneme recognition errors is to use the end point detection method. Voice dialing experiment was conducted with and without the end point detection method. As expected, using end point detection method reduces percentage word error rates from 4.2% to 3.8%.

3.3 Performance of the Speaker Adaptation Techniques

Two kinds of stochastic matching methods are used to estimate the biases. In the phonetic class based stochastic matching, the number of bias is selected according

Table 1. The number of bias according to the phonetic class for phonetic class based stochastic matching

number of bias	phonetic class
1	no phonetic class
2	silence and speech
3	silence, vowel and consonants
9	silence, vowel, diphthongs, semivowels, stops, ...
14	silence, {front, mid, back} vowel, diphthongs, liquids, glides, {voiced, unvoiced} stops, ...
42	silence and all phones
180	all of SI HMM states

to the acoustic phonetic class. Table 1 shows the number of bias according to the phonetic class. In the codebook-based stochastic matching, the number of bias is the same as the size of the codebook.

Fig. 1 shows the percentage word error rates according to the number of transformation vector after several iteration. In Figure 1(a), the performance of proposed system did not enhanced when the number of transformation vector is fewer than 9. However, error reduction could be achieved when the number of transformation vector is more than 14. When 42 transformation vectors are used, the best performance (2.3%) could be achieved. It corresponds to about a 40% reduction in word error rate as compared to the performance of the baseline system. Figure 2(b) illustrates the effect of varying the codebook size from one to 512. This experiment seems to indicate that a codebook size of 32 and beyond produce error reduction. Using a codebook size of 128, 2.6% word error rate could be achieved.

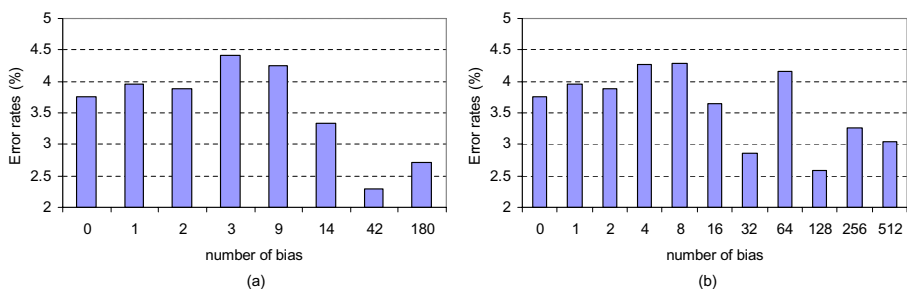


Fig. 1. Performance of the voice dialing system using speaker adaptation algorithm according to the phonetic class(a) and codebook(b)

Table 2. Performance comparison of the proposed speaker adaptation method with conventional ones

Type of system	Baseline	Baseline with bias	Proposed system	Ideal case	SD system
Error rate(%)	3.8	3.3	2.3	2.3	1.8

Table 2 shows the performance comparison of the proposed method with the conventional methods. The error rate of baseline system is 3.8%. The error rate of system adapted with the transformation vector only(i.e., transcriptions are not updated) is 3.3%. This shows that adaptation with the transformation vector only could reduce the error rate. The error rate of proposed system in which the transformation vectors and the transcriptions are estimated iteratively is 2.3%. This is the same error rate of ideal case in which correct transcriptions are assumed to be known. Finally, the error rate of SD system is 1.8%. Although this is the lowest, large amount of storage space are required.

4 Conclusion

A new method that jointly estimates the transcriptions and the transformation vectors for the speaker adaptation is presented in order to improve the performance of the personal voice dialing system in which SI phoneme HMMs are used. The transformation vectors and transcriptions are estimated iteratively from the training data of each user with maximum likelihood stochastic matching using SI phoneme models. Two kinds of stochastic matching methods, phonetic class and codebook based, are used to estimate the biases. Experimental result shows that the performance of proposed system corresponds to about a 40% reduction in word error rate, when compared to the performance of the baseline system.

Acknowledgement

This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Biometrics Engineering Research Center (BERC) at Yonsei university.

References

1. Jain, N., Cole, R. Barnard, E.: Creating Speaker-Specific Phonetic Templates with a Speaker-Independent Phonetic Recognizer: Implications for Voice Dialing. Proc. of ICASSP (1996) 881–884
2. Fontaine, V., Boulard, H.: Speaker-Dependent Speech Recognition Based on Phone-Like Units Models-Application to Voice Dialing. Proc. of ICASSP (1997) 1527–1530
3. Ramabhadran, B., Bahl, L.R., deSouza, P.V., Padmanabhan, M.: Acoustic-Only Based Automatic Phonetic Baseform Generation. Proc. of ICASSP (1998) 2275–2278
4. Deligne, S., Mangu, L.: On the use of Lattices for Automatic Generation of Pronunciations. Proc. of ICASSP (2003) 204–207
5. Sankar, A., Lee, C.H.: A Maximum-Likelihood Approach to Stochastic Matching for Robust Speech Recognition. IEEE Trans. on Speech and Audio Processing, Vol. 4. (1996) 190–202
6. Sukkar, R.A., Lee, C.H.: Vocabulary Independent Discriminative Utterance Verification for Non-keyword Rejection in Subword based Speech Recognition. IEEE Trans. on Speech and Audio Processing, Vol. 4. (1996) 420–429

Fuzzy QoS Controllers in Diff-Serv Scheduler Using Genetic Algorithms

Baolin Sun¹, Qiu Yang², Jun Ma¹, and Hua Chen¹

¹ Department of Mathematics and Physics,
Wuhan University of Science and Engineering, Wuhan 430073, China
b1sun@163.com

² School of Mathematics and Physics,
China University of Geosciences, Wuhan 430074, China

Abstract. Quality of Service (QoS) requirements in networks with uncertain parameters has become a very important research issue in the areas of Internet, mobile networks and distributed systems. This is also a challenging and hard problem for the next generation Internet and mobile networks. It attracts the interests of many people. In this paper we propose a methodology to choose optimized fuzzy controller parameters using the genetic algorithms. Specifically, differentiated service scheme with feedback preference information (FPI) is studied in more detail to illustrate the implement of the new approach. Simulation shows that the approach is efficient, promising and applicable in ad hoc networks. The performance of this scheduler is studied using NS2 and evaluated in terms of quantitative metrics such as packet delivery ratio, average end-to-end delay. Simulation shows that the approach is efficient, promising and applicable in Diff-Serv networks.

1 Introduction

The provision of Quality of Service (QoS) requirements is of utmost importance for the development of future networks. Recent advances in switching and transmission technologies allow the implementation of very high speed networks that carry vast amounts of traffic which is generated by applications that are more sensitive to data quality (such as video or audio), and at the same time less predictable than current fixed rate sources [1-6]. In the next telecommunication age it will be possible to support new multimedia applications in a global environment and design new services on flexible platforms without upgrading the physical infrastructure. This requires new network architectures capable of offering transport and computation services to communication applications with stringent QoS requirements. A key issue is the provision of network resources so as to meet these requirements.

In [5-7], they proposed the use of fuzzy logic controllers for the dynamic reconfiguration of edge and core routers. Lorenz and Orda demonstrate in [8] that this uncertainty places additional constraints on QoS provisioning. A fuzzy controller is specified by fuzzy sets definition (membership function) and a set of rules (rule base). Ghosh et al [9] present a survey on the use of fuzzy logic

in telecommunication networks. Several works make use of controllers based on fuzzy logic.

In this paper, we propose a fuzzy based Scheduler for scheduling the packets based on its priority index. The priority index for each packet is determined based on number of hops the packet has suffered and the buffer size. It is found that the proposed fuzzy scheduler provides improved packet delivery ratio, reduced average end-to-end delay and increased throughput, when tested under various mobility conditions.

The rest of the paper is organized as follows. Section 2 previous a brief description of related works. Section 3 presents the fuzzy QoS controller. Section 4 shows the genetic algorithms optimization. Some simulation results are provided in Section 5. Finally, Section 6 presents the conclusions.

2 Related Works

2.1 FPI Scheme

The different priority bytes in the FPI theme are shown in Table 1. According to this table, one can understand that bit 0 at a different priority byte indicates if the packet is real-time one, and bit 1 indicates if it is a preference service. Bits 0 and 1 are referred to as the TOS bits in the FPI concept.

Table 1. The DPCP allocation in the FPI scheme

Priority Class	Code point space	Type of service
4	11XXXX	Real-time flow with preference service
3	10XXXX	Real-time flow with normal service
2	01XXXX	Non-real-time flow with preference service
1	00XXXX	Non-real-time flow with normal service

2.2 Different Priority Controller Architecture

The controllable elements in the different priority architecture are shown in Fig. 1. In this architecture, all nodes have a separate queue for each service class; a classifier places the packets into the respective queue and the scheduler selects packets from these queues for transmission in the output links. The proposed architecture implements two controllers: one that controls the queues and scheduler, which is used in the core and edge nodes, and the other that controls the policer, which is only used on the edge nodes.

3 Fuzzy QoS Controller

3.1 Different Priority Controller Architecture

Fuzzy control system is rule-based system in which a set of so-called fuzzy rules represents a control decision mechanism to adjust the effects of certain causes

coming from the system. The aim of fuzzy control system is normally to substitute for or replace a skilled human operator with a fuzzy rule-based system.

There are generally two kinds of fuzzy logic controllers. One is feedback controller, which is not suitable for the high performance communication networks. Another one, which is used in this paper, is shown in Fig. 2. The output of the fuzzy logic controller in Fig. 2 is used to tune the controlled system’s parameters based on the state of the system. This control mechanism is different from the conventional feedback control and considered as an adaptive control.

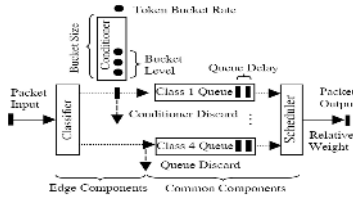


Fig. 1. Different priority controllable architecture

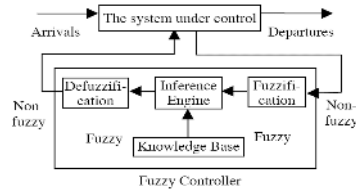


Fig. 2. The fuzzy controller

3.2 Scheduler Controller

The packet scheduler used in our architecture is WRR (Weighted Round Robin). In this scheduler, queues are served according to a configurable weight that can be changed during network operation. This allows having control of the bandwidth assigned to each service class. An example of membership function of schedule controller is showed in Fig. 3. Other membership functions are: packet delay in the EF queue and discard rate due to queue overflow in the BE class.

3.3 Rule Base and Inference

Rule base is an IF-THEN rule group with fuzzy sets that represents the desired behavior of a fuzzy system. It can be defined in agreement with the administrative policy.

1) Scheduler Controller. A synthesis of the scheduler controller rule base is presented below: (a) If the delay in EF queue is medium, then the queue weight is increased by one level; e.g., if the weight was low it goes to medium. And if EF delay is high the queue weight is increased by two levels. (b) If the delay in EF queue is low and the packet discard rate in BE queue is medium, then queue weight is reduced by one level. And if BE packet discard rate is high the queue weight is reduced by two levels.

2) Policer Controller. A synthesis of the policer controller rule base is presented below: (a) If the EF queue delay in core node is medium, then the policer rate is reduced by one level. If EF queue delay in core nodes is high the policer rate is reduced by two levels. (b) If the packet discard rate of BE class in edge node is high and EF queue delay in core nodes is low, then the policer rate is increased by one level. If packet discard rate of BE class in edge node is high and EF queue delay is low, the policer rate is increased by two levels.

3.4 Model Description

The controlled model is consists of two buffers, shown in Fig. 4. Buffer 10 contains the real-time packets with normal service selection and 01 contains the non-real-time packets with preference service selection. Both buffers have limited lengths, $B_{10} \leq B_{01}$ [6], and all the packets at each buffer implement the FIFO discipline. The length for the real-time buffer should has a maximal limit, which can guarantee that the maximal sojourn time for a FIFO packet will not exceed the maximal tolerance time for the real-time requirement.

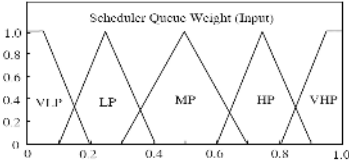


Fig. 3. Scheduler membership functions

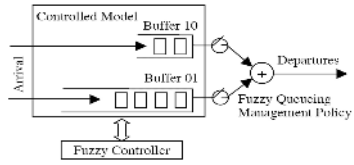


Fig. 4. Fuzzy controlled model

4 Genetic Algorithm Optimization

Genetic algorithms are based on the mechanics of natural evolution. Throughout their artificial evolution, successive generations each consisting of a population of possible solutions, called individuals (or chromosomes, or vectors of genes), search for beneficial adaptations to solve the given problem. This search is carried out by applying the Darwinian principles of "reproduction and survival of the fittest" and the genetic operators of crossover and mutation which derive the new offspring population from the current population. Reproduction involves selecting, in proportion to its fitness level, an individual from the current population and allowing it to survive by copying it to the new population of individuals. The individual's fitness level is usually based on the cost function given by the problem under consideration. Then, crossover and mutation are carried on two randomly chosen individuals of the current population creating two new offspring individuals. Crossover involves swapping two randomly located sub-chromosomes of the two mating chromosomes. Mutation is applied to randomly selected genes, where the values associated with such a gene is randomly changed to another value within an allowed range. The offspring population replaces the parent population, and the process is repeated for many generations. Typically, the best individual that appeared in any generation of the run is designated as the result produced by the genetic algorithm.

5 Simulation

The platform used was the Network Simulator (NS), version 2.26 [11]. It is a DS domain composed of 50 nodes, where 30 core nodes and 20 edge nodes. There are

5 ingress edge nodes and 5 egress edge nodes. The delay of each link of 2 Mbps is 10 ms, all queues have a maximum size of 50 packets, which gives a maximum delay of 100 ms in each node. The simulation model uses a WRR scheduler, Drop Tail queues in both classes and Token Bucket policer for EF class.

In this performance evaluation the following performance metrics were evaluated: percentile of edge-to-edge delay and jitter in EF class and discard rates in the EF and BE classes. For each evaluation, we used CBR and exponential On/Off traffic. We show the tables of the Diff-Serv domain without controller, with a conventional controller and with the proposed fuzzy controller. All simulations start with initial scheduler configuration with 60% of the bandwidth for each class. To eliminate simulation results with an empty network, we start collecting results 30 seconds after the beginning of the simulation.

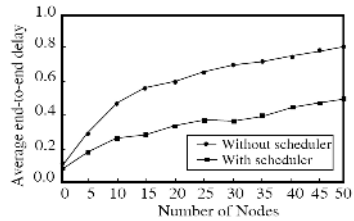
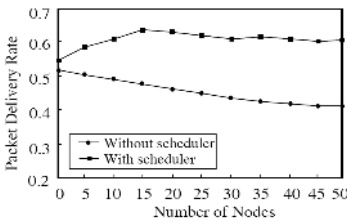


Fig. 5. Packet delivery ratio vs. networks size **Fig. 6.** Average end-to-end vs. networks size

Fig. 5 shows the graph of average edge-to-edge delay of an EF flow using CBR traffic. It compares the original fuzzy controller (without optimization) and the optimized fuzzy controller (after optimization with genetic algorithm). The average and to end delay performance as shown in the Fig. 6, proves that the end-to-end delay improves by 0.3 sec when scheduler is included. This is due to increase in number of hops the packets have to take for reaching the destination. Both graphs show an improvement in QoS, as we expected.

6 Conclusion

This paper addresses a fuzzy controllers based priority scheduling scheme, which improves the Quality of service parameters in Diff-Serv networks. The fuzzy controllers scheduler algorithm attaches a priority index to each packet in the queue of the node. It combines the input parameters such as queue length, data rate and expiry time to find the priority index. The membership functions and rule bases of the fuzzy scheduler are carefully designed. The use of fuzzy logic improves the handling of inaccuracy and uncertainties of the ingress traffic into the domain. Simulation shows that the approach is efficient and promising.

References

1. Sun, B.L., Li, L.Y.: A QoS Based Multicast Routing Protocol in Ad Hoc Networks. *Chinese Journal of Computers*, Vol. 27, No.10, (2004) 1402-1407 (in Chinese)
2. Sun B. L., Li L. Y.: An Optimizing on Multiple Constrained QoS Multicast Routing Algorithm Based on GA. *Journal of Systems Engineering and Electronics*, Vol. 15, No. 4, (2004) 677-683
3. Sun, B.L., Li, L.Y., Chen, H.: An Intrusion Detection in Mobile Ad Hoc Networks. The 4th International Conference on Cryptology and Network Security (CANS2005), LNCS, Springer Verlag, Dec. (2005)
4. Sun, B. L., Yin, X. H., and Li, L. Y.: Optimizing Fuzzy Controllers for QoS Improvement in DiffServ Networks. *Proceedings of the 7th Joint Conference on Information Sciences (JCIS 2003)*, Cary, North Carolina, USA, September (2003) 521-525
5. Zhang, R., Ma, J.: On the Enhancement of a Differentiated Services Scheme. *Proc. NOMS'2000*, Honolulu, USA, 2000.
6. Zhang, R., Phillis, Y.: Fuzzy Control of Queueing System with Heterogeneous Servers. *IEEE Trans. Fuzzy Systems*, Vol. 7, No. 1, (1999) 17-26
7. Fernandez, M. P., de Castro, A., Pedroza, P., and de Rezende, J. F.: QoS provisioning across a diffserv domain using policy-based management. in *Globecom 2001*, San Antonio, USA, Nov. (2001)
8. Lorenz, D., Orda, G.: QoS routing in networks with uncertain parameters. in *IEEE Infocom 98*, (1998)
9. Ghosh, S., Razouqi, Q., Schmacher, H. J., and Celmins, A.: A survey of recent advances in fuzzy logic in telecommunications networks and new challenges. *IEEE Transactions on Fuzzy Systems*, vol. 6, Aug. (1998) 443-447
10. Velasco, J., Magdalena, L.: Genetic algorithms in fuzzy control systems. in *Genetic Algorithms in Engineering and Computer Science*, John Wiley & Sons, (1995) 141-165
11. The Network Simulator - ns-2,: <http://www.isi.edu/nsnam/ns/>. (2004)

Neural Network Based Algorithms for Diagnosis and Classification of Breast Cancer Tumor

In-Sung Jung, Devinder Thapa, and Gi-Nam Wang

Department of Industrial and Information Engineering, Ajou University, South Korea
{gabriel17, debu, gnwang}@ajou.ac.kr

Abstract. This paper outlines an approach for applying acquiring numerical breast cancer image data and diagnosis using neural network algorithm in a way that is easy to classify between benign and malignance. This paper is an extended work related to our previous work [1]. In our previous work we used k-means algorithm to detect and diagnosis breast cancer tumor's region. However, to find the better results from the algorithm we need to add more numerical parameters of the breast cancer image and this algorithm has limited usage when applied with more number of parameters. Even if the cancer tumor is abnormal it was quite difficult to distinguish among those tumors. This paper summarizes the different comparative study of neural network algorithms to get the best classification of the breast cancer and explains how to acquire more numerical parameters from the breast cancer image data, so that it can help doctors to diagnosis efficiently between benign and malignance tumors.

1 Introduction

Diagnoses with X-ray have many difficulties like radiation risks, cancer risks from breast compression etc. Alternatively, detection and diagnosis using ultrasound is less risk prone than mammography. However, it was quite difficult to get the understandable image by using ultrasound due to sensitivity to noise while transferring to computers. In this paper, we illustrate the use of ultrasound image with computer aided diagnosis. This research papers is an extended work to reduce the variability of radiologist diagnosis by interpretation of the ultrasound images. Previous work has used the k-means algorithm to detect and diagnosis breast cancer tumor while this algorithm has some flaws when the tumor is abnormal. In this paper, attempts are given to make comparative study of different neural network algorithms to detect a slight change of abnormal symptoms by tracing and comparing different test results systematically. Detection of any abnormal condition is performed, and classification of correct status is also performed. Various detection and classification technologies are utilized for constructing CAD system. A diagnosis expert could add diagnosis class according to a specific symptom and modify classification criteria. Synthetically generated and real data (i.e., ultrasound images) are utilized to verify the proposed approach. Experimental results are presented to show the possibility of real time implementation.

2 Related Works

Not much work has been done to find out the comparative results of neural network techniques for breast cancer diagnosis. Reference [6] has used the SOM for the clustering of breast cancer images for the computer aided diagnosis database based on mammographic findings and patient age. But our approach is to use mammography as well as ultrasound images, ultrasound images are more noise prone and difficult to make clustering. A comparison between back-propagation and GA has been made by reference [5]. They found that the result of GA is slightly better than the BP, although not significantly, than BP for the training of the CAD schemes. SOM and MLP trained with the back-propagation algorithm has been used to extract features and suggest the final diagnosis. All the previous works are basically focused on mammography images and they have not made any comparative analysis of the different algorithms [4]. The objective of our work is to comparative different neural network techniques for the better detection and diagnosis of the breast cancer. Novel approach of this paper is to use one more parameter called ratio analysis to classify the cancer tumor more precisely.

3 Proposed Method

In our previous work we used on-line k-means clustering algorithms to detect and diagnosis the abnormalities [1] but as the figure 1 shows that K-means cannot detect some images which are at the boundary line of benign and malignance. It may create another class of clustering which is not required for the diagnosis. The main objective of this paper is to make a comparative study of the neural network algorithms such as ART , modified K-Means Algorithm, SOM and LVQ, which can be better utilized to overcome the previous drawbacks.











	Name	Type	Image	C	Standard Deviation
Benign	Smooth			10.90	1.52
	Lobulate			14.32	1.83
	Micro lobulate			14.76	2.22
	Irrgulate			21.06	23.2
Cancer	Spiculate			77.21	24.9

Fig. 1. Clustering created by using k-means algorithm

4 Our Approach

In our previous work we have used the two parameters to detect the cancer tumor but this paper will use one more attributes ratio analysis [1]. These 3 dimensional attributes can give clear images of the benign and malignance tumor types. The first algorithm used for this purpose was Adaptive K-means algorithm. The following are the steps used to create better clustering applying this technique.

4.1 Methodology

This paper outlines an approach for acquiring numerical data of the breast cancer image and diagnosis using neural network algorithm for easy classification between benign and malignance.

4.1.1 Method for Acquiring Numerical Data

Our previous work [1] has already presented breast lesion detection and segmentation using morphology, so we use the edge of tumor image for applying to acquire numerical data of tumor.

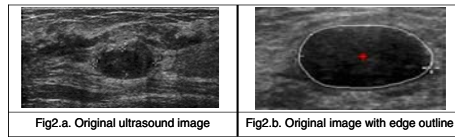


Fig. 2. Original ultrasound image with edge outline

Breast cancer diagnosis requires hospital domain knowledge to obtain numerical data of the shape because a shape is one of the important factors based on domain knowledge. Therefore we computed the tumor’s distance from center to edge with Euclid method as follows:

$$distance = \sqrt{((y - y_c)^2 + (x - x_c)^2)} \tag{1}$$

4.1.2 Plotting Data on Graph

This calculated value can be plotted to the graph (Fig 3). If the curve looks like figure (b), then it will be a benign, or if the plotted line seems to be like in figure(c), it will be a cancer tumor. X-axis is angle and Y-axis is deviation.

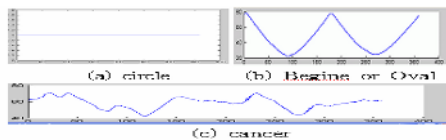


Fig. 3. Type of distance plotting

4.1.3 Analysis of the Tumor Circularity

Circularity is one of the important factors for the diagnosis. If shape of the tumor seems like a circle, it can be a benign but if the shape is irregular, it can be a malignance. Hence, we computed the tumor circularity for analysis. In our paper we have tried to analyze the tumor circularity as well as compute standard deviation for more detail study of the image. As per computation, If C is closed to zero that means there is the chances or possibility that the tumor is benign.

C=Circularity ;P=Perimeter ; A = Area

$$C = P^2/4 * A \tag{2}$$

$$\sigma = \sqrt{1/(N-1)} * \sum_{k=1}^N (distance_i - distance_{average}) \quad (3)$$

4.1.4 Acquiring Oscillation Number

Step 1: Divide the plotting graph each 90 angle.

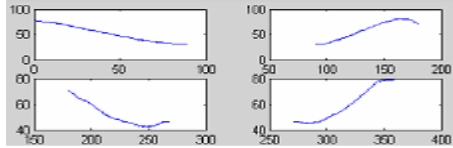


Fig. 4. Distance between tumor center to segment line (each 90)

Step2: Compute inclination of graph value.

We calculate inclination of graph value but in this approach, we need only positive and negative. In this step, we compute movement of graph for identification of tumor type especially rolobuate, irregular and speculate.

Step 3: Detection of changing inclination

```

for i = 2:360
    gab(1,i-1) = D(1,i)-D(1,i-1);
end
for i = 1:359
    if (gab(1,i) >=0);
        gab(1,i)=1;
    else
        gab(1,i)=-1;
    end
    if (gab(1,i) !=1);
        cnt=cnt+1;
    end
end
end

```

4.2 Diagnosis Method

In this paper instead of using one algorithm we have made a comparative analysis of the various algorithms and plot our findings. The results of the algorithm are completely based on the numbers of parameters and availability of numerical test data. However, in our approach applying 2 and 3 parameters consequently, BP and LVQ proves to be better with 3 parameters while implementing with 2 parameters could not show any drastic difference among them. The experimental results show that to get the more accurate classification of the breast cancer

tumor. we need to add more parameters in our training data input. Comparative analysis shows that the performance of BP and LVQ outperforms K-Means and SOM with large number of parameters and test data. The step by step process has been described as follows:

4.2.1 Adaptive SOM

Step 0 : Initialize reference vectors and learning rate, $a(0)$

Step 1 : While stopping condition = false do 2-6

Step 2 : For each training input vector x , do 3-4

Step 3 : Find J so that

$$\|X - W_k\| = \min \quad (4)$$

Step 4 : Update as follows:

$$W_j(\text{new}) = W_j(\text{old}) + aX - W_j(\text{old}) \quad (5)$$

Step 5 : Reduce learning rate

Step 6 : Test stopping condition:

The condition may specify a fixed number of iterations or the learning rate reaching a sufficiently small value.

4.2.2 Iterative LVQ

LVQ is a supervised competitive ANN which transforms high dimensional data to a two dimensional grid without regarding data topology. LVQ is intended for statistical classification. It uses pre-assigned cluster labels to data items to facilitate the two dimensional transformation so as to minimize the average expected misclassification probability. Each output unit represents a particular class. After training, an LVQ net classifies an input vector by assigning it to the same class as the output unit that has its weight vector (reference vector) closest to the input vector. The motivation for the algorithm for the LVQ net is to find the output unit that is closest to the input vector.

Step 0 : Initialize reference vectors and learning rate

Step 1 : While stopping condition = false do 2-6

Step 2 : For each training input vector x , do 3-4

Step 3 : Find J so that

Step 4 : Update as follows:

$$T = C_j, \text{ then; } W_j(\text{new}) = W_j(\text{old}) + aX - W_j(\text{old}) \quad (6)$$

$$T \neq C_j, \text{ then; } W_j(\text{new}) = W_j(\text{old}) - aX - W_j(\text{old}) \quad (7)$$

Step 5 : Reduce learning rate

Step 6 : Test stopping condition:

The condition may specify a fixed number of iterations or the learning rate reaching a sufficiently small value.

4.2.3 Multi-layer Perceptrons (Back Propagation)

Figure 5 shows the architectural graph of a multi-layer perceptron with some hidden layers and an output. To set the stage for a description of the multi-layer perceptron in its general form, the network shown here is fully connected. This means that a neuron in any layer of the network is connected to all the nodes in the previous layer. Back-propagation is one of supervised learning algorithms; it learns to adjust neuron’s weight, so that the error rate gets reduced between output value and target value. The numerical data of breast image like standard deviation, circularity, peak point number and ratio test are input through the input node in a forward direction, it passes through the hidden layer to adjust weight and calculate the error rate. The calculated error has been propagated backwards to all the nodes and readjusts the weight from left to right. This process has been repeated till the error ratio is acceptable as per the threshold. Finally, we get the output and as per target set by the user. The full step by step process has been de-scribed below:

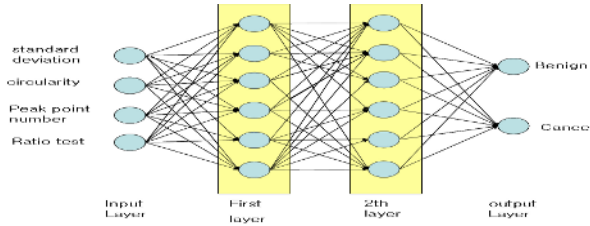


Fig. 5. Back-propagation modeling

Step 1: Setup all the node weight, Target, Input vector

Step 2: Compute between input and hidden layer’s node for calculation of output (follow equation 9 10)

$$O_k^o = X_k^u \forall k \tag{8}$$

$$O_i^m = g(h_k^u) = g\left(\sum_j w_{ij}^m O_j^{m-1} - \theta_i^m\right) \tag{9}$$

Step 3: Compare between output and Target and then calculate error (follow equation 11)

$$\delta_i^{m-1} = [T_i^m - O_i^m]g'(g_i^M) \tag{10}$$

Step 4: Backward computing of error for mediation weight from output layer’s node to input node (follow equation 12)

$$\delta_i^m = g'(g_i^{M-1}) \sum_i w_{ij} \delta_j^m (m = m, m - 1, m - 2 \dots 5) \tag{11}$$

Step 5: Compute alternative value of weight (follow equation 13 to 15)

$$\Delta w_{ij}^m = n \delta_i^m O_j^{m-1} \tag{12}$$

$$\Delta w_i^m = -n\delta_i^m \quad (13)$$

$$newweight = W_{ij}^{new} + \Delta W_{ij}, \theta_{ij}^{new} = \theta_{ij}^{old} + \Delta\theta_{ij} \quad (14)$$

Step 6: Go to step 2 until the error rate is enough.

5 Experimental Results

According to the comparative study the result of the back propagation algorithm has been proved to be the best for the detection and diagnosis of the ultrasound images of the breast cancer tumor. Given table describes the evaluation results of k-means, SOFM, LVQ and back-propagation (BP). This test has been done by using 200 test data, first we used 2 parameters (standard deviation and ratio analysis), and while using two parameters BP has given the best results with less number of error rates. Second comparison has been done by using the same test data or ultrasound images with three different parameters like standard deviation, circulatory and ratio analysis. This test has also given the similar type of results. K-means and SOFM presents very close error rates while BP and LVQ outperform the other algorithms with lesser number of inaccuracies. However, this performance evaluation has been done by using 200 test data only if the number of data will be increased or the parameters has been changed the results may be differ from these values.

Table 1. Comparative analysis of tumor

Name	Using 2 parameters of tumor				Using 3 parameters of tumor			
	K-Means	SOFM	LVQ	BP	K-Means	SOFM	LVQ	BP
Hit	86.6%	86.2%	87.6%	91.5%	87.4%	87.9%	91.5%	96.1%
Error	13.4%	13.8%	12.4%	8.5%	12.6%	12.1%	18.5%	3.9%
Total	100%	100%	100%	100%	100%	100%	100%	100%

6 Conclusions

Recently, it has been observed that ultrasound could be utilized for breast cancer diagnosis. It can be better conducted by using ultrasound technology with computer aided diagnosis. In our previous approach we describe the auto-segmentation and diagnosis method; however, to diagnosis the more abnormal tumor was sometimes difficult. The objective of this paper is to make comparative study of different neural network algorithms to detect a slight change of abnormal symptoms by tracing and comparing different test results systematically. This test has showed the experimental results of different neural network algorithms used for clustering with different numbers of parameters. An experiment shows that the BP and LVQ outperform the K-means and SOFM algorithms with lesser number of inaccuracies. This study will help the physicians

and radiologist to improve the efficiency in accurate detection of the image and appropriate diagnosis of the cancer tumor. Our future endeavor is to develop all the technique and tools, which can originate high reliable CAD system.

References

1. Insung Jung, D. Thapa, and Gi-Nam Wang, "Automatic segmentation and Diagnosis of breast lesions using morphology method based on ultrasound," *ICNC'05-FSKD'05, LNCS*, pp. 1079-1088, 2005.
2. D. Thapa, Insung Jung, and Gi-Nam Wang, "Agent Based Decision Making System under Emergency Circumstances," *ICNC'05-FSKD'05, LNCS*, pp. 888-892, 2005.
3. In-Sung Jung and Gi-Nam Wang, "Development of an adaptive-intelligent CAD framework," *Proceedings of HCI 2004*, 2003.
4. Tulio C. S.,A.C., Roque Da Silva, "A Neural Network Made of a Kohonen's SOM Cou-pledto a MLP Trained Via Back propagation for the Diagnosis of Malignant Breast Cancer from Digital Mammograms," *IEEE*, 1999.
5. Y. H. Chang, BinZheng, Xiao-Hui Wang, W.F.Good, "Computer-Aided Diagnosis of Breast Cancer Using Artificial Neural Networks: Comparison of Backpropagation and Genetic Algorithms," *IEEE*, 1999.
6. G. D. Tourassi, J. Y. Lo, G. D. Tourassi, C. E. Floyd Jr., "Self-oganizing map for cluster analysis of a breast cancer database," *Artificial Intelligence in Machine 27(2003)*, pp. 113-127, 2003.

New Learning Algorithm for Hierarchical Structure Learning Automata Operating in P-Model Stationary Random Environment

Yoshio Mogami

Faculty of Engineering, The University of Tokushima,
Minami-Jyosanjima, Tokushima 770-8506, Japan
moga@is.tokushima-u.ac.jp

Abstract. In this paper, based on the concept of Discretized Generalized Pursuit Algorithm (DGPA), the discretized generalized pursuit hierarchical structure learning algorithm is constructed which is applied to the hierarchical structure learning automata operating in the P-model stationary random environment. The efficacy of our algorithm is demonstrated by the numerical simulation, in which the hierarchical structure learning automata is applied to the problem of the mobile robots going through an unknown maze (the maze passage problem of mobile robots).

1 Introduction

The purpose of the learning automaton is to find the optimal solution fast and precisely in the unknown random environment. The studies of learning automaton have been advanced by many researchers[1]-[7]. But, there is such a drawback for the learning automaton that the convergence rate to the optimal solution is slow. In order to overcome this drawback, the hierarchical structure learning automata was proposed by Thathachar and Ramakrishnan[8], while a large number of learning algorithms have been constructed. In P-model stationary random environment, Estimator-type learning algorithm has a good learning property, which was first proposed by Thathachar and Sastry[9] and used the estimate of reward probabilities of the unknown random environment. On the other hand, by discretizing the action probabilities, Thathachar and Oommen[10] constructed discretized learning algorithm which increased the action probabilities faster than the continuous learning algorithms. Based on the concept of the discretized learning algorithm and of the pursuit algorithm, discretized pursuit learning algorithms (DPA) were designed by Oommen and Lanctôt[11] and Oommen and Agache[12]. DPA is such an algorithm that only the action probability whose current estimate of reward probability is maximal increases, and all of other action probabilities decrease. Hence, if the action having the maximal estimate is not optimal, then the automaton pursues a wrong action. In order to minimize the probability of pursuing an erroneous action, Agache and Oommen[13] proposed Generalized Pursuit Algorithm (GPA) and Discretized Generalized Pursuit Algorithm (DGPA), and verified the usefulness of them. Furthermore, they showed that DGPA was the fastest converging discretized pursuit estimator algorithm.

In this paper, by applying DGPA to hierarchical structure learning automata, the discretized generalized pursuit hierarchical structure learning algorithm is constructed. By the numerical simulation, the usefulness of our learning algorithm is shown, in which the hierarchical structure learning automata is applied to the problem of the mobile robots going through an unknown maze (the maze passage problem of mobile robots).

2 Hierarchical Structure Learning Automata Having Input and Output to Each Level

The hierarchical structure learning automata having input and output to each level is shown in Fig.1, where each automaton in hierarchy has r actions, respectively. In level 1, a single learning automaton Z and the unknown environment R_1 exist, in level 2, r learning automata $\{Z_1, Z_2, \dots, Z_r\}$ and the unknown environment R_2 exist, and in level 3, r^2 learning automata $\{Z_{11}, Z_{12}, \dots, Z_{1r}, Z_{21}, \dots, Z_{rr}\}$ and the unknown environment R_3 . In Fig.1, the number of levels is 3, but generally it is N , and, in level s ($s = 1, 2, \dots, N$), r^{s-1} learning automata $\{Z_{i_1 i_2 \dots i_{s-1}} (i_k = 1, 2, \dots, r; k = 1, 2, \dots, s - 1)\}$ and the unknown environment R_s exist.

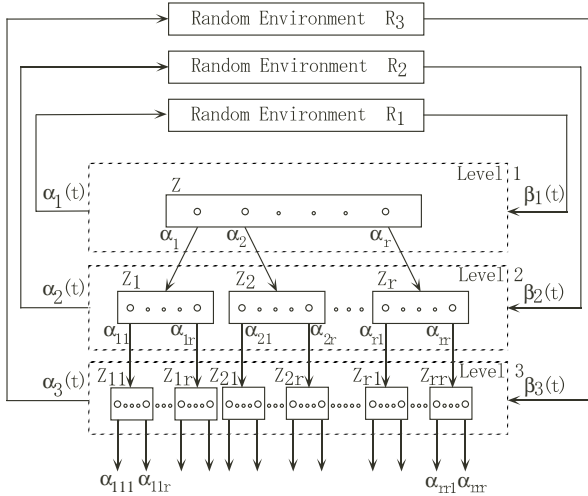


Fig. 1. Hierarchical structure learning automata

Hierarchical structure learning automata works as follows. At time t , in level 1, learning automaton Z selects an action α_{j_1} based on the action probability distribution $(p_1(t), p_2(t), \dots, p_r(t))'$, and the output $\alpha_1(t) = \alpha_{j_1}$ is given to the unknown environment R_1 . Then, R_1 returned the reward $\beta_1(t) = \beta_{j_1}$ to Z . The action α_{j_1} actuates the learning automaton Z_{j_1} in level 2, then, Z_{j_1} selects an action

$\alpha_{j_1 j_2}$ based on the action probability distribution $(p_{j_1 1}(t), p_{j_1 2}(t), \dots, p_{j_1 r}(t))'$ and the output $\alpha_2(t) = \alpha_{j_1 j_2}$ is given to the unknown environment R_2 , and, R_2 returned the reward $\beta_2(t) = \beta_{j_1 j_2}$ to Z_{j_1} . The action $\alpha_{j_1 j_2}$ actuates the learning automaton $Z_{j_1 j_2}$ in level 3. After all of the above procedures have been completed to the bottom level, then hierarchical structure learning automata update the action probability distributions according to the learning algorithm by using the rewards obtained from the unknown environments. Then, time t is set to be $t + 1$, and, learning automaton Z in level 1 selects an action. With repeating such a cycle, hierarchical structure learning automata proceeds with learning.

The sequence of actions $\{\alpha_{j_1}, \alpha_{j_1 j_2}, \dots, \alpha_{j_1 j_2 \dots j_N}\}$ having been chosen by N automata is called the path. Let $\phi_{j_1 j_2 \dots j_N}$ denote the path, and let $\pi_{j_1 j_2 \dots j_N}(t)$ denote the path probability that the path $\phi_{j_1 j_2 \dots j_N}$ is chosen at time t . Then, $\pi_{j_1 j_2 \dots j_N}(t) = p_{j_1}(t)p_{j_1 j_2}(t) \dots p_{j_1 j_2 \dots j_N}(t)$.

3 P-Model Stationary Random Environment

In this paper, we shall consider the hierarchical structure learning automata operating in the P-model stationary random environment. In P-model stationary random environment, the reward from the environment for the output of the learning automaton is generated based on the unknown constant reward probability. The success reward is 1 and the penalty reward is 0, respectively. Therefore, at time t , for the output $\alpha_s(t) = \alpha_{i_1 i_2 \dots i_s}$ of level s ($s = 1, 2, \dots, N$) of the hierarchical structure learning automata, the reward $\beta_s(t) = \beta_{i_1 i_2 \dots i_s} = 1$ or 0 is given from the level s unknown environment R_s based on the unknown constant reward probability $d_{i_1 i_2 \dots i_s}$.¹

Let $\chi_{i_1 i_2 \dots i_N}$ denote the path reward probability for the path $\phi_{i_1 i_2 \dots i_N}$. Then, $\chi_{i_1 i_2 \dots i_N} = d_{i_1} d_{i_1 i_2} \dots d_{i_1 i_2 \dots i_N}$. In this paper, we shall consider the learning behaviors of the hierarchical structure learning automata under the P-model stationary random environment which has the following property.

Property 1. The optimal path $\phi_{j_1^* j_2^* \dots j_N^*}$ satisfying (1) exists uniquely:

$$\chi_{j_1^* j_2^* \dots j_N^*} > \chi_{i_1 i_2 \dots i_N}, \quad \max\{|j_1^* - i_1|, |j_2^* - i_2|, \dots, |j_N^* - i_N|\} > 0. \quad (1)$$

Furthermore, the P-model stationary random environment considered in this paper also includes the following rather general environment satisfying (2).

Property 2. For some learning automaton $Z_{j_1^* j_2^* \dots j_{s-1}^*}$ ($\exists s < N$) on the optimal path $\phi_{j_1^* j_2^* \dots j_N^*}$, there exists the case that the following condition (2) holds:

$$d_{j_1^* j_2^* \dots j_{s-1}^* j_s^*} \leq d_{j_1^* j_2^* \dots j_{s-1}^* i_s} \quad (\exists i_s \neq j_s^*). \quad (2)$$

¹ With probability $d_{i_1 i_2 \dots i_s}$, the reward $\beta_s(t) = 1$ is given, and with probability $1 - d_{i_1 i_2 \dots i_s}$, the reward $\beta_s(t) = 0$ is given.

4 DGPA-Type Hierarchical Structure Learning Algorithm

In this section, based on the concept of Discretized Generalized Pursuit Algorithm (DGPA) proposed by Agache and Oommen [13], we will construct the new hierarchical structure learning algorithm.

Let us define the estimate $\hat{d}_{i_1 i_2 \dots i_s}(\cdot)$ of the reward probability $d_{i_1 i_2 \dots i_s}$, which is used in DGPA, as follows:

$$\hat{d}_{i_1 i_2 \dots i_s}(t+1) = \frac{R_{i_1 i_2 \dots i_s}(t+1)}{z_{i_1 i_2 \dots i_s}(t+1)} = \frac{R_{i_1 i_2 \dots i_s}(t) + \beta_{i_1 i_2 \dots i_s}}{z_{i_1 i_2 \dots i_s}(t) + 1}, \quad (i_s = 1, 2, \dots, r) \quad (3)$$

where $z_{i_1 i_2 \dots i_s}(t)$ is the selected number of the output $\alpha_{i_1 i_2 \dots i_s}$ of level s by the time t , and, $R_{i_1 i_2 \dots i_s}(t)$ is the total sum of the reward given from the random environment to the output. Note that the P-model stationary random environment satisfies the property P.2. Because DGPA converges to the action having the maximal reward probability[13], then, in such random environment, DGPA converges to the action $\alpha_{j_1^* j_2^* \dots j_{s-1}^* i_s}$ which is not on the optimal path $\phi_{j_1^* j_2^* \dots j_N^*}$ in the learning automaton $Z_{j_1^* j_2^* \dots j_{s-1}^*}$, that is, DGPA cannot find the optimal path. In order to resolve this problem, let us define the reward parameter $\gamma_{i_1 i_2 \dots i_{s-1}}(t) = (\gamma_{i_1 i_2 \dots i_{s-1} 1}(t), \gamma_{i_1 i_2 \dots i_{s-1} 2}(t), \dots, \gamma_{i_1 i_2 \dots i_{s-1} r}(t))'$. The each element of the reward parameter is given as follows:

(i) At bottom level N :

$$\gamma_{i_1 i_2 \dots i_N}(t) = \hat{d}_{i_1 i_2 \dots i_N}(t). \quad (4)$$

(ii) At level s ($s = 1, 2, \dots, N - 1$):

$$\gamma_{i_1 i_2 \dots i_s}(t) = \hat{d}_{i_1 i_2 \dots i_s}(t) \max_{l_{s+1}} \{ \gamma_{i_1 i_2 \dots i_s l_{s+1}}(t) \}. \quad (5)$$

Now, let us denote the parameters as follows: S = resolution parameter (the number of division of the range $[0,1]$ of action probability), $K(t)$ = number of actions having the higher estimates of the reward probabilities than that of the selected action at time t , and, Δ = smallest step size $\Delta = 1/(rS)$. Then, the new hierarchical structure learning algorithm is constructed as follows.

⟨⟨ DGPA-type hierarchical structure learning algorithm ⟩⟩

Step 1. Set $t = 0$ and $p_{i_1 i_2 \dots i_s}(0) = 1/r$ ($s = 1, 2, \dots, N$) Initial values

$\{\hat{d}_{i_1 i_2 \dots i_s}(0)\}$ of the estimates of the reward probabilities are calculated by trying sequentially each action a certain number of times.

Step 2. Denote the action probabilities $\tilde{p}_{i_1 i_2 \dots i_s}(t)$ ($s = 1, 2, \dots, N$) on the path $\phi_{i_1 i_2 \dots i_N}$ which has the maximal path probability. Then, if $\tilde{p}_{i_1 i_2 \dots i_s}(t) > \rho, \forall s$, then stop this algorithm, else go to Step 3.

Step 3. Update the action probability distribution by the following reinforcement scheme: At time t , let us assume that the actions $\{\alpha_{j_1}, \alpha_{j_1 j_2}, \dots, \alpha_{j_1 j_2 \dots j_s}, \dots, \alpha_{j_1 j_2 \dots j_N}\}$ on the path $\phi_{j_1 j_2 \dots j_s \dots j_N}$ are selected, and, from the environment in each level, the rewards $\{\beta_{j_1}, \beta_{j_1 j_2}, \dots, \beta_{j_1 j_2 \dots j_s}, \dots, \beta_{j_1 j_2 \dots j_N}\}$ are given respectively.

- (i) If $\gamma_{j_1 j_2 \dots j_{s-1} j_s}(t) < \gamma_{j_1 j_2 \dots j_{s-1} i_s}(t)$ ($i_s \neq j_s$),

$$p_{j_1 j_2 \dots j_{s-1} i_s}(t+1) = \min\{p_{j_1 j_2 \dots j_{s-1} i_s}(t) + \frac{\Delta}{K(t)}, 1\}. \tag{6}$$

- If $\gamma_{j_1 j_2 \dots j_{s-1} j_s}(t) > \gamma_{j_1 j_2 \dots j_{s-1} i_s}(t)$ ($i_s \neq j_s$),

$$p_{j_1 j_2 \dots j_{s-1} i_s}(t+1) = \max\{p_{j_1 j_2 \dots j_{s-1} i_s}(t) - \frac{\Delta}{r - K(t)}, 0\}. \tag{7}$$

$$p_{j_1 j_2 \dots j_{s-1} j_s}(t+1) = 1 - \sum_{i_s \neq j_s} p_{j_1 j_2 \dots j_{s-1} i_s}(t+1). \tag{8}$$

- (ii) By (3), evaluate the estimates of the reward probabilities

$$\{\hat{d}_{j_1 j_2 \dots j_s}(t+1) : s = 1, 2, \dots, N\}.$$

- (iii) Set $t = t+1$, and, based on (4) and (5), calculate the reward parameters

$$\{\gamma_{j_1 j_2 \dots j_s}(t) : s = 1, 2, \dots, N\}. \text{ Go to Step 2.}$$

5 Maze Passage Problem of Mobile Robots

We apply our hierarchical structure learning algorithm to the maze passage problem of mobile robots shown in Fig.2 (in which the maze is comprised of 2×2 small rooms). In the maze, there are many gates each of which is opened with some probability (opening probability) that is unknown to the mobile robotos, and a sequence of gates is called a path. In Fig.2, $\{A1, A2, B1, B2, B3, B4\}$ are the gates, and the paths are $A1 \rightarrow B1$, $A1 \rightarrow B2$, $A2 \rightarrow B3$ and $A2 \rightarrow B4$, respectively. The task assigned to the mobile robots is to find the path which is the easiest to go through the maze, The control center decides the action to be taken by the mobile robot appeared at the starting point (Start), and recieves the information concerning whether the mobile robot has succeeded in passing the gates or not. To find the best path, the control center exploits the learning performance of the proposed algorithm (DGPA-type hierarchical structure learning algorithm).

In this simulation, we consider the more large maze, which has 4×4 small rooms. (Due to the limitation of space, I don't go into details of this simulation. Interested readers are kindly asked to attend my presentation.)

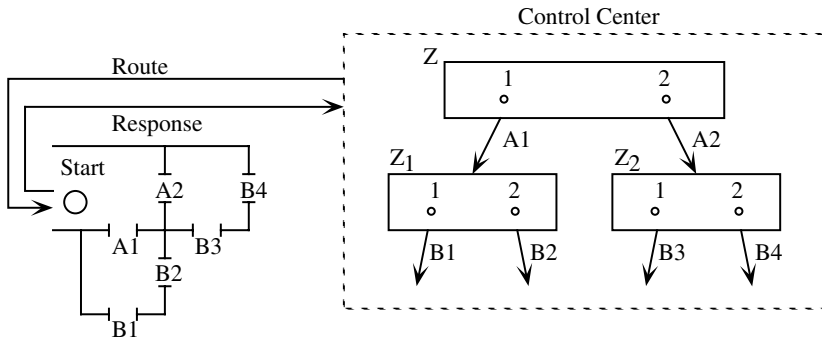


Fig. 2. Maze passage problem

6 Conclusions

Based on the concept of Discretized Generalized Pursuit Algorithm (DGPA)[13], we have proposed a new hierarchical structure learning algorithm of the hierarchical structure learning automata operating in the general P-model stationary random environment in each level. In order to investigate whether the proposed algorithm can be successfully utilized in the P-model stationary random environment, we apply our hierarchical structure learning algorithm to the maze passage problem of mobile robots, and, we confirmed the efficacy of the proposed hierarchical structure learning algorithm.

References

1. Tsetlin, M.L.: On the Behavior of Finite Automata in Random Media. *Avtomatika i Telemekhanika* **22** (1961) 1345–1354
2. Varshavskii, V.I., Vorontsova, I.P.: On the Behavior of Stochastic Automata with Variable Structure. *Automation and Remote Control* **24** (1963) 327–333
3. Lakshmivarahan, S.: *Learning Algorithms Theory and Applications*. Springer-Verlag, Berlin Heidelberg New York (1981)
4. Baba, N.: *New Topics in Learning Automata Theory and Applications*. Springer-Verlag, Berlin Heidelberg New York (1985)
5. Narendra, K.S., Thathachar, M.A.L.: *Learning Automata: An Introduction*. Englewood Cliffs, NJ: Prentice-Hall (1989)
6. Poznyak, A.S., Najim, K.: *Learning Automata and Stochastic Optimization*. Springer-Verlag, Berlin Heidelberg New York (1997)
7. Thathachar, M.A.L., Sastry, P.S.: *Networks of Learning Automata Techniques for Online Stochastic Optimization*. Kluwer Academic Publisher (2004)
8. Thathachar, M.A.L., Ramakrishnan, K.R.: A Hierarchical System of Learning Automata. *IEEE Trans. Syst. Man Cybern.* **SMC-11** (1981) 236–241
9. Thathachar, M.A.L., Sastry, P.S.: A Class of Rapidly Converging Algorithms for Learning Automata. *IEEE International Conference on Cybernetics and Society* (1984) 602–606
10. Thathachar, M.A.L., Oommen, B.J.: Discretized Reward-Inaction Learning Automata. *J. Cybern. Inf. Sci.* (1979) 24–29
11. Oommen, B.J., Lanctôt, J.K.: Discretized Pursuit Learning Automata. *IEEE Trans. Syst. Man Cybern.* **20** (1990) 931–938
12. Oommen, B.J., Agache, M.: Continuous and Discretized Pursuit Learning Scheme: Various Algorithms and Their Comparison. *IEEE Trans. Syst. Man Cybern. B* **31** (2001) 277–287
13. Agache, M., Oommen, B. J.: Generalized Pursuit Learning Scheme: New Families of Continuous and Discretized Learning Automata. *IEEE Trans. Syst. Man Cybern. B* **32** (2002) 738–749

A TFN-Based AHP Model for Solving Group Decision-Making Problems

Jian Cao, Gengui Zhou, and Feng Ye

Institute of Information Intelligence and Decision-making Optimization,
Zhejiang University of Technology, Hangzhou 310032, China
jcao@iipc.zju.edu.cn

Abstract. Because the expert(s) usually give the judgment with an uncertainty degree in general decision-making problems, combined the analytic hierarchy process (AHP) with the basic theory of the triangular fuzzy number (TFN), a TFN-based AHP model is suggested. The proposed model makes decision-makers' judgment more accordant with human thought mode and derives priorities from TFN-based judgment matrices regardless of their consistency. In addition, formulas of the model are normative, they can be operated by programming easily and no human intervention is needed while applying the model-based software system. The results of an illustrative case indicate that, by applying the proposed model, fair and reasonable conclusions are obtained and the deviation scope of the priority weight of every decision element is given easily.

1 Introduction

Based on the pair-to-pair comparison values for a set of objects, the analytic hierarchy process (AHP) elicits a corresponding priority vector that represents preference [1-3]. Pairwise comparison values are the judgments obtained from an appropriate 1~9 semantic scale, while in practice, the decision-makers usually give these values with an uncertainty degree rather than precise ratings. Combined with the fuzzy set theory, a fuzzy AHP model was approached by Chang and Zhu, et al. to cope with those ambiguous and vague judgments [4,5]. However, there would still be a problem on calculation if the judgment matrices were not in consistency. Different from those papers, combined the AHP with basic theory of the triangular fuzzy number (TFN) [6-8], and adopted the logarithmic least square method to formulate the calculation of relative weights vector for decision-making, this paper proposes a new TFN-based AHP model that is sufficiently robust to permit conflict and imprecision under a group decision-making environment.

2 The TFN-Based AHP Model

The core of the AHP is the conformation of judgment matrix by 1~9 semantic scale and their according reciprocals [1]. Such judgment often ignores the vagueness of

people’s mind. If the comparative weight of decision elements C_i and C_j is not decided with a precise rating, and what we can only get is that the scope of its value is from l to u and the most possibility of its value is m , such judgment is usually expressed as a TFN (l, m, u) . Consequently, the TFN-based AHP model is established, which is a modification of the AHP model combined with the concept and algorithm of the TFN [6,7].

In the AHP, the solution $\omega, \omega=(\omega_1, \omega_2, \dots, \omega_n)^T$, to Eq. (1) is called the priority vector of the logarithmic least square method [9,10]. After the normalization of ω , the normalized priority weights vector of decision criteria $w, w=(w_1, w_2, \dots, w_n)^T$, is obtained.

$$Z = \sum_{i=1}^n \sum_{j=1}^n [\ln a_{ij} - \ln(\frac{\omega_i}{\omega_j})]^2. \tag{1}$$

Taking different opinions of experts into account, the pairwise comparison ratio r_{ij} of judgment matrix R has more than one value. Hence, by Eq. (1), we have

$$Z = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^b (\ln a_{ijk} - \ln \omega_i + \ln \omega_j)^2. \tag{2}$$

where b is a constant representing the number of experts who take part in this group decision-making problem.

Take the partial derivative with respect to $\omega_p, p=(1, 2, \dots, n)$, for the function in Eq. (2), and let

$$\frac{\partial}{\partial \omega_p} \left[Z = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^b (\ln a_{ijk} - \ln \omega_i + \ln \omega_j)^2 \right] = 0, \tag{3}$$

i.e.,

$$\sum_{\substack{k=1 \\ j \neq p}}^b \sum_{j=1}^n (\ln a_{ijk} - \ln \omega_p + \ln \omega_j) = \sum_{\substack{k=1 \\ i \neq p}}^b \sum_{i=1}^n (\ln a_{ipk} - \ln \omega_i + \ln \omega_p). \tag{4}$$

By Eq. (4), and noting that $\ln a_{ipk} = \ln(1/a_{ijk})$, it follows that

$$n \ln \omega_p - \sum_{j=1}^n \ln \omega_j = \frac{1}{b} \sum_{j=1}^n \sum_{k=1}^b \ln a_{ijk}. \tag{5}$$

Let $\omega_p = (l_p, m_p, u_p), \omega_j = (l_j, m_j, u_j)$ and $a_{ijk} = (r_{ijk}, s_{ijk}, t_{ijk})$, we get

$$n(\ln l_p, \ln m_p, \ln u_p) \oplus \sum_{j=1}^n (\ln \frac{1}{u_j}, \ln \frac{1}{m_j}, \ln \frac{1}{l_j}) = \frac{1}{b} \sum_{j=1}^n \sum_{k=1}^b (\ln r_{ijk}, \ln s_{ijk}, \ln t_{ijk}). \tag{6}$$

Hence, we obtain

$$\begin{cases} l_p = \left(\prod_{k=1}^b \prod_{j=1}^n r_{pjk} \right)^{\frac{1}{bn}} \cdot \left(\prod_{j=1}^n u_j \right)^{\frac{1}{n}} \\ m_p = \left(\prod_{k=1}^b \prod_{j=1}^n s_{pjk} \right)^{\frac{1}{bn}} \cdot \left(\prod_{j=1}^n m_j \right)^{\frac{1}{n}} \\ u_p = \left(\prod_{k=1}^b \prod_{j=1}^n t_{pjk} \right)^{\frac{1}{bn}} \cdot \left(\prod_{j=1}^n l_j \right)^{\frac{1}{n}} \end{cases} \quad (7)$$

Normalizing ω to w , $w = (w_1, w_2, \dots, w_n)^T$, $\sum_{p=1}^n w_p = 1$, and according to $t_{ijk} \cdot r_{jik} = 1$, we

have

$$\begin{cases} l'_p = \frac{l_p}{\sum_{p=1}^n u_p} = \frac{\left(\prod_{k=1}^b \prod_{j=1}^n \prod_{i=1}^n t_{ijk} \right)^{\frac{1}{bn^2}}}{\sum_{i=1}^n \left(\prod_{k=1}^b \prod_{j=1}^n t_{ijk} \right)^{\frac{1}{bn}}} \cdot \left(\prod_{k=1}^b \prod_{j=1}^n r_{pjk} \right)^{\frac{1}{bn}} \\ m'_p = \frac{m_p}{\sum_{p=1}^n m_p} = \frac{1}{\sum_{i=1}^n \left(\prod_{k=1}^b \prod_{j=1}^n s_{ijk} \right)^{\frac{1}{bn}}} \cdot \left(\prod_{k=1}^b \prod_{j=1}^n s_{pjk} \right)^{\frac{1}{bn}} \\ u'_p = \frac{u_p}{\sum_{p=1}^n l_p} = \frac{\left(\prod_{k=1}^b \prod_{j=1}^n \prod_{i=1}^n r_{ijk} \right)^{\frac{1}{bn^2}}}{\sum_{i=1}^n \left(\prod_{k=1}^b \prod_{j=1}^n r_{ijk} \right)^{\frac{1}{bn}}} \cdot \left(\prod_{k=1}^b \prod_{j=1}^n t_{pjk} \right)^{\frac{1}{bn}} \end{cases} \quad (8)$$

From above, it indicates that the proposed model makes experts' judgment more accordant with human thought mode and synthesizes group opinions more effectively. The model can also derive priorities from TFN-based judgment matrices, regardless of their consistency. Moreover, the arithmetic formulas of the model are simple and normative, they can be operated by programming easily and no human intervention is needed while applying the model-based software system.

3 The Process of Decision-Making Based on the Proposed Model

Generally, the application of the proposed TFN-based AHP model for group decision-making is as follows:

- Step 1. Determine several experts who qualified for the decision-making of the problem.
- Step 2. Structure the hierarchical form from the top (the objective of the problem) through the intermediate levels (criteria on which subsequent levels depend) to the lowest level which usually contains the list of alternatives.
- Step 3. From the top down, by pairwise comparisons of decision elements, construct a set of TFN-based pairwise comparison matrices, and determine the priority weights of criteria and sub-criteria in the intermediate levels.

- Step 4. Calculate the priority weights vector of alternatives under each lowest sub-criterion.
- Step 5. Determine the global priority weights vector of all alternatives.

4 An Illustrative Case

Combined with a group decision-making for the selection of potential manager in personnel promotion problem, the application of the TFN-based AHP model is demonstrated in this section. For easy interpretation, only three-hierarchy structure is constructed in the illustrative example, as shown in Fig.1. Four criteria including morality, ability, health and cooperation are proposed. Three experts take part in this group decision-making problem for the selection of the best one from three evaluated candidates A, B and C. Ranking of these candidates is obtained by programming with the proposed model in Matlab environment. Moreover, the disadvantages and advantages of candidates are mentioned accordingly.

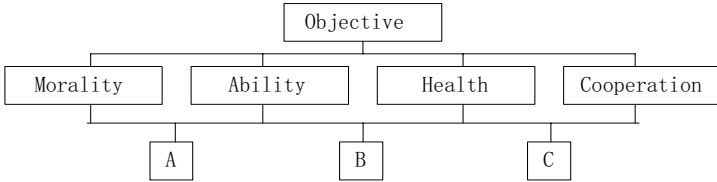


Fig. 1. A decision hierarchy of personal promotion problem

	<i>Morality</i>	<i>Ability</i>	<i>Health</i>	<i>Cooperation</i>
<i>Morality</i>	(1,1,1)	(2/5,1/2,3/5) (3/7,1/2,4/7) (2/5,3/5,4/5)	(1/2,2/3,3/4) (1/2,2/3,3/4) (3/5,2/3,4/5)	(5/3,2,7/3) (2,7/3,5/2) (5/3,2,5/2)
<i>Ability</i>	(5/3,2,5/2) (7/4,2,7/3) (5/4,5/3,5/2)	(1,1,1)	(1,4/3,3/2) (5/4,3/2,5/3) (1,4/3,3/2)	(7/2,4,9/2) (4,5,6) (4,9/2,5)
<i>Health</i>	(4/3,3/2,2) (4/3,3/2,2) (5/4,3/2,5/3)	(2/3,3/4,1) (3/5,2/3,4/5) (2/3,3/4,1)	(1,1,1)	(3,7/2,4) (7/2,4,9/2) (3,4,9/2)
<i>Cooperation</i>	(3/7,1/2,3/5) (2/5,3/7,1/2) (2/5,1/2,3/5)	(2/9,1/4,2/7) (1/6,1/5,1/4) (1/5,2/9,1/4)	(1/4,2/7,1/3) (2/9,1/4,2/7) (2/9,1/4,1/3)	(1,1,1)

Based on the pairwise comparison between four criteria, three experts give the TFN-based judgment matrix (9), from which the priority weights of these criteria are obtained in table 1. Then, the pair-to-pair comparisons of three candidates under four criteria are obtained respectively. For space limitation, only judgment matrix (10) under morality criterion is given here, and the corresponding priority weights of these

Table 1. The priority weights of four criteria

Criterion	The priority weight
Morality	(0.1706,0.2015,0.2288)
Ability	(0.3437,0.4009,0.4576)
Health	(0.2764,0.3092,0.3612)
Cooperation	(0.0798,0.0884,0.1013)

candidates are obtained and shown in table 2. Finally, the global priority weights of these candidates are calculated, $w_A=(0.2406,0.3387,0.4588)$, $w_B=(0.2476,0.3440,0.4895)$ and $w_C=(0.2340,0.3173,0.4468)$.

$$\begin{matrix}
 & A & B & C \\
 A & \left[\begin{matrix} (1,1,1) \\ (2/3,1,3/2) \\ (1/2,3/4,4/3) \\ (1/2,2/3,1) \end{matrix} \right. & \left. \begin{matrix} (2/3,1,3/2) \\ (3/4,4/3,2) \\ (1,3/2,2) \\ (1,1,1) \end{matrix} \right. & \left. \begin{matrix} (1,4/3,3/2) \\ (3/2,2,5/2) \\ (2/3,1,4/3) \\ (2/3,1,3/2) \\ (1,3/2,2) \\ (3/4,1,3/2) \end{matrix} \right] \\
 B & & & \\
 C & \left[\begin{matrix} (2/3,3/4,1) \\ (2/5,1/2,2/3) \\ (3/4,1,3/2) \end{matrix} \right. & \left. \begin{matrix} (2/3,1,3/2) \\ (1/2,2/3,1) \\ (2/3,1,4/3) \end{matrix} \right. & \left. \begin{matrix} (1,1,1) \end{matrix} \right]
 \end{matrix} \tag{10}$$

Table 2. The priority weights of three candidates under four criteria accordingly

Criteria	The priority weight		
	A	B	C
Morality	(0.3063,0.3975,0.4831)	(0.2510,0.3196,0.4226)	(0.2341,0.2829,0.3576)
Ability	(0.2795,0.3145,0.3498)	(0.2836,0.3306,0.3878)	(0.3117,0.3549,0.4244)
Health	(0.2531,0.3191,0.3898)	(0.3250,0.4021,0.5005)	(0.2313,0.2788,0.3450)
Cooperation	(0.2830,0.3832,0.4675)	(0.2191,0.2573,0.3416)	(0.2884,0.3595,0.4561)

According to the comparative rule of the TFN [6], we get the following conclusion: candidate B is the best solution, A ranks next to B and better than C. From table 2, the rankings of these candidates under each criterion are obtained, and the performances of them are shown obviously. For example, the health ranks B first, but B's morality and cooperation need to be improved. In practice, there is lower level of criteria on which these four criteria are depended, and the performance of every candidate under each lower criterion could be indicated clearly by using the proposed model.

5 Conclusion

A new fuzzy AHP model based upon the AHP and basic theory of the triangular fuzzy number is provided in this paper. To apply the model to group decision-making

problems, it makes experts' judgment more accordant with human thought mode, and it doesn't seem worth bothering consistency of the judgment matrix in the process of evaluation. Meanwhile, the consistent degree of judgments or opinions provided by different experts is indicated clearly. In addition, the arithmetic formulas of the model are simple and normative and can be operated by programming easily. Then, no human intervention is needed and a quick decision is feasible.

Acknowledgements

This research work was partially supported by Research Planning Fund of Zhejiang Provincial Education Department, China (No.20040580), Zhejiang Provincial Nature Science Foundation, China (No.Y104171 and NO.602096) and Zhejiang Provincial Philosophy & Society Science Foundation, China (No.NX05GL07).

References

1. Saaty, T.L.: The Analytic Hierarchy Process. McGraw-Hill, New York (1980)
2. Saaty, T.L.: How to Make a Decision: The AHP. *European Journal of Operational Research*. 48 (1990) 9-26
3. Millet, I., Saaty, T.L.: On the Relativity of Relative Measures – Accommodating Both Rank Preservation and Rank Reversals in the AHP. *European Journal of Operational Research*. 121 (2000) 205-212
4. Chang, D.Y.: Application of the Extent Analysis Method on Fuzzy AHP. *European Journal of Operational Research*. 95 (1996) 649-655
5. Zhu, K.J., Yu, J.: A Discussion on Extent Analysis Method and Applications of Fuzzy AHP. *European Journal of Operational Research*. 116 (1999) 450-456
6. Kaufmann, A., Gupta, M.M.: Introduction to Fuzzy Arithmetic Theory and Applications. Van Nostrand Reinhold, New York (1991)
7. Dubois, D., Prade, H.: Fuzzy Sets and Systems: Theory and Applications. Academic Press, New York (1980)
8. Yu, C.S.: A GP-AHP Method for Solving Group Decision-making Fuzzy AHP Problems. *Computer & Operation Research*. 29 (2002) 1969-2001
9. Grawford, G., Williams, C.A.: A Note on the Analysis of Subjective Judgment Matrices. *Journal of Mathematical Psychology*. 29 (1985) 387-405
10. Kwiesielewicz, M., Uden, E.V.: An Additional Result of Monsuur's Paper about Intrinsic Consistency Threshold for Reciprocal Matrices. *European Journal of Operational Research*. 140 (2002) 88-92

A Tactics for Robot Soccer with Fuzzy Logic Mediator

Jeongjun Lee¹, Dongmin Ji¹, Wonchang Lee², Geuntaek Kang²,
and Moon G. Joo²

¹ Dept. of Electronic Engineering, Pukyong National Univ.

² Dept. of Telecommunication Engineering, Pukyong National Univ.,
Busan, Korea

Abstract. This paper presents a tactics using fuzzy logic mediator that selects proper robot action depending on the positions and the roles of adjacent two robots. Conventional Q-learning algorithm, where the number of states increases exponentially with the number of robots, is not suitable for a robot soccer system, because it needs so much calculation that processing cannot be accomplished in real time. A modular Q-learning algorithm reduces a number of states by partitioning the concerned area, where mediator algorithm for cooperation of robots is used additionally. The proposed scheme not only reduces a number of a calculation but also combines a robot action selection with robot cooperation by means of fuzzy logic system.

1 Introduction

The robot soccer competition is classified as HuroSot, KheperaSot, MiroSot, NaroSot, RoboSot and SimuroSot by specifications of robot and game rule [1]. The Middle League MiroSot Environment is used in this paper, where 5 robots of 7.5 x 7.5 x 7.5cm size are used and 2 attackers, 2 defenders, and 1 goalkeeper is used. And host computer gathers information of ball and robot positions by image processing, determines traveling algorithm, and order robots according to the strategy and tactics [2][3].

As an algorithm for strategy and tactics, Q-learning algorithm has been reported. The advantage of the algorithm is the capability of automated learning where parameters of the robots are adapted by trial and errors. However, this algorithm needs many trials and errors for stabilization of Q values and the number of states increases exponentially when the number of individual increases. Therefore, this algorithm is hard to apply to robot soccer system in real time. On the other hand, Modular Q-Learning algorithm [4] was suggested which reduces the number of operation but it needs an additional mediator module to decide robot action when more than 2 robots are engaged in a partitioned area.

This paper proposes a cooperation scheme using fuzzy logic mediator. The proposed scheme is suitable for robot soccer strategy and tactics because this algorithm computes simultaneously both traveling algorithm of robot and cooperation algorithm among robots, and therefore, the number of calculation is much less than modular Q-learning.

2 Traveling Algorithm

Traveling algorithm, in other words, navigation of a soccer robot, creates path for robot and demands robot to follow the path. Actions in traveling algorithm are SHOOT, ASSIST, DEFEND and SCREEN.

2.1 Algorithm for Attacking Robots

SHOOT algorithm creates a path to drive a ball to opponent goal area. For the purpose, Limit-cycle Navigation scheme [5] is adopted in this paper, which creates circular path from arbitrary position more simply than Uni-Vector Field algorithm [6].

ASSIST algorithm describes a robot action to move to symmetric position from the ball and to be ready to shoot when the role is changed to SHOOT.

Host computer can swap the role of two attackers by the tactics using the difference of distances from the ball, location of two robots, and so on.

2.2 Algorithm for Defending Robots

DEFEND algorithm creates path to kick a ball away to opponent area. It is similar to SHOOT algorithm, but a target position is not opponent goal, but opponent area.

SCREEN algorithm describes a robot action that helps goalkeeper robot. When one defending robot is selected for SCREEN action, the robot blocks the way to goal to help goalkeeper robot and the other defending robot takes a role of DEFEND to interrupt opponent attacker's action.

In addition, host computer can swap the role of two defenders by the tactics using the difference of distances from the ball, location of two robots, and so on. Furthermore, attacker becomes defender, and defender becomes attacker by the host computer's tactics.

3 Tactics Using Fuzzy Logic Mediator

This section describes a strategy to decide the roles between 2 attackers, between 2 defenders, and between attacker and defender. By the use of the information gathered by host computer, it gives role to each robot. The roles of 4 robots are determined according to the ball position, opponent goal position, the robot position, and so on. Fuzzy logic system is used as mediator between robots when more than one robot is responsible for the area.

Basic strategy in robot soccer game is area-defense by a responsible robot for that area. Each robot has its responsible area. For example, when a robot is given a role of attacker 1, then the robot is responsible for the allocated area in Fig. 1 until host computer gives a new role.

However, when a ball is located in the overlapped region as shown shaded in Fig. 1, more than one robot should cooperate. In region 1, one of 2 attackers becomes a shooter and the other, assistor. In region 3, one of the defender screens and the other defends. In region 2, the roles of an attacker and a defender can be changed for effectiveness and time saving.

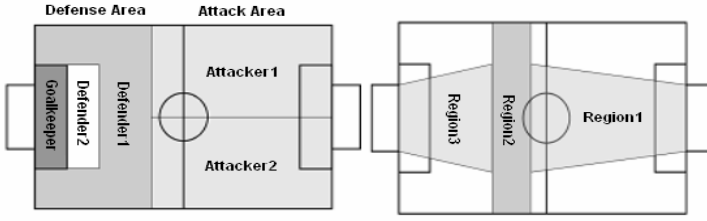


Fig. 1. Responsible regions and overlapped regions

3.1 Fuzzification and Defuzzification

Input variables for fuzzy logic mediator are d , Θ_b , and Θ_g defined as

$$d = d_m - d_n \tag{1}$$

$$\theta_b = |\theta_m| - |\theta_n| \quad \theta_g = |\theta_m'| - |\theta_n'| \tag{2}$$

where d_m is the distance between robot m and the ball; Θ_m is the angle between robot m and the ball; Θ'_m is the angle between robot m and the goal. The membership functions for d , Θ_b , Θ_g are defined by triangular type as shown in Fig. 2, where BN, SN, Z, SP, and BP represent Big Negative, Small Negative, Zero, Small Positive, and Big Positive, respectively.

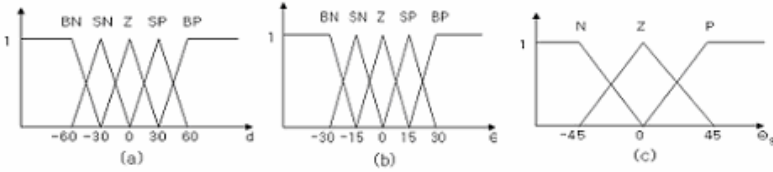


Fig. 2. Membership functions for (a) d , (b) Θ_b , and (c) Θ_g

The output membership function is defined by single-tone type as shown in Fig. 3. In region 1, the output of the fuzzy logic mediator is the degree offensive posture and negative value corresponds to SHOOT, positive value, ASSIST. In region 3, fuzzy logic mediator gives either DEFEND or SCREEN to the robot. In region 2, whether attacker and defender change their role or not is determined by fuzzy logic mediator. The product inference and center-average defuzzification are used as follows [7].

$$(\text{Robot_action}) = \frac{\sum_i \mu^i \times \Omega^i}{\sum_i \mu^i} \quad \mu^i = \mu_d^i \times \mu_{\theta_b}^i \times \mu_{\theta_g}^i \tag{3}$$



Fig. 3. Membership functions in (a) Region 1, (b) Region 2, and (c) Region 3

3.2 Fuzzy Rules

Region 1 is the overlapped area between two attackers. One takes the role of SHOOT, and the other takes ASSIST. Fuzzy rules are as follows and the rule tables are shown in Fig. 4.

$$R^i: \text{If } d^i \text{ is } A^i \text{ and } \theta_b^i \text{ is } B^i \text{ and } \theta_g^i \text{ is } C^i, \text{ Then Robot_action is } \Omega^i \quad (4)$$

Region 2 is the overlapped area between an attacker and a defender. Output fuzzy set is composed of CHANGE (exchange role between two robots) and NO CHANGE (do not exchange). Fuzzy rules are as equation (4) and the rule tables are shown in Fig. 5.

Region 3 is the overlapped area between two defenders. Hence, θ_g doesn't have to be considered and the fuzzy rules are as follows and the rule table is shown in Fig. 6.

$$R^i: \text{If } d^i \text{ is } A^i \text{ and } \theta_b^i \text{ is } B^i, \text{ Then Robot_action is } \Omega^i \quad (5)$$

$\theta_a^i \backslash \theta_d^i$	BN	SN	Z	SP	BP
BN	Shoot	Shoot	Shoot	Shoot	No change
SN	Shoot	Shoot	Shoot	No change	No change
Z	Shoot	Shoot	No change	No change	No change
SP	Shoot	No change	No change	No change	Assist
BP	No change	No change	No change	Assist	Assist

$\theta_a^i \backslash \theta_d^i$	BN	SN	Z	SP	BP
BN	Shoot	Shoot	No change	No change	No change
SN	Shoot	No change	No change	No change	No change
Z	No change	No change	No change	No change	No change
SP	No change	No change	No change	No change	Assist
BP	No change	No change	No change	Assist	Assist

(a) when θ_g is 'N'

(b) when θ_g is 'Z'

$\theta_a^i \backslash \theta_d^i$	BN	SN	Z	SP	BP
BN	Defense	Defense	Defense	No change	No change
SN	Defense	Defense	No change	No change	No change
Z	Defense	No change	No change	No change	Screen
SP	No change	No change	No change	Screen	Screen
BP	No change	No change	Screen	Screen	Screen

(c) when θ_g is 'P'

Fig. 4. Fuzzy rules in Region 1

$\theta_a^i \backslash \theta_d^i$	BN	SN	Z	SP	BP
BN	No change	No change	No change	No change	No change
SN	No change	No change	No change	No change	No change
Z	No change	No change	No change	No change	No change
SP	No change	No change	No change	No change	Change
BP	No change	No change	No change	Change	Change

$\theta_a^i \backslash \theta_d^i$	BN	SN	Z	SP	BP
BN	No change	No change	No change	No change	No change
SN	No change	No change	No change	No change	No change
Z	No change	No change	No change	No change	No change
SP	No change	No change	No change	No change	Change
BP	No change	No change	No change	Change	Change

(a) when θ_g is 'N'

(b) when θ_g is 'Z'

$\theta_a^i \backslash \theta_d^i$	BN	SN	Z	SP	BP
BN	No change	No change	No change	No change	No change
SN	No change	No change	No change	No change	Change
Z	No change	No change	No change	Change	Change
SP	No change	No change	Change	Change	Change
BP	No change	Change	Change	Change	Change

(c) when θ_g is 'P'

Fig. 5. Fuzzy rules in Region 2

$\frac{\theta_b}{d}$	BN	SN	Z	SP	BP
BN	Defense	Defense	Defense	No change	No change
SN	Defense	Defense	No change	No change	No change
Z	Defense	No change	No change	No change	Screen
SP	No change	No change	No change	Screen	Screen
BP	No change	No change	Screen	Screen	Screen

Fig. 6. Fuzzy rules in Region 3

4 Simulation

The proposed algorithm is implemented and tested by SimuroSot, one of the robot soccer game using only simulation. In Fig. 7-9, big triangle, big rectangle, and big circle are starting point of the simulation and small ones are their trajectories. As shown in the simulation results, the proposed scheme is proven to be feasible and to have a good performance.

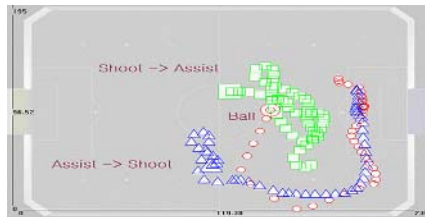


Fig. 7. Simulation result of SHOOT and ASSIST in region 1

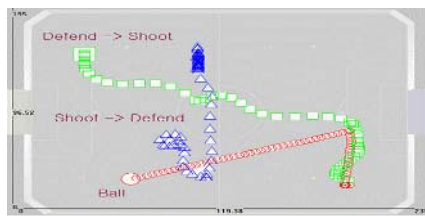


Fig. 8. Simulation result of SHOOT and DEFEND in region 2



Fig. 9. Simulation result of DEFEND and SCREEN in region 3

5 Conclusion

In this paper, a tactics using fuzzy logic system is proposed where a fuzzy logic mediator selects proper robot action depending on the status of adjacent two robots and the position of a ball. The proposed algorithm reduces a number of a calculation by combining the selection of robot action with cooperation among robots.

By SimuroSot simulation, it is shown to be feasible with good performance.

References

1. Jonghwan Kim, "Robot Soccer System," dae-young-sa, 2000.
2. Jonghwan Kim and 8 people, "Robot Soccer Engineering," brainkorea, 2002.
3. J.H. Kim, K. C. Kim, D. H. Kim, Y. J. Kim and P. Vadakkepat, "Path Planning and Role Selection Mechanism for Soccer Robots," IEEE Int'l Conf. on Robotics and Automation, Belgium, pp. 3216-3221, May 1998.
4. K. H. Park, Y. J. Kim and J. H. Kim, "Modular Q-learning based Multi-agent Cooperation for Robot Soccer," Robotics and Autonomous Systems, vol. 35, no. 2, pp. 109-122, May. 2001.
5. Dong-Han Kim and Jong-Hwan Kim, "Limit-cycle Navigation Method for Soccer Robot," International Conference on Artificial Intelligence, Las Vegas, June 25-28, 2001.
6. D. H. Kim, Y. J. Kim, K. C. Kim, J. H. Kim and P. Vadakkepat, "Vector Field Based Path Planning and Petri-net Based Role Selection Mechanism with Q-learning for the Soccer Robot System," Intelligent Automation and Soft Computing, Vol. 6, No. 1, 2000.
7. Li-Xin Wang, A course in Fuzzy Systems and Control, Prentice Hall PTR, 1997.

Gait Control for Biped Robot Using Fuzzy Wavelet Neural Network

Pengfei Liu and Jiuqiang Han

School of Electronics and Information Engineering, Xi'an Jiaotong University,
Xi'an 710049, China
pfliu77@yahoo.com.cn

Abstract. A new reference trajectory of walking on the ground for a five-link biped robot, considering both the SSP and the DSP, is developed firstly. And a fuzzy wavelet neural network controller to generate walking gaits to follow the reference trajectories is presented subsequently. Furthermore, an error compensation algorithm is presented for high accuracy. The reference trajectories are designed by solving the coefficients of time polynomial functions of the trajectories of the hip and the swing tip, through the constraint equations. The FWN controller is trained as inverse kinematic model of the biped robot by back-propagation algorithm offline. Simulation results show that the FWN controller can generate the gaits following the reference trajectories as close as possible, and the error compensation algorithm can decrease the error rapidly by iterative calculation.

1 Introduction

The design of reference trajectory for gait cycle is a crucial step for biped motion control. Zarrugh *et al.* [1] have investigated the walking pattern for a biped robot by recording human kinematic data. McGeer [2] described a natural walking pattern generated by the passive interaction of gravity and inertia on a downhill slope. Silva *et al.* [3] have investigated the required actuator power and energy by adjusting walking parameters. To ensure the dynamic stability of a biped robot, Hirai *et al.* [4] have proposed methods of walking pattern synthesis based on zero moment point (ZMP). But most studies have focused on motion generation during the single support phase (SSP), and the double support phase (DSP) has received less attention. The DSP plays an important role in keeping a biped walking stably with a wide range of speeds, and thus can't be neglected. Many researchers have shown that fuzzy set and neural network theory can solve dynamic biped locomotion problems [5,6].

With the reference trajectories including both the SSP and the DSP, this paper presents a new control scheme by using a fuzzy wavelet neural network as the controller, and an error compensation algorithm was presented for high accuracy. The presented control scheme can generate the gait that satisfies specified constraints such as the step length, crossing clearance and walking speed. Simulation results are given for a five-link biped robot.

2 Biped Robot Model and Reference Trajectory

The walking machine BLR-G1 robot [7] is used as the simulation model. As shown in Fig.1, this robot consists of five links, namely, a torso, two lower legs and two upper legs. This robot has no feet (no ankle). The motion of the biped is limited to only the sagittal plane (X-Z plane). The ground condition is assumed to be rigid and nonslip. The contact between the tip and the ground is assumed to be a single point.

The dynamic equations of the biped model are given in [7] as follows:

$$A(\theta)\ddot{\theta} + B(\theta)\dot{h}(\theta) + Cg(\theta) = DT \tag{1}$$

Where $\theta = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]^T$, $T = [\tau_1, \tau_2, \tau_3, \tau_4]^T$ τ_i is the torque at the i Th joint, and θ_i and $\dot{\theta}$ are the position and velocity of link i . Other parameters are expressed in [7].

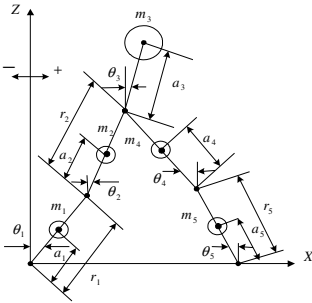


Fig. 1. Five-link Biped model

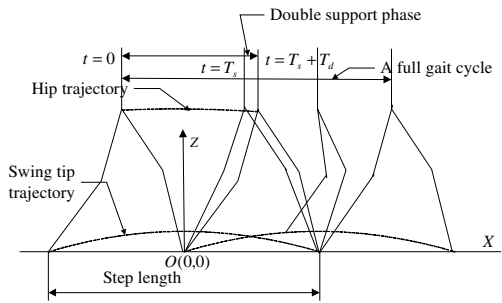


Fig. 2. Full gait cycle of a planar five-link biped

A complete step can be divided into a SSP and a DSP [8]. Mu presented a method for gait synthesis of a five-link biped walking on level ground during SSP and DSP, but it can't keep the continuity when switching the swing limb. The joint angle profiles can be determined when compatible trajectories for hip and the tip of the swing limb can be known. We prescribe that both knees only bend in one direction, thus, the joint angle profiles can be uniquely determined by the given hip and swing limb trajectories. The torso is kept at the upright position that the trajectory $\theta_3(t) = 0$ in both the SSP and the DSP. The trajectory of the tip of the swing limb is denoted by the vector $X_a : (x_a(t), z_a(t))$, where $(x_a(t), z_a(t))$ is the coordinate of the swing limb tip position in the coordinate system in Fig.2. They are shown below:

$$\begin{cases} x_a(t) = a_0 + a_1t + a_2t^2 + a_3t^3 \\ z_a(t) = b_0 + b_1t + b_2t^2 + b_3t^3 + b_4t^4 + b_5t^5 \end{cases}, 0 \leq t \leq T_s \tag{2}$$

There are four basic quantities in the SSP: step length S_l , step period for the SSP T_s , maximum height of the tip of the swing limb H_m and its location S_m . The constraint equations are described as follows:

$$x_a(0) = -\frac{S_l}{2}, x_a(T_s) = \frac{S_l}{2}, z_a(0) = 0, z_a(T_s) = 0, x_a(T_m) = S_m, \tag{3}$$

$$z_a(T_m) = H_m, \dot{z}_a(T_m) = 0, \dot{x}_a(0) = 0, \dot{z}_a(0) = 0, \dot{x}_a(T_s) = 0, \dot{z}_a(T_s) = 0$$

Next the trajectory of the hip is designed. The trajectory of the hip should be divided into the SSP and the DSP, which are separately denoted by the coordinate as $x_{hs} : (x_{hs}(t), z_{hs}(t))$ in the SSP and $x_{hd} : (x_{hd}(t), z_{hd}(t))$ in the DSP. Two third order polynomial functions are used to describe $x_{hs}(t)$ and $x_{hd}(t)$.

$$\begin{aligned} x_{hs}(t) &= c_0 + c_1 t + c_2 t^2 + c_3 t^3, 0 \leq t \leq T_s, \\ x_{hd}(t) &= d_0 + d_1 t + d_2 t^2 + d_3 t^3, 0 \leq t \leq T_d \end{aligned} \quad (4)$$

We assume $z_{hs}(t)$ and $z_{hd}(t)$ as a constraint Z_h during the whole step cycle. Considering the constraints of repeatability, continuity and stability of the gait, we can develop the constraint equations as follows:

$$\begin{aligned} x_{hs}(0) &= -S_{s0}, x_{hd}(T_d) = \frac{1}{2} S_l - S_{s0}, \dot{x}_{hs}(0) = V_{h1}, \dot{x}_{hd}(T_d) = V_{h1}, x_{hd}(0) = S_{d0} \\ & , x_{hs}(T_s) = S_{d0}, \dot{x}_{hs}(T_s) = V_{h2}, \dot{x}_{hd}(0) = V_{h2} \end{aligned} \quad (5)$$

Where S_{s0} and S_{d0} are positions of the hip at the beginning of the SSP and DSP, we assume S_{s0} as $-\frac{S_l}{4}$; V_{h1} is the hip velocity at the beginning of the SSP and V_{h2} is the hip velocity at the beginning of the DSP. V_{h1} And V_{h2} can be determined by obtaining the largest stability margin through ZMP criterion [8].

With the designed hip and swing tip trajectories and the biped model, the joint angle profiles can be determined by the following equations:

$$\begin{cases} \theta_1(t) = \arcsin\left(\frac{A_1 C_1 - B_1 \sqrt{A_1^2 + B_1^2 - C_1^2}}{A_1^2 + B_1^2}\right) \\ \theta_2(t) = \arcsin\left(\frac{A_1 C_2 + B_1 \sqrt{A_1^2 + B_1^2 - C_2^2}}{A_1^2 + B_1^2}\right) \\ \theta_3(t) = 0 \\ \theta_4(t) = \arcsin\left(\frac{A_4 C_3 - B_4 \sqrt{A_4^2 + B_4^2 - C_3^2}}{A_4^2 + B_4^2}\right) \\ \theta_5(t) = \arcsin\left(\frac{A_4 C_4 + B_4 \sqrt{A_4^2 + B_4^2 - C_4^2}}{A_4^2 + B_4^2}\right) \end{cases} \quad (6)$$

Where $C_1 = \frac{A_1^2 + B_1^2 + r_1^2 - r_2^2}{2r_1}$, $C_2 = \frac{A_1^2 + B_1^2 + r_2^2 - r_1^2}{2r_2}$, $C_3 = \frac{A_4^2 + B_4^2 + r_2^2 - r_1^2}{2r_2}$, $C_4 = \frac{A_4^2 + B_4^2 + r_1^2 - r_2^2}{2r_1}$, $B_4 = Z_h$. For the SSP: $A_1 = x_{hs}(t) - x_a(t)$, $B_1 = Z_h - Z_a(t)$, $A_4 = x_{hs}(t)$, and for the DSP: $A_1 = -\frac{1}{2} S_l + x_{hd}(t)$, $B_1 = Z_h$, $A_4 = x_{hd}(t)$.

3 FWN Control System

In order to generate the gait in the reference trajectory given above, a control system is presented, as shown in Fig.3. θ_d and $\dot{\theta}_d$ are expected joint angle and angular velocities, θ and $\dot{\theta}$ are actual value of joint angle and angular velocities.

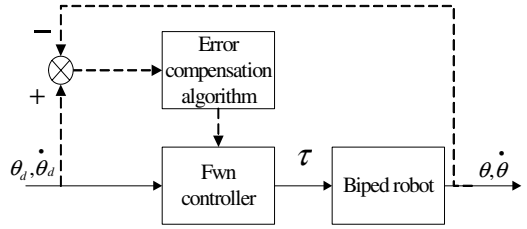


Fig. 3. Control system for five-link biped

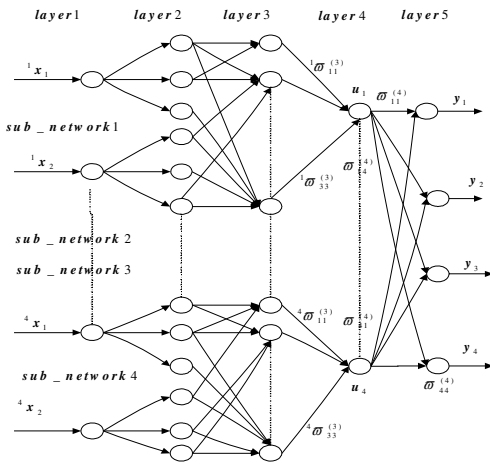


Fig. 4. Architecture of FWN controller

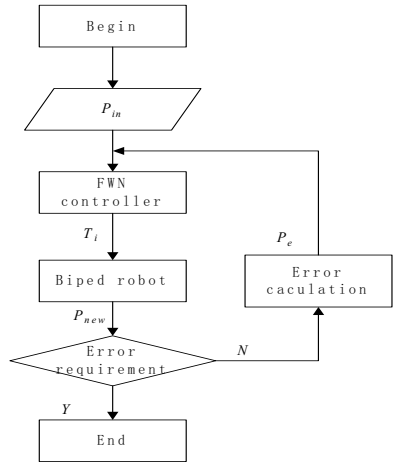


Fig. 5. Error compensation algorithm

The FWN controller is shown as Fig.4. The whole network consists of four sub-networks. Layer 2 is the fuzzification layer in which each node represents a fuzzy set. There are three fuzzy sets for each component of x , namely, P, Z and N. We choose mother wavelet function as

$$\psi(x) = (1 - x^2)e^{-\frac{x^2}{2}} \tag{7}$$

$${}^k O_{ij}^{(2)} = {}^k I_{ij}^{(2)} = 2^{k m_{ij}/2} \psi(2^{k m_{ij}} x_i - {}^k n_{ij}), (k = 1, 2, 3, 4; i = 1, 2; j = 1, 2, 3) \tag{8}$$

Where ${}^k m_{ij}$ are scalable coefficients, and ${}^k n_{ij}$ are shiftable coefficients. And Let ${}^k I_j^{(i)}$ be input of the j Th node of layer i in sub-network k . Let ${}^k O_j^{(i)}$ be output of the j Th node of layer i in sub-network k . The parameters $\varpi_{kl}^{(4)}$ 、 ${}^k \varpi_{ij}^{(3)}$ 、 ${}^k m_{ij}$ and ${}^k n_{ij}$ need to be trained in the algorithm. And the learning scheme can be educed by error back-propagation. The evaluation function is chosen as

$$J_{off} = \frac{1}{2} \sum_{l=1}^4 (y_l - y_l^-)^2 \tag{9}$$

As the fuzzy wavelet neural network is trained enough times, the error will not decrease any more. In order to meet the higher error requirement of the biped control, we present an error compensation algorithm, which decrease the error by iterative calculation. In the Fig.6, P_i is the expected input of the biped posture, T_i is the output of the FWN controller, P_{new} is the output of forward kinematics model of the biped robot, and $P_e = P_{new} - P_i$. The error compensation is realized by using $P_i - P_e$ as the $i+1$ th input of the FWN controller.

4 Simulations

The joint profiles for five-link biped walking on level ground with both the SSP and the DSP are simulated. The parameters are given as follows:

$S_l = 0.7m$, $T_s = 0.6s$, $T_d = 0.1s$, $H_m = 0.06m$, $S_m = 0m$, $S_{d0} = 0.14m$, $V_{h1} = 0m/s$, $V_{h2} = 1m/s$, $Z_h = r_1 + r_2 - H_m$.

Fig.6 is the stick diagram of the biped robot with the prescribed conditions above. Put the data in Fig.6 as the training set, and train the network with the presented algorithm. Fig.7 shows joints angle profiles in Mu's method during the SSP and the DSP. It shows that the joint angles have a little problem with repeatability and continuity when switching the swing limb. Fig.8 shows joints angle profiles in our method. It is seen that the joint angles are repeatable and continuous. Fig.9 (a) shows the error of θ_1 of the trained network and the error after third time iterative calculation. The error maximum of θ_1 of the trained network is $0.0025rad$, and the error maximum after compensation is $3.441 \times 10^{-4}rad$. Fig.9 (b) shows the error of θ_2 of the trained network and the error. The error maximum of θ_2 of the trained network is $0.0021rad$, and the error maximum after compensation is $3.376 \times 10^{-4}rad$.

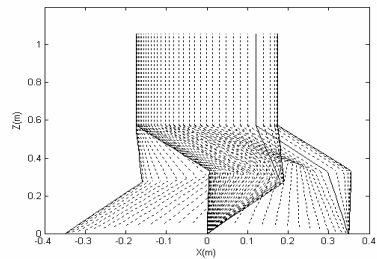


Fig. 6. Stick program of the biped robot

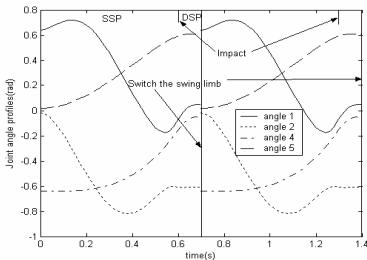


Fig. 7. Joint angle profiles in Mu's method

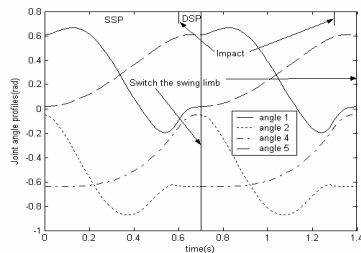


Fig. 8. Joint angle profiles in our method

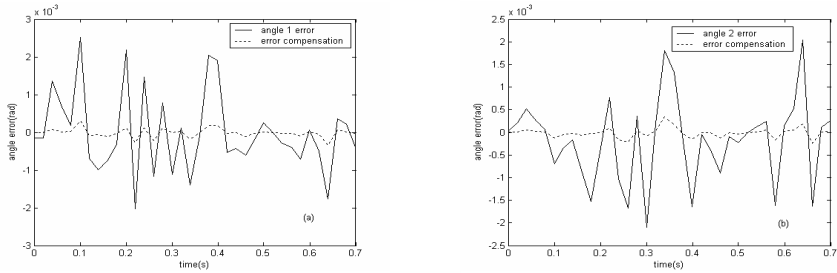


Fig. 9. Angle error of θ_1 and θ_2

5 Conclusion

In this paper, systematic reference trajectories are developed for five-link biped walking in the sagittal plane. Unlike the previous work focusing on the SSP, our reference trajectories include both the SSP and the DSP, which give the biped a wider range of walking speeds and more stable locomotion. Furthermore, a fuzzy wavelet neural network controller for gait synthesis is presented in this paper. The uncertainty of the network size in the conventional neural network learning scheme has been overcome by the use of FWN. The trained fuzzy controller can generate control sequences and drive the biped along the reference trajectories given in the Section 2, including both the SSP and the DSP. The simulation results show that the designed FWN controller can generate the gait according to the reference trajectories well, and error compensation algorithm can improve the accuracy rapidly. The proposed controller and compensation algorithm can follow reference trajectory as closely as possible.

References

1. Zarrugh, M.Y., Radcliffe, C.W.: Computer generation of human gait kinematics. *Journal of Biomechanics*, vol.12, (1979) 99-111
2. McGeer, T.: Passive walking with knees. *Proc. IEEE Int. Conf. Robotics and Automation*, (1990) 1640-1645
3. Silva, F.M., Machado, J.A.T.: Energy analysis during biped walking. *Proc. IEEE Int. Conf. Robotics and Automation*, (1999) 59-64
4. Hirai, K., Hirose, M., Haikawa, Y., Takenaka, T.: The development of Honda humanoid robot. *Proc. IEEE Int. Conf. Robotics and Automation*, (1998) 1321-1326
5. Murakami, S., Yamamoto, E., Fujimoto, K.: Fuzzy Control of Dynamic Biped Walking Robot. *Proc. IEEE Conf. Fuzzy Systems*, vol.1, no.4 (1995) 77-82
6. Juang, J.G.: Fuzzy Neural Network Approaches for Robotic Gait Synthesis. *IEEE Transactions on System, Man, Cybernetics*, vol.30 (2000) 594-601
7. Furusho, J., Masubuchi, M.: Control of a Dynamical Biped Locomotion System for Steady Walking. *Journal of Dynamic Systems, Measurement, and Control*, vol.108 (1986) 111-118
8. Mu Xiuping, Wu Qiong: Sagittal Gait Synthesis for a Five-Link Biped Robot. *Proc. Of the 2004 American Control Conference*, (2004) 4004-4009

A New Approach for Regression: Visual Regression Approach*

Deyu Meng^{1,2}, Chen Xu², and Wenfeng Jing²

¹ Institute for Information and System Science, Xi'an Jiaotong University,
Xi'an 710049, China

mengdeyu1978@163.com

² School of Science, Xi'an Jiaotong University, Xi'an 710049, China

Abstract. The regression is one of the fundamental problems in data mining, which is central to many applications of information technology. Various approaches have been presented for regression problem nowadays. However, many problems still exist, such as efficiency and model selection problem. This paper proposes a new approach to regression problem, visual regression problem (VRA) in order to resolve these problems. The core idea is to transfer the regression problem to classification problem based on Ancona theorem, which gives the mathematical equivalence between two problems; and then use visual classification approach, which is an efficient classification approach developed based on mimicking human sensation and perception principle, to solve the transformed classification problem and get an implicit regression function; and finally utilize some mathematical skills to obtain the explicit solution of the regression problem. We also provide a series of simulations to demonstrate that the proposed approach is not only effective but also efficient.

1 Introduction

Data Mining (DM) is the main procedure of knowledge discovery from database (KDD), aiming at discovering useful knowledge from large collections of data. The regression is one of fundamental problems in DM. Its end is to find the relationship (a function normally) between the input data set and the output set generated from an unknown but fixed distribution. The regression problem has attracted extensive attention in past decades due to its very wide-scope applications in scientific research and engineering [1][2][3][4][5].

Various approaches for regression have been developed. To sum up, three types of methods are mainly adopted prevalently: the interpolation-like methods (say, the splines methods [1]), the statistical methods (say, the parameter and non-parameter regression [2]) and the computational intelligent methods ([3][4][5]). Many algorithms for the approaches are developed for particular applications and most can satisfy the needs. However, when the applications are of huge sized or very high-dim data sets, the efficiency problem, even the feasibility

* This research was supported by the NSFC project under contract 10371097.

problem, is always encountered. Besides, most of algorithms are very sensitive to sample neatness and model parameters, and there is no general rule to specify the model parameters involved. Consequently, model selection problem is also a big challenge for most of the regression approaches.

In this paper, we will initiate a new approach for regression, with an expectation of resolving the problems mentioned above. The core idea is first to transfer a regression problem to a classification problem based on the Ancona theorem which shows that in certain sense a regression problem is equivalent to a classification problem. Then we adopt visual classification approach (VCA) [7], which is an efficient classification approach developed by mimicking our human sensation and perception principle and has a robust model selection strategy, to obtain the classification discriminant function, (i.e. the implicit solution of the regression problem). Some mathematical skill is then used to transform the implicit solution into explicit solution of the regression problem. Finally we use some simulations to verify the feasibility and efficiency of the new approach.

The paper is outlined as follows. Ancona theorem, which proves that a regression problem can be transferred into a classification problem, is introduced in section 2. In section 3, we give a brief review for visual classification approach. A new approach for regression, named visual regression approach (VRA), will be formulated in section 4, extended from the VCA. A series of simulation results are provided to demonstrate the effectiveness of the VRA, in contrast to the prevalently used SVR. The conclusion is given finally.

2 Ancona Theorem

This section is devoted to constructing relationship between regression problem and classification problem. A regression problem is the problem of determining a function from a given sample pairs $D = \{(x_i, y_i) : i = 1, 2, \dots, l\}$ generated from a unknown but fixed distribution on $X \times R$. To transform a regression problem into a classification problem, we observe the fact that if we remove the regression data upward and downward, the margin between them naturally forms a "tube" whose central curve gives the regression function. This provides us a hint that one can find the expected regression function through finding the "tube" and the corresponding central curve. The regression data removing upward and downward with width ε forms the following two datasets:

$$D^+ = (x_i, y_i + \varepsilon), i = 1, 2, \dots, l, \quad D^- = (x_i, y_i - \varepsilon), i = 1, 2, \dots, l;$$

where $\{x_i, y_i\}_{i=1}^l$ are the given training samples. With the datasets D^+ and D^- , N.Ancona ([6]) verified the following conclusion:

Theorem 1. (*Ancona, 1999*): *The regression problem for dataset $D = \{x_i, y_i\}_{i=1}^l$ is equivalent to the two-class classification problem for dataset $D^+ \cup D^-$ in the following sense: If $f(x)$ is a discriminant function between of $D^+ \cup D^-$, then it is an ε -insensitive function of D , that is*

$$-\varepsilon \leq y_i - f(x_i) \leq \varepsilon, i = 1, 2, \dots, l. \quad (1)$$

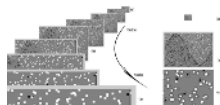


Fig. 1. Observing the data set from close to far away. A perfect discriminant function is perceived at a proper distance.

Ancona's theorem suggests us a method to obtain an ε -insensitive regression function of D from the two-class classification surface of $D^+ \cup D^-$. We thus can apply any classification approach to obtain a classification surface of $D^+ \cup D^-$, say, $f(x, y, \sigma^*) = 0$, which offers us an implicit regression function. The problem is then how to get an explicit expression of the regression function from $f(x, y, \sigma^*) = 0$, and also, how to choose a suitable ε in the implementation.

We adopt the following method for choosing ε : To choose ε is large enough so that D^+ and D^- is separable, say,

$$\varepsilon > \left(\frac{\max_{1 \leq i \leq l}(y_i) - \min_{1 \leq i \leq l}(y_i)}{2} \right), \quad (2)$$

and apply any centralized classification method (namely, the classification surface is nearly located on the center of the margin between two classes) like SVM [5] and VCA [7].

VCA is a very novel classification approach [7] which has a very high efficiency and robust model selection strategy deduced from visual sensation principle. Due to many advantages of this approach, we adopt it as the classification approach in our approach. In the next section, a brief review of it is given.

3 A Brief Review of VCA

Different from the existing approaches for classification, which have been developed mainly based on exploring the intrinsic structure of data set itself, the VCA classifies data by mimicking the way we sense and perceive the data (as shown in Fig 1). Below we give a brief review of the VCA.

Let be given a two-label training sample set $\mathcal{D}_l = \{x_i^+, +1\}_{i=1}^{N_+} \cup \{x_i^-, -1\}_{i=1}^{N_-}$, which is generated independently from an unknown but fixed distribution $F(x, y)$. The VCA determines a discriminant rule h through following three steps.

The first step of the VCA is viewing the data set \mathcal{D}_l as an imaginary image. The method is to view every positive (negative) training sample x_i^+ (x_i^-) as a spot light with unit strength $\delta(x - x_i)$ ($-\delta(x - x_i)$) (here $(\delta(x))$ is a Dirac generalized function) and consequently all the data \mathcal{D}_l form an image:

$$f(x, \mathcal{D}_l) = \frac{1}{N_+ + N_-} \left(\sum_{i=1}^{N_+} \delta(x - x_i^+) + \sum_{i=1}^{N_-} (-\delta(x - x_i^-)) \right). \quad (3)$$

The second step is embedding the image $f(x, \mathcal{D}_l)$ into a continuous family $F(x, \sigma, \mathcal{D}_l)$, which are gradually smoother versions of the original image by

mimicking our sensation and perception. The smoother versions of image are the projections of the original image to the retina in different distances, given by $f(x, \sigma, \mathcal{D}_l) = f(x, \mathcal{D}_l) * g(x, \sigma)$, where ‘*’ denotes the convolution operation, $g(x, \sigma)$ is the Gaussian function. Then, the structure of the original image is analyzed through exploring connections and variations among the images of the family $\{f(x, \sigma, \mathcal{D}_l) : \sigma \geq 0\}$. At each fixed scale σ_0 , we propose to classify the data through defining the discriminant function by $\text{sgn}(f(x, \sigma, \mathcal{D}_l))$ and the classification boundary (or classification surface) by $\Gamma = \{x : f(x, \sigma, \mathcal{D}_l) = 0\}$.

The final step is finding a suitable scale σ^* , at which the $\text{sgn}(f(x, \sigma^*, \mathcal{D}_l))$, can perfectly classify the data, through certain model selection strategy. It has been proved that the expected scale σ^* must be in a bounded interval $[\varepsilon, E]$ [7]. Through adopting the scheme deduced from Weber’s law in psychophysiology [8], we obtain a reasonable discretization scheme: $\sigma_i - \sigma_{i-1} = 0.029\sigma_{i-1}, i = 1, 2, \dots, M$. Applying any cross validation approach [9] to $\{\sigma_i : i = 1, 2, \dots, M\}$ we can give the expected scale σ^* and discriminant function: $h(x, \sigma^*) = \text{sgn}(f(x, \sigma^*, \mathcal{D}_l))$ Then the classification surface obtained can be expressed as $f(x, y, \sigma^*) = 0$.

The only question left is how we can transfer the implicit regression function $f(x, y, \sigma^*) = 0$ into the explicit expression of the output y .

4 Visual Regression Approach

To get an explicit expression of the output y , we need some mathematics. Let

$$\begin{aligned} D_{**} &= \{x_i, z_i\}_{i=1}^l = \{x_i, \ln(y_i + C)\}_{i=1}^l \\ D_{**}^+ &= \{x_i, z_i + \varepsilon\}_{i=1}^l, D_{**}^- = \{x_i, z_i - \varepsilon\}_{i=1}^l, i = 1, 2, \dots, l \end{aligned} \quad (4)$$

where C is a positive constant such that $\min_{1 \leq i \leq l} (y_i + C) > 0$, and let $\theta_i = (x_i, z_i) \in R^{n+1}, i = 1, 2, \dots, l$. We define

$$d(\theta_1, \theta_2) = (\|x_1 - x_2\|_2^2 + |z_1 - z_2|)^{\frac{1}{2}}. \quad (5)$$

Then we can show that the classification problem for data set $D^+ \cup D^-$ is equivalent to that of $D_{**}^+ \cup D_{**}^-$ (in the sense that the related discriminant functions can be mutually transformable), and $d(\theta_1, \theta_2)$ is a distance of R^{n+1} .

Now, instead of using the dataset $D^+ \cup D^-$, we apply the VCA to the dataset $D_{**}^+ \cup D_{**}^-$ and yield the implicit ε -insensitive regression function detailed by

$$f(\theta, \sigma^*) = \sum_{\theta_i \in D_{**}^+} \frac{1}{(\sigma^* \sqrt{2\pi})^{\frac{n}{2}}} e^{-\frac{d(\theta, \theta_i)}{2(\sigma^*)^2}} + \sum_{\theta_i \in D_{**}^-} -\frac{1}{(\sigma^* \sqrt{2\pi})^{\frac{n}{2}}} e^{-\frac{d(\theta, \theta_i)}{2(\sigma^*)^2}} = 0 \quad (6)$$

Based on the above expression, we then can prove the following theorem.

Theorem 2. Assume that ε is chosen to satisfy (2) and $f(\theta, \sigma^*) = 0$ is defined by (6). Then the function $g(x, \sigma^*)$ defined by (7)-(8) is a regression function of the original data set $D = \{x_i, y_i\}_{i=1}^l$, where

$$g(x, \sigma^*) = b(x, \sigma^*)^{-2(\sigma^*)^2} - C, \quad (7)$$

$$b(x, \sigma^*) = \sqrt{\frac{\sum_{i=1}^l e^{-\frac{\|x-x_i\|_2^2}{2(\sigma^*)^2}} (y_i + C)^{-\frac{1}{2(\sigma^*)^2}}}{\sum_{i=1}^l e^{-\frac{\|x-x_i\|_2^2}{2(\sigma^*)^2}} (y_i + C)^{\frac{1}{2(\sigma^*)^2}}}}; \quad C \in (-\min_{1 \leq i \leq l} (y_i), \infty); \quad (8)$$

From Theorem 2, we then can summarize our approach, the visual regression approach (VRA), for regression problem as follows.

VRA Algorithm

Step I (Preprocessing): With an appropriate C and ε , form the positive class D_{**}^+ and negative class D_{**}^- as (4) defined;

Step II (Classification): Let $\theta = (x, y)$ and define the distanced(θ^1, θ^2) as (5) defined; apply the VCA to obtain a classification surface $f(x, y, \sigma^*) = 0$ With the defined distance;

Step III (Regression): Define the expected regression function by $g(x, \sigma^*) = b(x, \sigma^*)^{-2(\sigma^*)^2} - C$ according to (7) - (8).

Table 1. Two well-known time-series prediction datasets

Problems	Input dim	Size of training set	Size of test set
Mackey Glass	6	1000	1000
Data set D	20	277	25

Table 2. The performance of VRA compared with SVR

Problems	Computation times(hours)		Prediction error	
	SVR	VMR	SVM	VRA
Mackey Glass	120	0.2	0.0003	0.051
Data set D	3.4	0.01	0.0026	0.0308
Mean	61.7	0.105	0.00145	0.0409

To support the effectiveness of VRA, we apply it to two well-known time-series prediction data sets (Table 1), the Mackey Glass and Data set D data sets (cf. <http://ida.first.fraunhofer.de/projects/bench/ts/>) are selected in application. The simulation results are shown in Table 2 compared with SVR (Gaussian kernel is used). The results reveal that VRA and SVR both can perform regression very well. But, as compared, VRA requires much less computation time, whereas the SVR has potential advantage enable to achieve a higher precision of solution. Why VRA has not yielded a comparable precision solution is maybe because of the specific application of the distance $d(\theta_1, \theta_2)$, which, on one hand, assures deduction of an explicit expression of the regression function, but, on the other hand, introduces some type of an isotropy causing the loss of precision of solution. Our further simulation shows that without pursuing the explicit expression (instead, the prediction value will be computed via solving the implicit classification surface $f(x, y, \sigma^*) = 0$). The precision can be much improved, particularly achieving a same or higher degree with the SVR.

5 Concluding Remarks

This paper proposes a new regression approach, visual regression approach, with an expectation of resolving the problems of the current regression approaches. The core idea is first to construct relationship between regression and classification problems, based on Ancona theorem, and to transfer the regression problem to classification problem; then use VCA, which is developed based on mimicking human sensation and perception principle, to solve the transformed classification problem to get an implicit regression function; and finally utilize some mathematical skills to obtain the explicit solution of the regression problem. The preliminary simulations have demonstrated that the new approach is highly efficient and very often brings a significant reduction of computation effort without loss of prediction capability, especially as compared with the prevalently adopted SVR approach.

There are still many problems remaining for further research. Some of them, for instance, are: (i) assessing theoretically the approach proposed (say, analyzing the properties like convergence and complexity); (ii) applying the more dedicated nonlinear scale space theory for further efficiency speed-up; (iii) looking for a more appropriate distance to improve the VRA's precision.

References

1. Lehmann, T., Gonner, C., Spitzer, K.: Survey: interpolation methods in medical image processing. *IEEE T. Med. Imaging*. **18** (1999) 1049–1075
2. Angleton, G., Bonham, D.: Least squares regression vs. geometric mean regression for ecotoxicology studies. *Appl. Math. Comput.* **72** (1995) 21–32
3. Leski, M.: ε -insensitive fuzzy c-regression models: introduction to ε -insensitive fuzzy modeling. *IEEE T. Syst. Man CY. B.* **34** (2004) 4–16
4. Lee, E., Chee, L., Yuen R., Lo S.: A hybrid neural network model for noisy data regression. *IEEE T. Syst. Man CY. B.* **34** (2004) 951–960
5. Scholkopf, B., Burges, C., Smola, A.: *Advances in Kernel Methods - Support Vector Learning*. MIT Press. (1999)
6. Ancona, N.: Classification Properties of Support Vector Machines for Regression. Technical Report RI-IESI/CNR - Nr. 02/99. (1999)
7. Xu, Z.B., Meng, D.Y., Jing, W.F.: A New Approach for Classification: Visual Simulation Point of View. ISBN 2005, LNCS3497. **Part II** (2005) 1-7
8. Coren, S., Ward, L., Enns, J.: *Sensation and Perception*. Harcourt Brace College Publishers. (1994)
9. Rasch, G.: Robust boosting via convex optimization. PhD thesis, University of Potsdam, Neues Palais 10, 14469 Potsdam, Germany. (2001)

Orthogonally Rotational Transformation for Naive Bayes Learning

Limin Wang¹, Chunhong Cao², Haijun Li^{1,3}, Haixia Chen¹, and Liyan Dong¹

¹ College of Computer Science and Technology, JiLin University,
ChangChun 130012, China
jeffreywlm@sina.com

² College of Information Science and Engineering, Northeastern University,
ShenYang 110004, China

³ College of Computer Science, YanTai University, YanTai 264005, China

Abstract. Naive Bayes is one of the most efficient and effective learning algorithms for machine learning, pattern recognition and data mining. But its conditional independence assumption is rarely true in real-world applications. We show that the independence assumption can be approximated by orthogonally rotational transformation of input space. During the transformation process, the continuous attributes are treated in different ways rather than simply applying discretization or assuming them to satisfy some standard probability distribution. Furthermore, the information from unlabeled instances can be naturally utilized to improve parameter estimation without considering the negative effect caused by missing class labels. The empirical results provide evidences to support our explanation.

1 Introduction

Naive Bayes [1][2] is one of the most efficient and effective learning algorithms for machine learning, pattern recognition and data mining. It is based on one assumption that predictive attributes X_1, \dots, X_n are conditionally independent given the class label C , which can be expressed as:

$$P(x_1, \dots, x_n | c) = \prod_{i=1}^n P(x_i | c)$$

Where lower-case letters denote specific values taken by corresponding attributes (for instance, x_i represents the event that $X_i = x_i$). $P(\cdot)$ denotes the probability. According to Bayes' theorem, we can get

$$P(c | x_1, \dots, x_n) = \frac{P(c) \prod_{i=1}^n P(x_i | c)}{P(x_1, \dots, x_n)} \propto P(c) \prod_{i=1}^n P(x_i | c)$$

When instance space T contains continuous attributes, [3] proposed General Naive Bayes (GNB) to handle this situation. Suppose the first m of n attributes are continuous, the classification rule is:

$$P(c|x_1, \dots, x_n) \propto P(c) \prod_{i=1}^m p(x_i|c) \prod_{j=m+1}^n P(x_j|c)$$

where $p(\cdot)$ refers to the probability density. Naive Bayes provides a simple and effective approach to classifier learning. Although it is obvious that the conditional independence assumption is rarely true in most real-world applications, experiments on benchmark databases have repeatedly shown it to be competitive with much more sophisticated induction algorithms including decision tree learning [4], rule learning [5]. To mitigate the negative effect of limited instances, researchers commonly discretize continuous attributes or assume them to satisfy some standard distribution (e.g., normal distribution). Hsu et al. [6] proposed a theoretical analysis of discretization’s effectiveness in Naive-Bayes learning. João Gama [7] presents Linear-Bayes for the continuous attributes, which assumes a multivariate normal distribution to compute the probabilities.

This paper proposed a new Naive-Bayes model, Self-adaptive Naive Bayes (SNB), which introduces independence characteristics into data based on orthogonally rotational transformation (ORT) and constructs discrete attributes on the basis of probability distribution. Furthermore, SNB can utilize the information from unlabeled instances without considering the negative effect caused by missing class labels.

The remainder of this paper is organized as follows. Sect. 2 describes the basic idea of General Naive Bayes. Sect. 3 and 4 describe how to introduce independence characteristics into instance space based on ORT and select appropriate attributes for discretization, respectively. Sect. 5 presents the corresponding experimental results of compared performance with regarding to Self-adaptive Naive Bayes and Linear Bayes. Sect. 6 wraps up the discussion.

2 General Naive Bayes (GNB)

Suppose instance space T contains two attributes: X_1 (continuous) and X_2 (discrete). And the values of X_1 have been discretized into a set of intervals, each corresponding to a discrete value. Then the independence assumption should be:

$$P(x_1 \leq X_1 \leq x_1 + \Delta, x_2|c) = P(x_1 \leq X_1 \leq x_1 + \Delta|c)P(x_2|c) \tag{1}$$

where $[x_1, x_1 + \Delta]$ is arbitrary interval of the values of attribute X_1 . This assumption, which is the basis of GNB, supports very efficient algorithms for both classification and learning. By the definition of a derivative,

$$\begin{aligned} P(c|x_1 \leq X_1 \leq x_1 + \Delta, x_2) &= \frac{P(c)P(x_1 \leq X_1 \leq x_1 + \Delta|c)P(x_2|c)}{P(x_1 \leq X_1 \leq x_1 + \Delta|x_2)P(x_2)} \\ &= \frac{P(c)p(\zeta|c)\Delta P(x_2|c)}{p(\eta|x_2)\Delta P(x_2)} \\ &= \frac{P(c)p(\zeta|c)P(x_2|c)}{p(\eta|x_2)P(x_2)} \end{aligned} \tag{2}$$

where $x_1 \leq \zeta, \eta \leq x_1 + \Delta$. When $\Delta \rightarrow 0$, $P(c|x_1 \leq X_1 \leq x_1 + \Delta, x_2) \rightarrow P(c|x_1, x_2)$ and $\zeta, \eta \rightarrow x_1$, hence

$$\lim_{\Delta \rightarrow 0} P(c|x_1 \leq X_1 \leq x_1 + \Delta, x_2) = P(c|x_1, x_2) = \frac{P(c)p(x_1|c)P(x_2|c)}{p(x_1|x_2)P(x_2)} \quad (3)$$

Suppose the first m of n attributes are continuous and the remaining attributes are discrete. Similar to the induction process of Eq.(3), there will be

$$\begin{aligned} P(c|x_1, \dots, x_n) &= \frac{P(c) \prod_{i=1}^m p(x_i|c) \prod_{j=m+1}^n P(x_j|c)}{p(x_1, \dots, x_m|x_{m+1}, \dots, x_n)P(x_{m+1}, \dots, x_n)} \\ &\propto P(c) \prod_{i=1}^m p(x_i|c) \prod_{j=m+1}^n P(x_j|c) \end{aligned} \quad (4)$$

The aim of Bayesian classification is to decide and choose the class that maximizes the posteriori probability, then the classification rule of GNB is:

$$c^* = \arg \max_{c \in C} P(c|x_1, \dots, x_n) = \arg \max_{c \in C} P(c) \prod_{i=1}^m p(x_i|c) \prod_{j=m+1}^n P(x_j|c) \quad (5)$$

3 Orthogonally Rotational Transformation (ORT)

Suppose $X = \{X_1, \dots, X_m\}$ and $X' = \{X_{m+1}, \dots, X_n\}$. Similar to the induction process of Eq.(3), there will be

$$P(c|x, x') = \frac{p(x|x', c)P(x', c)}{p(x|x')P(x')} \quad (6)$$

Let S_X denote the covariance matrix of X , $\{u_1, \dots, u_m\}$ are the eigenvectors of S_X . We construct new attribute set $Y^T = (Y_1, \dots, Y_m)^T = UX^T$ with orthogonal matrix $U = (u_1, \dots, u_m)^T$. Then for attribute set $\{Y_1, \dots, Y_m, X_{m+1}, \dots, X_n\}$, we will correspondingly have

$$P(c|y, x') = \frac{p(y|x', c)P(x', c)}{p(y|x')P(x')} \quad (7)$$

Let S_Y denote the covariance matrix of Y , then there will be

$$S_Y = US_XU^T = \text{diag}[\sigma_1, \dots, \sigma_m] \quad (8)$$

Since the off diagonal elements of S_Y i.e. the covariances are zero, Y_1, \dots, Y_m are statistically irrelevant. If they all satisfy normal distribution, they are independent of each other. On the other hand, the diagonal elements of S_Y are just the variances of the individual elements of Y , which is defined as follows:

$$\sigma_i = E[(y_i - E(y_i))^2]$$

Where $E[\cdot]$ denotes the expected value. According to the definition of Jacobian, there will be

$$|J| = |U| = 1 \quad (9)$$

The relationship between the joint probability density of Y and X is:

$$p(y) = \frac{p(x)}{|J|} = p(x)$$

Because no matter what values the other discrete attributes take, $Y^T = UX^T$ always holds. Accordingly,

$$\begin{cases} p(y|x', c) = \frac{p(x|x', c)}{|J|} = p(x|x', c) \\ p(y|x') = \frac{p(x|x')}{|J|} = p(x|x') \end{cases} \quad (10)$$

By comparing Eq.(6) with Eq.(7) and from Eq.(10), it is easy to see that

$$P(c|x, x') = P(c|y, x') \quad (11)$$

According to Eq.(5) and Eq.(11), there will be:

$$c^* = \arg \max_{c \in C} P(c|y, x') = \arg \max_{c \in C} P(c) \prod_{i=1}^m p(y_i|c) \prod_{j=m+1}^n P(x_j|c) \quad (12)$$

From the induction process of Eq. (11) we know that, since ORT is irrelevant to other discrete attributes (including class attribute C) we can naturally utilize the information learned from unlabeled class instances to construct covariance matrix S_X and then the information is transferred from U to Y by $Y^T = UX^T$. So the prediction of probability density of new attributes Y_i will be much more precise.

4 Selection of Continuous Attributes for Discretization

Principal Component Analysis (PCA) is directly based on a whole vector pattern and acquires a set of projections that can realize the best reconstruction for original data. Variance is a simple but valuable measure of how far a randomly drawn value is likely to depart from the mean. Those predictive attributes which have low variance are prone to be affected by noise and should be discarded. If the eigenvalues $\{\sigma_1, \dots, \sigma_m\}$ in Eq.(8) are ranked in descending order, the first l attributes can retain original information and circumvent the negative effect of noise. l is the smallest value that is determined by the criterion:

$$\sum_{i=1}^l \sigma_i / \sum_{j=1}^m \sigma_j \geq \theta \quad (13)$$

where θ is a user-predetermined parameter. In our experiments, we initialize it to be 95%. When attribute values are sparsely distributed it is unreasonable to

assume them to satisfy any probability distribution (e.g. normal distribution). So we only discretize the first l attributes and leave the other attributes with low variance intact, then the final classification rule is:

$$c^* = \arg \max_{c \in C} P(c) \prod_{i=1}^l P(\hat{y}_i|c) \prod_{j=l+1}^m p(y_j|c) \prod_{k=m+1}^n P(x_k|c) \quad (14)$$

where \hat{y}_i is the discretized version of y_i ($1 \leq i \leq l$). To ensure the classification accuracy and reduce the computational complexity, we discretize attribute y_i on the basis of MDL principle [8].

5 Experiments

We chose 15 data sets from the UCI machine learning repository¹ for our experiments. In data sets with missing values, we considered the most frequent attribute value as a candidate.

Table 1. Average classification accuracy and standard deviation

Data set	LB	SNB
Anneal	97.2625 ± 1.7262	98.3622 ± 1.2254√
Australian	98.0734 ± 0.7721	99.3782 ± 0.2240√
Breast	86.9534 ± 0.9325	88.8221 ± 0.7721√
Cleve	82.7432 ± 3.8273	85.7337 ± 2.8215√
Crx	87.0343 ± 1.7227	90.7283 ± 1.0571√
Diabetes	74.1421 ± 0.8272	78.8772 ± 0.6154√
German	73.1451 ± 1.8281	71.1476 ± 0.9725
Glass	71.8251 ± 1.9342	73.8715 ± 2.7251√
Heart-c	82.0242 ± 3.1622	85.7195 ± 2.7612√
Hepatitis	84.1415 ± 2.7226	82.1851 ± 3.8254
Horse-colic	81.1351 ± 1.4727	86.8511 ± 2.5219√
Hypothyroid	97.9141 ± 2.5832	96.8614 ± 1.7511
Iris	94.0014 ± 0.089	97.6295 ± 1.5891√
Sick-enthyroid	95.3515 ± 0.2732	96.5411 ± 0.5976√
Vehicle	60.1515 ± 1.3721	63.6071 ± 2.6144√

Our experiments compared SNB with another state-of-the-art Naive-Bayes method, Linear Bayes (LB), for classification. The classification performance was evaluated by ten-folds cross-validation for all the experiments on each data set. Table 1 shows classification accuracy and standard deviation for SNB and LB, respectively. '√' indicates that the accuracy of SNB is higher than that of LB at a significance level better than 0.05 using a two-tailed pairwise t-test on the results of the 20 trials in a data set. From Table 1, the significant advantage of SNB over LB in terms of higher accuracy can be clearly seen.

¹ <ftp://ftp.ics.uci.edu/pub/machine-learning-databases>

6 Conclusion

Naive Bayes is a well known and studied algorithm both in statistics and machine learning. Although its limitations with respect to conditional independence assumption, this procedure has a surprisingly good performance in a wide variety of domains, including many where there are clear dependencies between attributes. This paper presents Self-adaptive Naive Bayes that uses, for the continuous attributes, ORT to introduce independence characteristics and applies discretization when the distribution of continuous attributes can not be safely estimated. With respect to the final model, SNB can utilize the information learned from unlabeled instances very naturally.

References

1. Harry, Z., Charles, X. L.: A Fundamental Issue of Naive Bayes. *Lecture Notes in Computer Science*, Vol. 2671. Springer-Verlag, Berlin Heidelberg New York (2003) 591–602
2. Stijn, V., Richard, D., Guido, D.: Boosting Naive Bayes for Claim Fraud Diagnosis. *Lecture Notes in Computer Science*, Vol. 2454. Springer-Verlag, Berlin Heidelberg New York (2002) 202–211
3. LiMin, W., SenMiao, Y., Ling, L., HaiJun, L.: Improving the Performance of Decision Tree: A Hybrid Approach. *Lecture Notes in Computer Science*, Vol. 3288. Springer-Verlag, Berlin Heidelberg New York (2004) 1056–1064
4. Marco, B., Jordi, V.: Improving Naive Bayes Using Class-Conditional ICA. *Lecture Notes In Computer Science*, Vol. 2527. Springer-Verlag, Berlin Heidelberg New York (2002) 1–10
5. Marco, R., Paola, S.: Robust Bayes classifiers. *Artificial Intelligence* 125 (2001) 209–226
6. ChunNan, H., HungJu, H., TzuTsun, W.: Implications of the Dirichlet Assumption for Discretization of Continuous Variables in Naive Bayesian Classifiers. *Machine Learning* 53 (2003) 235–263
7. João, G.: A Linear-Bayes Classifier. *Lecture Notes in Computer Science*, Vol. 1952. Springer-Verlag, Berlin Heidelberg New York (2000) 49–58
8. Limin, W., Senmiao Y.: Induction of hybrid decision tree based on post-discretization strategy. *Progress in Natural Science* 14 (2004) 541-545

Efficient Learning Bayesian Networks Using PSO

Tao Du, S.S. Zhang, and Zongjiang Wang

Shanghai Jiaotong University, Shanghai 200030, China
dutao@sjtu.edu.cn

Abstract. In this paper, we firstly introduce particle swarm optimization to the problem of learning Bayesian networks and propose a novel structure learning algorithm using PSO. To search in DAG spaces efficiently, a discrete PSO algorithm especially for structure learning is proposed based on the characteristics of Bayesian networks. The results of experiments show that our PSO based algorithm is fast for convergence and could obtain better structures compared with GA based algorithms.

1 Introduction

Bayesian networks are a graphical representation of a multivariate joint probability distribution that exploits the dependency structure of distributions [2]. Bayesian networks provide a proper representation for encoding uncertain expert knowledge in expert systems. But in real circumstances, the experts of domains can not define the dependencies for all variables explicitly and specifically. Therefore it is very valuable to learn Bayesian networks from data.

The problem of learning Bayesian networks from data is NP-hard. Therefore several heuristic searching techniques, such as greedy hill-climbing, simulated annealing and GA have been used. Among those heuristics, GA and evolution computation [4, 6, 7] have been intensively researched and proved being effective in learning Bayesian networks. However, there exist two drawbacks of GA in learning Bayesian networks which are its expensive computational cost and premature convergence. When the number of variables in Bayesian networks is large, those drawbacks would degrade performance of the learning algorithm and make it tend to return a network structure which is local optimal.

In this paper, a novel structure learning algorithm based on Particle Swarm Optimization (PSO) is proposed. PSO converges rapidly and has to be found being robust in solving problems featuring nonlinearity and non-differentiability, multiple optima, and high dimensionality. However, existed PSO algorithms could not be directly used for the problem of learning Bayesian networks. To search in DAG spaces efficiently, in this paper, a novel discrete PSO algorithm especially for structure learning is proposed according to characteristics of Bayesian networks.

2 Structure Learning

In this paper, a Bayesian network is denoted as $B = (G, \theta)$ that $G = (X, E)$ is a directed acyclic graph (DAG) where the set of nodes represents a set of

random variables $X = \{X_1, \dots, X_n\}$ and that $\theta_i = [P(X_i|\Pi_{X_i})]$ is the conditional probability of node i given the state of its parents Π_{X_i} .

The Bayesian network represents a probability distribution over X which admits the following joint distribution decomposition:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i|\Pi_{X_i}) \quad (1)$$

Given a data set D , the objective of structure learning is to identify the best network structure G that best matches D . For search-and-scoring approaches, how goodness the structure matches data set is measured by adopted scoring metric.

Because of the decomposition characteristic of Bayesian network shown in equation (1), commonly used scoring metric such as Bayesian score, bic score and MDL score could be decomposed into summation of sub-scores of each variable given the states of their parents, which is shown in the equation (2).

$$Score(G) = \sum_{X_i \in X} Score(X_i|\Pi_{X_i}) \quad (2)$$

If a score metric follows (2), we say that it is “decomposable”. For any decomposable score metric, it is easy to see that, the best network structure G_{best} could be obtained by searching the best combination of parents Π_{X_i} for every variable X_i in X .

Therefore the problem of structure learning could be converted to an optimization problem. The object function is illustrated in equation (3).

$$\max(Score(G)) = \sum_{X_i \in X} (\max(Score(X_i|\Pi_{X_i}))) \quad (3)$$

3 Structure Learning Algorithm Using PSO

PSO is a population based optimization technique, which developed out of work of simulating of the movement of flock of birds [5]. PSO is initially developed for solving problems in real-number spaces. But the idea of PSO is also applicable for problems in discrete spaces. In recent years, there have been many contributions reported for applying PSO to discrete problems [1, 8]. As we know, in this paper, PSO is firstly introduced for the problem of learning Bayesian networks from data.

The search space of Bayesian networks is a DAG space, i.e. every “point” in the space is a directed acyclic graph (DAG). To use PSO in DAG spaces, the mathematical objects and operations listed below should be defined according to the characteristics of DAG space.

1. The position of a particle.
2. Velocity of a particle.
3. The operation: $move(position, velocity) \rightarrow position$.
4. The operation: $subtraction(position, position) \rightarrow velocity$.
5. The operation: $add(velocity, velocity, velocity) \rightarrow velocity$.

Since the search space is a DAG space, it is natural to define “the position of a particle” as a DAG. In implement, the position of a particle is represented as an adjacent matrix. Suppose at step time t , $G_{i,t}$ denotes the position of particle i , $G_{i,t}^p$ denotes the best previous position of particle i , and G_t^g denotes the best previous position among the particle swarm.

In PSO, the velocity is used to move the particle and measure the difference between two positions. For consistence with the definition of “position”, the velocity of a particle is also defined as an $n \times n$ matrix. Here, the velocity matrix for particle i at step t is denoted by $V_{i,t}$. And the element of velocity matrix which is in row i and column j is denoted by $v_{i,j}$, where, $v_{i,j} \in \{-1, 0, 1\}$. When the velocity is applied to one position, $v_{i,j} = -1$ means that the edge from node i to node j in the DAG would be removed, $v_{i,j} = 0$ means that the corresponding edge would remain unchanged, and $v_{i,j} = 1$ means that the edge from node i to node j would be added.

When a position G is moved to G' by a velocity V , the element $g_{i,j}$ is changed according to $v_{i,j}$. If $g_{i,j} = 0$, when $v_{i,j} = 1$, then $g'_{i,j} = 1$, which means the edge from node i to node j would be added; otherwise $g'_{i,j} = g_{i,j} = 0$. If $g_{i,j} = 1$, $v_{i,j} = -1$ would make edge from node i to node j to be removed, and $v_{i,j} = 0$ or 1 would make $g'_{i,j} = g_{i,j} = 1$.

When the operation $subtraction(G, G') = V$ is executed, the value of $v_{i,j}$ is determined by $g_{i,j}$ and $g'_{i,j}$. If $g_{i,j} = g'_{i,j}$, then $v_{i,j} = 0$, If $g_{i,j} = 1$ and $g'_{i,j} = 0$, then $v_{i,j} = 1$, if $g_{i,j} = 0$ and $g'_{i,j} = 1$, then $v_{i,j} = -1$.

The operation $add(V_a, V_b, V_c) \rightarrow V_{new}$ is used to combine the three tendencies for moving a particle. Here V_a is original velocity of a particle. V_b is velocity resulted from $subtraction(G_{i,t}^p, G_{i,t})$ by which the particle would move toward its best previous position. V_c is resulted from $subtraction(G_t^g, G_{i,t})$ by which the particle would move to global best previous position among the particle swarm.

Suppose $V_a = [v_{a,1}, \dots, v_{a,n}]$, $V_b = [v_{b,1}, \dots, v_{b,n}]$, $V_c = [v_{c,1}, \dots, v_{c,n}]$, and $V_{new} = [v_{new,1}, \dots, v_{new,n}]$, the operation $add(V_a, V_b, V_c) \rightarrow V_{new}$ is defined by equation (4).

$$v_{new,i} = \begin{cases} v_{a,i} & \text{with probability } Pr_a \\ v_{b,i} & \text{with probability } Pr_b \\ v_{c,i} & \text{with probability } Pr_c \end{cases} \quad (4)$$

subject to $Pr_a + Pr_b + Pr_c = 1$

In the initial phase, Pr_a could be set a larger value to make particles moving to search new area. And during the searching, Pr_b and Pr_c should be increased such that the neighbor areas of best previous positions could be searched more intensively.

Our algorithm which is named after "BN-PSO" is shown in Fig.1. In BN-PSO, a swarm of particles is firstly initialized. During searching, each particle is moving in the search space. In our algorithm, the values of elements in V_a are randomly chosen in $\{-1, 0, 1\}$. And then this velocity is modified by V_b and V_c to obtain a new velocity which may make the particle moving toward its best previous position and the global best previous position.

Algorithm BN-PSO**Inputs:**

- Training data $D = \{x^1, \dots, x^N\}$, N is the number of data case
- Number of particles N_{swarm}

Outputs:

- the DAG G that approximates maximizing the $Score(G)$

Begin

For each particle

 Initialize position $G_{i,0}^p; G_{i,0}^p = G_0^g = G_{i,0}$;

End for

Loop until the stop criterion is satisfied.

For each particle i

 Generate V_a at random;

$V_b = subtraction(G_{i,t}^p, G_{i,t}); V_c = subtraction(G_t^g - G_{i,t});$

$V_{new} = add(V_a, V_b, V_c);$

If check_constraint(V_{new}), **then** $G_{i,t+1} = move(G_{i,t} + V_{new});$

Else, continue;

End If

If $Score(G_{i,t+1}) > Score(G_{i,t}^p)$, **then** $G_{i,t+1}^p = G_{i,t+1}$; **End If**

If $Score(G_{i,t+1}) > Score(G_t^g)$, **then** $G_{i,t+1}^g = G_{i,t+1}$; **End If**

End for

End loop

Return G_t^g

End

Fig. 1. Structure Learning Algorithm using PSO: BN-PSO

After V_{new} is generated, the operation check_constraint(V_{new}) should be executed to check the validation of V_{new} to make sure that no cycles exist in the resulted graph. In implementation, this constraint could be accomplished by maintaining a path number matrix (PNM) [4].

If V_{new} is valid, the particle would move from $G_{i,t}$ to $G_{i,t+1}$. And then the score of position $G_{i,t+1}$ would be evaluated. If the score of $G_{i,t+1}$ is better than that of $G_{i,t}^p$ or G_t^g , $G_{i,t}^p$ or G_t^g would be updated to $G_{i,t+1}$.

The searching process would be continued until one of stop criterions is satisfied. For BN-PSO, the searching process would be terminated when there has been no improvement of the best position in the whole swarm or when the numbers of iterations of particles all exceed a predefined number.

When the searching process is terminated, the DAG of the global best position G_t^g among all particles is returned as the result of our algorithm.

4 Experiment Evaluation

The purpose of this experiment is to evaluate the performance of our structure learning algorithm BN-PSO.

In this experiment, three well-known benchmarks of Bayesian networks including ASIA, ALARM and INSURANCE are used. For every Bayesian networks, the data set with 1000 instances is generated using probabilistic logic sampling.

To evaluate the performance of our algorithms, two metrics are used in this paper. The first is the score of the obtained DAG which indicates the “accuracy” of the algorithms. Here, Bayesian score [3] is adopted as the score metric. The second metric adopted in test is the number of the positions (DAGs) that have been evaluated during the search process.

We compare our algorithm with structure learning algorithm using GA presented in [6] on the data sets shown in table 1. The algorithm proposed in [6] is named “BN-GA” in this paper. The results of experiment are shown in table 1. All values shown in table 1 are averages over 10 times running BN-PSO and the BN-GA.

Table 1. Performance of algorithms in experiment

Data sets	BN-PSO		BN-GA	
	N_e	Score	N_e	Score
D1	562	-2281	642	-2281
D2	12438	-15532	19439	-15894
D3	23469	-11794	65420	-11806

From table 1 we could see that, for all data sets, the average evaluated number of DAGs N_e of BN-PSO is much smaller than that of BN-GA. And compared with BN-GA, more larger the number of variables is, more quicker our algorithm is. The reason for our algorithm’s superiority on “efficiency” over BN-GA is that, PSO could converge much more rapidly than GA. Therefore PSO is very promising for learning Bayesian networks when the number of variables is large.

From table 1 it is shown that, for most data sets, the average scores of BN-PSO is better than that of BN-GA, which indicate that our algorithm can not only reduce the time of learning Bayesian networks but also improve score of finally learned network structure.

5 Conclusion

In this paper, we firstly introduce PSO to the problem of learning Bayesian networks from data and propose an efficient structure learning algorithm using PSO.

The search space in learning Bayesian networks is a DAG space, and there are no PSO algorithms on such space have been found published. To solve this problem, we give the new definitions on mathematical objects and operations in PSO according to the characteristics of DAG space and problem of structure

learning. And based on those definitions, an efficient structure learning algorithm using PSO is proposed.

The results of experiments have shown that, compared to structure learning algorithm using GA, the algorithm BN-PSO proposed in this paper could reduce the time for learning Bayesian network greatly. In addition, the score of the finally obtained network structure could be also improved.

Acknowledgments

This work is supported in part by the National Natural Science Foundation under Grant No. 60374071, and Zhejiang Commission of Science and Technology under Grant No. 2003C11009.

References

1. Clerc, M.: Discrete Particle Swarm Optimization, illustrated by the Traveling Salesman Problem. in *New Optimization Techniques in Engineering*. Springer-Verlag, Heidelberg, Germany (2004) 219–239
2. Heckerman, D.: A tutorial on learning with Bayesian networks. *Learning in Graphical Models*. Kluwer, Dordrecht Netherlands (1998)
3. Cooper, G., Herskovits, E.: A Bayesian method for the induction of probabilistic networks from data. *Mach. Lear.* 9 (1992) 309–347
4. Wong, M.L., Leung, K.S.: An Efficient Data Mining method for learning Bayesian networks using an evolutionary algorithm-based hybrid approach. *IEEE trans. On Evolutionary computation* 8 (2004) 378–404
5. Kennedy, P., Eberhart, R.: Particle Swarm Optimization. In *Proceedings of IEEE international Conference of Neural Networks (ICNN95)*. Piscatawa, NJ. 4 (1995) 1942–1948
6. Larraaga, P., Poza, M., Yurramendi, Y., Murga, R., Kuijpers, C.: Structural learning of Bayesian network by genetic algorithms: performance analysis of control parameters. *IEEE Trans. Pattern Anal. Machine Intell.* 18 (1996) 912–926
7. LI, X.L., Yuan, S.M., He, X.D.: Learning Bayesian Networks Structures Based on Extending Evolutionary Programming. *Proceeding of the Third International Conference on Machine Learning and Cybernetics*. Shanghai (2004) 1594–1598
8. Gaing, Z.L.: Discrete particle swarm optimization algorithm for unit commitment. *Power Engineering Society General Meeting, Vol. 3. IEEE.* (2003) 418–424

Improving K-Modes Algorithm Considering Frequencies of Attribute Values in Mode

Zengyou He, Shengchun Deng, and Xiaofei Xu

Department of Computer Science and Engineering, Harbin Institute of Technology, China
zengyouhe@yahoo.com, {dsc, xiaofei}@hit.edu.cn

Abstract. In this paper, we present an experimental study on applying a new dissimilarity measure to the k -modes clustering algorithm to improve its clustering accuracy. The measure is based on the idea that the similarity between a data object and cluster mode, is directly proportional to the sum of relative frequencies of the common values in mode. Experimental results on real life datasets show that, the modified algorithm is superior to the original k -modes algorithm with respect to clustering accuracy.

1 Introduction

Clustering is an important data mining technique that groups together similar data records. Recently, more attention has been put on clustering categorical data [e.g., 1-4], where records are made up of non-numerical attributes.

The k -modes algorithm [1] extends the k -means paradigm to cluster categorical data by using (1) a simple matching dissimilarity measure for categorical objects, (2) modes instead of means for clusters, and (3) a frequency-based method to update modes in the k -means fashion to minimize the cost function of clustering. Because the k -modes algorithm uses the same clustering process as k -means, it preserves the efficiency of the k -means algorithm, which is highly desirable for data mining.

However, the dissimilarity measure used in k -modes doesn't consider the relative frequencies of attribute values in each cluster mode, this will result in a weaker intra-cluster similarity by allocating less similar objects to the cluster. Let's consider the following example shown in Fig.1.

Example 1: In this artificial example, the dataset is described with three categorical attributes $A1, A2, A3$ and there are two clusters with their modes. Assuming that we have to allocate a data object $Y = [a, p, w]$ to either cluster 1 or cluster 2. Since these two clusters have the same modes, according to the k -modes algorithm, we can assign Y to either cluster 1 or cluster 2. However, from the viewpoint of intra-cluster similarity, it is more desirable to allocate Y to cluster 1.

The above example shows that the similarity measure used in k -modes algorithm dose not always represent the real semantic distance between a data object and a cluster, and the notion of cluster mode is not adequate to represent the characteristic of a categorical cluster.

The main objective of this paper is to apply a new dissimilarity measure to the k -modes clustering algorithm to improve its clustering accuracy without sacrificing its

Cluster 1: mode1=[a, p, r] Cluster 2: mode2=[a, p, r]

A1	A2	A3
a	p	r
a	p	s
a	p	t

A1	A2	A3
a	q	r
b	p	t
a	p	k

Fig. 1. Two clusters with their modes in example 1

scalability. The main idea is to re-define the cluster’s representation as a mode with cluster-based relative frequencies of attribute values in the mode. The similarity between the data object and a mode is hence measured as sum of relative frequencies of their common values.

Continuing example 1, in our method, the modes of the 2 clusters will be represented as $mode1 = [a: 1, p: 1, r: 1/3]$ and $mode2 = [a: 2/3, p: 2/3, r: 1/3]$, respectively. As a result, the distance between Y and $mode1$ is computed as: $d(Y, mode1) = 3 - (1+1+0)=1$, and $d(Y, mode2) = 3 - (2/3+2/3+0)=5/3$. Therefore, $d(Y, mode1) < d(Y, mode2)$. That is to say, if this new dissimilarity measure is applied, the data object Y will be assigned to the proper cluster, i.e., cluster 1.

2 The Modified k -Modes Algorithm

The k -modes algorithm is modified from the k -means algorithm by using a simple matching dissimilarity measure for categorical data, and replacing the means of cluster with the modes. These modifications remove the numeric-only limitation of the k -means algorithm while maintain its efficiency in clustering categorical data sets.

Let X, Y be two categorical objects described by m categorical attributes. The simple dissimilarity measure between X and Y is defined by the total mismatches of the corresponding attribute values of the two objects. The smaller the number of mismatches is, the more similar the two objects. Formally,

$$d_1(X, Y) = \sum_{j=1}^m \delta(x_j, y_j) \tag{1}$$

where

$$\delta(x_j, y_j) = \begin{cases} 0 & (x_j = y_j) \\ 1 & (x_j \neq y_j) \end{cases} \tag{2}$$

Let X be a set of categorical objects described by m categorical attributes A_1, \dots, A_m . A mode of $X = \{X_1, X_2, \dots, X_n\}$ is a vector $Q = [q_1, q_2, \dots, q_m]$ that minimizes

$$D(X, Q) = \sum_{i=1}^n d_1(X_i, Q) \tag{3}$$

Here, Q is not necessarily an object of X .

Let $n_{c_{k,j}}$ be the number of objects having the k th category $c_{k,j}$ in attribute A_j and $f_r(A_j = c_{k,j} | X) = \frac{n_{c_{k,j}}}{n}$ be the relative frequency of category $n_{c_{k,j}}$ in X . The function $D(X, Q)$ is minimized iff $f_r(A_j = q_j | X) \geq f_r(A_j = c_{k,j} | X)$ for $q_j \neq c_{k,j}$ and all $j = 1, \dots, m$.

The above expression defines a way to find Q from a given X , and therefore is important because it allows the k -means paradigm to be used to cluster categorical datasets.

When the above is used as the dissimilarity measure for categorical objects, the optimization problem for partitioning a set of n objects described by m categorical attributes into k clusters becomes

$$\text{Minimize } P(W, Q) = \sum_{l=1}^k \sum_{i=1}^n \sum_j^m w_{i,l} \delta(x_{i,j}, q_{l,j}). \tag{4}$$

$$\begin{aligned} \text{Subject to } & \sum_{l=1}^k w_{i,l} = 1, \quad 1 \leq i \leq n. \\ & w_{i,l} \in \{0,1\}, \quad 1 \leq i \leq n, \quad 1 \leq l \leq k \end{aligned} \tag{5}$$

where W is an $n \times k$ partition matrix, $Q = \{Q_1, Q_2, \dots, Q_k\}$.

To minimize the cost function, the k -modes algorithm uses the procedure described in Fig.2.

-
1. Select k initial modes, one for each cluster.
 2. Allocate an object to the cluster whose mode is the nearest to it according to (1). Update the mode of the cluster after each allocation.
 3. After all objects have been allocated to clusters, retest the dissimilarity of objects against the current modes. If an object is found such that its nearest mode belongs to another cluster rather than its current one, reallocate the object to that cluster and update the modes of both clusters.
 4. Repeat 3 until no object has changed clusters after a full cycle test of the whole data set.
-

Fig. 2. The original k -modes algorithm

In this paper, we apply a new dissimilarity measure to the k -modes clustering algorithm to improve its clustering accuracy. The main difference is that similarity between the data object and a mode is measured as sum of cluster-based relative frequencies of their common values. Formally, the new dissimilarity measure is

$$d_2(X_i, Q_l) = \sum_{j=1}^m \phi(x_{i,j}, q_{l,j}). \quad (6)$$

where

$$\phi(x_{i,j}, q_{l,j}) = \begin{cases} 1 - f_r(A_j = q_{l,j} | X_i) & (x_{i,j} = q_{l,j}) \\ 1 & (x_{i,j} \neq q_{l,j}) \end{cases}. \quad (7)$$

Note that $f_r(A_j = q_{l,j} | X_i)$ is frequency of $q_{l,j}$ in cluster X_i .

Hence, the cost function (4) becomes

$$P(W, Q) = \sum_{l=1}^k \sum_{i=1}^n \sum_{j=1}^m w_{i,l} \phi(x_{i,j}, q_{l,j}). \quad (8)$$

Therefore, the modified k -modes algorithm can use the same procedure shown in Fig.2 except that the distance function (1) is replaced by (6). Obviously, the computational cost of the modified algorithm is still $O(Tkn)$, where T is the number of iterations and n the number of objects in the input data set. That is, our modifications do not destroy the scalability of the original k -modes algorithm, and we will show in the next section that the modified k -modes algorithm greatly improve the accuracy of clustering results.

3 Experimental Results

We experimented with two real-life datasets: the Congressional Voting dataset and the Mushroom dataset, which were obtained from the UCI Machine Learning Repository [5].

The clustering accuracy for measuring the clustering results is computed as follows. Suppose the final number of clusters is k , clustering accuracy r is defined as: $r = (\sum_{i=1}^k a_i) / n$, where n is number of objects in the dataset, a_i is number of objects occurring in both cluster i and its corresponding class, which has the maximal value. In other words, a_i is the number of objects with class label that dominate cluster i . Consequently, the clustering error is defined as $e = 1 - r$.

We compare modified k -modes algorithm with the original k -modes algorithm [1]. These two algorithms use the same initial points selection method, that is, selects the first k distinct records from the data set to construct initial k modes.

We used the modified k -modes and the original k -modes algorithms to cluster the *congressional voting* dataset and the *mushroom* dataset into different numbers of clusters. For each fixed number of clusters, the clustering errors of both algorithms were compared.

Fig.3 contrasts the clustering accuracies on the *congressional voting* dataset. From Fig.3, we can summarize the relative performance of two algorithms as Table 1. In Table 1, the numbers in column labelled by k ($k=1, 2$) are the times that an algorithm has rank k between the two algorithms. For instance, in the 8 experiments, the original k -modes algorithm performed second best in 2 cases, that is, it is ranked 2 for 2 times.

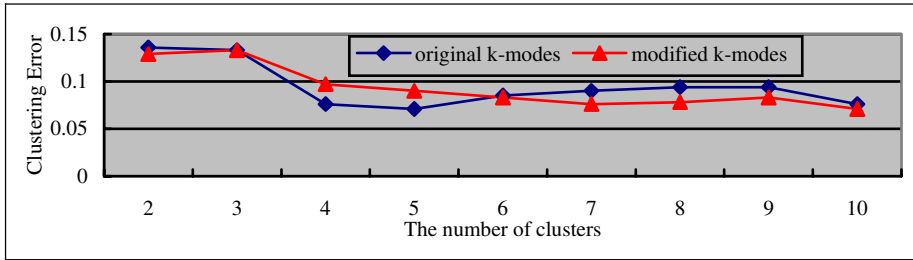


Fig. 3. Clustering error vs. Different number of clusters (*congressional voting* dataset)

Table 1. Relative performance of different clustering algorithms (*congressional voting* dataset)

Ranking	1	2	Average Clustering Error
Original <i>k</i> -modes	2	6	0.095
Modified <i>k</i> -modes	6	2	0.090

From Figure 3 and Table 1, although the average clustering accuracy of our algorithm is only a little better than that of the original *k*-modes algorithm, while the cases of our algorithm that beat the original *k*-modes algorithm is dominant in this experiment. Therefore, the test on the *congressional voting* dataset verified the improvement of our method on clustering accuracy.

The experimental results on the *mushroom* dataset are described in Figure 4 and the summarization on the relative performance of the 2 algorithms is given in Table 2. From these results, it can be seen that our algorithm almost performed the best in all cases. Even the cases that it performed the worst, the clustering errors are almost the same with that of the original *k*-modes algorithm. Furthermore, the average clustering error of our algorithm is significantly smaller than that of the original *k*-modes algorithm.

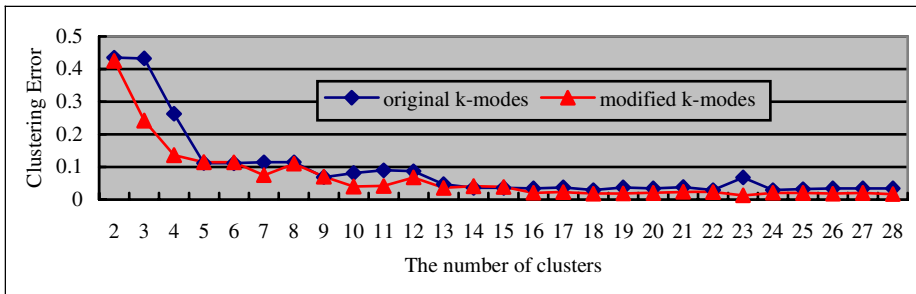


Fig. 4. Clustering error vs. Different number of clusters (*mushroom* dataset)

A cluster is called a pure cluster if all the data objects in it belong to a single class. On the *mushroom* dataset, mushrooms in a “pure” cluster will be either all edible or all poisonous. To find out the ability of the two algorithms on detecting pure clusters, we ran a series of experiments with increasing number of clusters (from 2 to 28). Compared to the original *k*-modes algorithm, our algorithm performed the best in all cases except one. Furthermore, when the number of clusters is increased, the outputs of our algorithm are significantly better than that of the original *k*-modes algorithm.

Table 2. Relative performance of different clustering algorithms (*mushroom* dataset)

Ranking	1	2	Average Clustering Error
Original <i>k</i>-modes	4	23	0.092
Modified <i>k</i>-modes	23	4	0.067

4 Conclusions

This paper presents an improved *k*-modes algorithm by applying a new dissimilarity measure. It is found that our modified *k*-modes algorithm is superior to original *k*-modes algorithm with respect to clustering accuracy.

For future work, we will take the proposed algorithm as background-clustering algorithm for detecting cluster based local outliers [6] and class outliers [7].

Acknowledgements

This work was supported by High Technology Research and Development Program of China (No.2003AA4Z2170, No. 2003AA413021) and IBM SUR research fund.

References

1. Huang, Z.: Extensions To The K-means Algorithm for Clustering Large Data Sets with Categorical Values. *Data Mining and Knowledge Discovery* **2** (1998) 283-304
2. Huang, Z., Ng, M. K.: A Fuzzy K-modes Algorithm for Clustering Categorical Data. *IEEE Transactions on Fuzzy Systems* **7(4)** (1999) 446-452
3. He, Z., Xu, X., Deng, S.: Squeezer: An Efficient Algorithm for Clustering Categorical Data. *Journal of Computer Science & Technology* **17(5)** (2002) 611-624
4. He, Z., Xu, X., Deng, S.: A Cluster Ensemble Method for Clustering Categorical Data. *Information Fusion* **6(2)** (2005) 143-151
5. Merz, C. J., Merphy. P.: UCI Repository of Machine Learning Databases. [<http://www.ics.uci.edu/~mlern/MLRRepository.html>] (1996)
6. He, Z., Xu, X., Deng, S.: Discovering Cluster-based Local Outliers. *Pattern Recognition Letters* **24** (2003) 1641-1650
7. He, Z., Xu, X., Huang, J.Z., Deng, S.: Mining Class Outliers: Concepts, Algorithms and Applications in CRM. *Expert Systems with Applications* **27(4)** (2004) 681-697

Distance Protection of Compensated Transmission Line Using Computational Intelligence

S.R. Samantaray¹, P.K. Dash², G. Panda¹, and B.K. Panigrahi³

¹Electronics and Communication Engg,
National Institute of Technology, Rourkela-769 008, India
gpanda@nitrkl.ac.in
Sbh_samant@rediffmail.com

²Electrical and Electronics Engg., College of Engineering Bhubaneswar-751024, India
Pkdash_India@yahoo.com

³Department of Electrical Engg., Indian Institute of Technology, Delhi, India
bkpanigrahi@ee.iitd.ac.in

Abstract. A new approach for protection of transmission line including TCSC is presented in this paper. The proposed method includes application of Fuzzy Neural Network for distance relaying of a transmission line operating with a thyristor controlled series capacitor (TCSC) protected by MOVs. Here the fuzzy neural network (FNN) is used for calculating fault location on the TCSC line. The FNN structure is seen as a neural network for training and the fuzzy viewpoint is utilized to gain insight into the system and to simplify the model. The number of rules is determined by the data itself and therefore, a smaller number of rules are produced. The network parameters are updated by Extended Kalman Filter (EKF) algorithm. with a pruning strategy to eliminate the redundant rules and fuzzification neurons resulting in a compact network structure. The input to the FNN are fundamental currents and voltages at the relay end, sequence components of current, system frequency and the firing angle with different operating conditions and the corresponding output is the location of the fault from the relaying point. The location tasks of the relay are accomplished using different FNNs for different types of fault (L-G, LL-G, LL, LLL).

1 Introduction

The series inductive reactance of ac transmission lines is one of the factors which governs the maximum amount of power that can stably be transferred by these lines under steady-state conditions. One method of increasing the steady-state maximum power transfer capability of an ac line is to reduce its net series inductive reactance, and in practice this has traditionally been achieved by connecting a fixed capacitive compensating reactance in series with the line using static capacitor banks. However, it has long been recognized that if the capacitive reactance provided by such a series compensator can be dynamically controlled it is possible not only to increase the steady-state power transfer capability of a transmission system, but also to improve dramatically the ability of the system to retain stability during the transient conditions

which follow system disturbances [1,2]. Subsequent studies have shown that the improvement in transient stability that results from the use of a dynamically-controllable series compensator allows for a significant potential improvement in the utilization of high-power transmission lines [3].

The TCSC [7] is one of the main FACTS devices, which has the ability to improve the utilization of the existing transmission system. However, the implementation of this technology changes the apparent line impedance, which is controlled by the firing angle of thyristors, and is accentuated by other factors including the metal oxide varistor (MOV). The presence of the TCSC in fault loop not only affects the steady state components but also the transient components. The controllable reactance, the MOVs protecting the capacitors and the air-gaps operation make the protection decision more complex and therefore conventional relaying scheme based on fixed settings has its limitation. Kalman filtering and artificial neural network techniques are applied for adaptive protection of transmission line possessing a TCSC.

Neural Network has the shortcoming of implicit knowledge representation, whereas, FLS is subjective and heuristic. The major limitations of FLS are the lack of a general systematic procedure for rule learning and tuning, and determining the best shape of membership functions. As NN and FLS have different advantages and drawbacks, it is quite reasonable to consider the possibility of integrating the two paradigms into the same system in order to benefit from both of them. One such approach is integrating the learning capabilities of neural network to the robustness of fuzzy logic systems in the sense that fuzzy logic concepts are embedded in the network structure. It also provides a natural framework for combining both numerical information in the form of input/output pairs and linguistic information in the form of IF-THEN rules in a uniform fashion. However, in the FNN, the training is carried out using the information from both designer's experiences and sample data sets. The other drawback of the approach is that the number of fuzzy rules increases exponentially with respect to inputs and as a consequence 17 rules are framed for 3 inputs only [8].

In this paper a simple neural network is used to implement a fuzzy-rule-based locator of a power system from input/output data for a transmission line operating with a TCSC. The MOVs, air-gaps and thyristor firing arrangement are designed and simulated using the EMTDC subroutines. In this approach, the FNN model can be viewed either as a fuzzy system, a neural network or a fuzzy-neural system. The structure is seen in neural viewpoint for training and fuzzy viewpoint is utilized to gain inside into the system and to simplify the model. Unlike earlier approach, in this strategy the number of rules needed is determined by the data itself and consequently a smaller number of rules are produced. The network is trained using EKF algorithm. To have a compact structure, a pruning strategy eliminates the redundant rules and fuzzification neurons. The location task is accomplished by using different FNNs.

2 System Studied

A 400kV, 50Hz power system is illustrated in Fig.1. In this system a TCSC is located on a transmission line is used for the distance protection study. The power system consists of two sources, TCSC and associated components and a 300 km transmission

line. The transmission line has zero sequence parameter $Z(0)=96.45+j335.26$ ohm and positive sequence impedance $Z(1)=9.78+j110.23$ ohm. $E_s = 400$ and $E_R = 400 \angle \delta$. The TCSC is designed to provide compensation varied from 30%(minimum) to 40%(maximum). All the components are modeled using the EMTDC subroutines. The sampling frequency is 1.0 kHz at 50 Hz base frequency. The metal oxide varistor (MOV) consists of a number of zinc oxide disks electrically connected in series and parallel. The purpose of the MOV is to prevent the voltage across the capacitor from rising to levels which will damage the capacitor. This is most likely to happen when a fault occurs at a point on the compensated line which minimizes the impedance of the fault loop. When instantaneous voltage across the capacitor approaches a dangerous level the MOV begins to draw a significant proportion of the line current thereby limiting the voltage across the capacitor at that level. Further, a bypass switch in parallel with the gap automatically closes for abnormal system conditions that cause prolonged current flow through the gap. The small inductance in the arrangement limits the current through the air-gap or switch circuit.

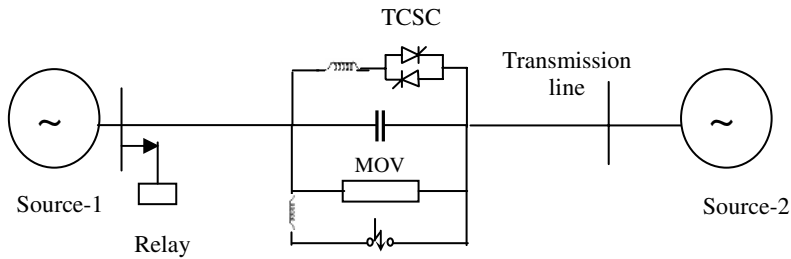


Fig. 1. The TCSE based line

3 Fuzzy Neural Network

The fuzzy neural network, comprising by input, fuzzification, inference and defuzzification layers. Further the network can be visualized as consisting of N inputs, with N neurons in the input layer and R rules, with R neurons in the inference layer. There are NxR neurons in the fuzzification layer and K neurons for output layer. The signal propagation and basic function in each layer of the FNN is introduced in the following. The input layer consists of x_i , $i = 1, 2, \dots, N$, along with unity. Each neuron in the fuzzification layer represents a fuzzy membership function for one of the input variables. The activation function used in this layer is $f(net_{ij}) = \exp(-|net_{ij}|^{l_{ij}})$ and the input to these neurons $net_{ij} = w_{ij1} x_i + w_{ij0}$, with w_{ij1} and w_{ij0} being the connecting weights between input layer and fuzzification layer. Thus, the output of the fuzzification layer becomes

$$\mu_{ij}(x_i) = \exp(-|w_{ij1}x_i + w_{ij0}|^{l_{ij}}) \tag{1}$$

where μ_{ij} is the value of fuzzy membership function of the i^{th} input variable corresponding to the j^{th} rule. The connections between fuzzification and inference layers have unity weights (shown in the figure as I). Each node j in the inference layer is denoted by Π , which multiplies the input signals and the output of the node becomes the result of product. Therefore, the output of the layer becomes

$$\rho_j(x_1, x_2, \dots, x_N) = \prod_i^N \mu_{ij}(x_i) \tag{2}$$

With v_{jk} being the output action strength of the k^{th} output associated with the j^{th} rule and utilizing weighted sum defuzzification, the network output becomes

$$\begin{aligned} o_k(x_1, x_2, \dots, x_N) &= \sum_j^R v_{jk} \rho_j(x_1, x_2, \dots, x_N) \\ &= \sum_j^R v_{jk} \prod_i^N \exp(-|w_{ij1}x_i + w_{ij0}|^{l_{ij}}) \end{aligned} \tag{3}$$

Inputs to the fuzzification layer are the variables used to define fuzzy operating regions each of these variables is transformed into several fuzzy sets in the fuzzification layer. appropriately shaped membership function at different positions can be obtained by changing the weights and parameters of the fuzzy neural network. Each neuron in rule layer corresponds to a fuzzy operating region of the particular classification objective. Its inputs are obtained from the fuzzification layer and its output is the product of its inputs and is the membership function of the corresponding fuzzy operating region. there are no weights to be estimated in this layer.

4 EKF Training Algorithm

The fuzzy neural network is trained using Extended Kalman Filter (EKF) algorithm.. Here the first step is to organize the weights as a state vector $w(t)$, where t counts for iterations. The desired output can be defined as

$$o(t) = f[w(t), x(t)] + \epsilon(t) \tag{4}$$

where $o(t)$ is the output, $x(t)$ is the input and $\epsilon(t)$ is discrepancy of the desired output from the estimated output. The state estimation is then to determine $w(t)$ to minimize the cost function J defined as

$$J = \frac{1}{2} \sum_t \|o(t) - f[w(t), x(t)]\|^2 \tag{5}$$

The EKF formula can be found out by Taylor series expansion of $f[w(t), x(t)]$. The estimated output $o'(t)$ is as:

$$o'(t) = f[w'(t), u(t) + H(t)(w(t) - w'(t)) + G(t)(x(t) - x'(t)) + HOT \tag{6}$$

where
$$H(t) = \left(\frac{\partial f[w(t), x(t)]}{\partial w(t)} \right)^T \tag{7}$$

and
$$G(t) = \left(\frac{\partial f[w(t), x(t)]}{\partial x(t)} \right)^T \tag{8}$$

HOT is the higher order terms and $x'(t)$ is the unpredicted input in the next iteration. Neglecting the higher order terms the discrepancy between desired output and estimated output will be

$$o''(t) = o(t) - o'(t) = H(t)(w(t) - w'(t)) + G(t)(x(t) - x'(t)) \quad (9)$$

Now the update equations are

$$S(t+1) = H(t+1)P(t)H(t+1)^T + G(t+1)\sum_x (t+1)G(t+1)^T + R(t+1) \quad (10)$$

$$K(t+1) = P(t)H(t+1)^T S(t+1)^{-1} \quad (11)$$

$$P(t+1) = P(t) - K(t+1)H(t+1)P(t) \quad (12)$$

$$w(t+1) = w(t) + K(t+1)\{o(t) - o'(t)\} \quad (13)$$

Where $K(t)$ is the Kalman gain and $P(t)$ is the weight covariance matrix. The variance of the each output can be obtained from the diagonal elements of the innovation matrix $S(t+1)$. $R(t)$ is the noise co-variance matrix. Initially $R(t)=I$. During training, the number of rules is increased from 1 till a satisfactory performance of the network is found. The initial weights are randomly selected in the interval $[-1, +1]$. The maximum number of iteration is set to 3000 in all cases. The training is continued till $\varepsilon(t) < 1 \cdot e^{-4}$ at all points for a window length of 100 or the number of iteration reaches its maximum during training. The initial value of covariance matrix is chosen as $P(t) = 100 I$, where I is unit matrix. As this rule does not contribute to the network performance, the rule should be pruned.

5 Training and Computational Result

For LG fault type the first two elements of the input vector are the faulty phase current and voltage, the seventh one for the zero sequence component, eighth is for system frequency and the ninth one for the firing angle of TCSC. The first four inputs of the LL or LLG locators are for the corresponding voltage and current of faulty phases, seventh is for the negative sequence component, eighth one represents for frequency and the ninth input element is the firing angle of the TCSC. However the LLL fault locator does not include sequence component as an input (total 8 inputs). The training data sets include fault situations for different prefault conditions, inception angles, system frequencies, fault resistances and fault distances. The total number such sets is 80 for all four FNNs. The networks are trained by EKF algorithm and pruning strategy is applied. The structure of FNN1 (L-G) locator becomes 9 inputs, 7 rules, 61 fuzzification neurons and 1 output. Similarly for FNN2 (LL-G) the number of rules and fuzzification neurons are 8 and 69, respectively and for FNN3 (LL) 8, 71 and FNN4 (LLL-8 inputs) 8, 64, respectively.

These networks are tested at different situations of the power system by varying frequency, load angle, fault resistance, inception angle and source capacities. Further network performance is studied by adding noise to the input signals Table-1 through Table-4 show the location for LG, LL-G, LL and LLL faults respectively at a fault distance of 15%, 35%, 55%, 75% and 95 % of the line, respectively and fault resistance of 10 to 200 Ω . The above results are obtained for the input vectors at 10ms after the fault. The maximum error found for these networks is less than 4%.

Table 1. Fault Location for L-G faults (FNN-1)

Distance (%)	Fault Resistance (R _f)	Error (%)
15	10	2.01
	200	2.58
35	10	1.57
	200	1.23
55	10	2.04
	200	2.33
75	10	2.56
	200	1.44
95	10	2.47
	200	2.89

Table 2. Fault Location for LL-G faults (FNN-2)

Distance (%)	Fault Resistance (R _f)	Error (%)
15	10	1.58
	200	1.98
35	10	2.01
	200	2.55
55	10	2.53
	200	1.98
75	10	1.05
	200	1.26
95	10	1.88
	200	2.04

Table 3. Fault Location for LL faults (FNN-3)

Distance (%)	Fault Resistance (R _f)	Error (%)
15	10	1.27
	200	2.01
35	10	1.36
	200	2.11
55	10	2.65
	200	2.32
75	10	2.17
	200	3.12
95	10	2.49
	200	3.55

Table 4. Fault Location for LLL faults (FNN-4)

Distance (%)	Fault Resistance (R _f)	Error (%)
15	10	2.11
	200	2.54
35	10	1.98
	200	2.22
55	10	2.58
	200	1.98
75	10	1.23
	200	1.56
95	10	1.23
	200	2.03

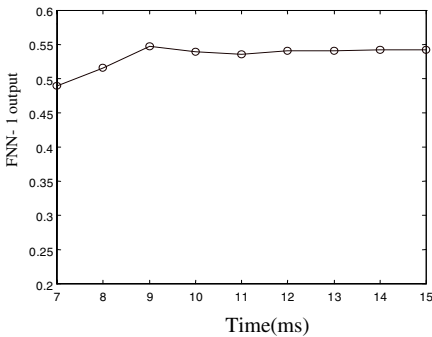


Fig. 2. Convergence loci of the FNN-1(L-G) for 'ag' fault at 55% of the line, 45° inception angle, R_f=100Ω

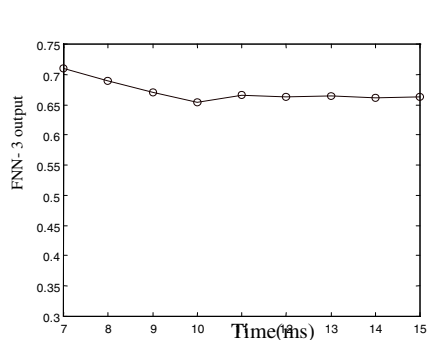


Fig. 3. Convergence loci of the FNN-3(LL) for 'ab' fault at 65% of the line, 30° inception angle, R_f=100Ω

The convergence speed of the designed networks (FNNs) are tested to the corresponding input vector on sample to sample basis. Results are presented in Figs.2 shows the output of FNN-1(L-G) for 55% of line, 45° inception angle at a fault resistance of 100 Ω. Similarly Fig.3 shows the output of FNN-3(LL) for 65% of line, 30°

inception angle at a fault resistance of 100Ω . The test cases demonstrate the superior estimation accuracy and speed of the FNNs protection scheme. As seen from the figures the FNN converges within half cycle of fault inception, which depicts the fastness of the proposed algorithm.

6 Conclusions

The paper presents a novel approach for distance relaying of flexible ac transmission line using fuzzy neural network. The FNN is used to calculate the location of the fault from the relaying point. The FNN parameters are updated by EKF algorithm and a pruning strategy result in a compact network structure. The fault location networks for the transmission line employing a TCSC are tested under a variety of different system conditions and fault situations. The results from the FNN indicate the accuracy of the proposed system for protection. Also the convergence speed is very fast to the requisite values of the output variables under fault conditions of the power system. Hence the proposed method found to be very accurate and fast for protection of flexible ac transmission line including TCSC.

References

1. Kimbark E W: "Improvement of System Stability by Switched Series Capacitors", IEEE Transactions on Power Apparatus and Systems (1966), Vol. PAS-85, NO. 2, pp. 180-188..
2. Smith O J M: "Power System Transient Control by Capacitor Switching", IEEE Transactions on Power Apparatus and Systems (1969), Vol. PAS-88, NO. 1, January 1969, pp. 28-35.
3. Flexible AC Transmission Systems (FACTS): Scoping Study, Vol. 2, Part 1: Analytical Studies", EPRI Report EL-6943, September 1991.
4. Larsen E, Bowler C, Damsky B, Nilsson S: "Benefits of Thyristor Controlled Series Compensation", CIGRE Paper 14137138-04, Paris, 1992.
5. L. Gyugyi, "Unified Power flow concept for flexible AC Transmission systems", IEE Proc.-C, Vol. 139, No. 4, 1992, pp.323-332. M. Noroozian, L. Angquist, M. Ghandhari, and G. Anderson, "Improving Power System Dynamics by series connected FACTS devices", IEEE Trans. on Power Delivery, Vol.12, No.4, 1997, pp.1635-1641.
6. E. V. Larsen, K. Clark, S. A. Miske and J. Urbanek, "Characteristics and Rating consideration of Thyristor Controlled Series Compensation", IEEE Trans. on Power Delivery, Vol. 9, No. 2, 1994, pp.992-999.
7. Y. H. Song, A.T. Johns, Q.Y. Xuan, "Artificial Neural Network Based Protection Scheme for Controllable series-compensated EHV transmission lines", IEE Proc. Gen. Trans. Disbn. Vol.143, No.6, 1996, pp.535-540
8. H. Wang, and W. W. L. Keertipala, "Fuzzy-Neuro Approach to Fault Classification for Transmission Line Protection", IEEE Trans. on Power Delivery, Vol. 13, No.4 1998, pp.1093-1103

Computational Intelligence for Network Intrusion Detection: Recent Contributions

Asim Karim

Dept. of Computer Science, Lahore University of Management Sciences,
Opp. Sector U, DHA, Lahore 54792, Pakistan
akarim@lums.edu.pk

Abstract. Computational intelligence has figured prominently in many solutions to the network intrusion detection problem since the 1990s. This prominence and popularity has continued in the contributions of the recent past. These contributions present the success and potential of computational intelligence in network intrusion detection systems for tasks such as feature selection, signature generation, anomaly detection, classification, and clustering. This paper reviews these contributions categorized in the sub-areas of soft computing, machine learning, artificial immune systems, and agent-based systems.

1 Introduction

Network intrusion detection (NID) is essentially a pattern recognition problem in which network traffic patterns are classified as either 'normal' or 'abnormal'. This is a difficult problem because of the wide diversity of traffic patterns and the need for accuracy in real-time operation. The NID problem has been tackled since the early days of computer networks but an efficient, effective, and practical solution is still being sought [30]. The incorporation of computational intelligence in network intrusion detection systems (NIDS) presents the greatest potential for an acceptable solution. Computational intelligence has yielded successful solutions to similar problems in other domains such as the highway incident detection problem [12]. Like the NID problem, the highway incident detection problem requires rapid and reliable identification of incidents (e.g. accidents) from raw traffic data obtained from sensors located at different points on the highway network.

In recent years significant contributions have been made towards the solution of the NID problem. Many of these contributions employ computational intelligence approaches derived and motivated from concepts of biological intelligence. This paper reviews recent computational intelligence contributions in the areas of soft computing (section 2), machine learning (section 3), artificial immune systems (section 4), and agent-based systems (section 5).

2 Soft Computing

Neural networks, fuzzy systems, and evolutionary algorithms have been used extensively for network intrusion detection. Neural networks are known to be

effective classifiers and feature selectors / dimensionality reducers. As such, neural networks have been a component of several recent NIDSs. As part of the MAID intrusion detection system, a BP neural network is used for identifying DoS attacks [16]. The inputs to the neural network are statistically preprocessed SNMP MIB (management information base) data. Ng et al. use a stochastic radial-basis function neural network to extract and rank features for DoS attacks from the DARPA database [37]. NIDSs based on support vector machines (SVMs) and neural networks, and the study of various features for intrusion detection based on the performance of SVM and neural network classifiers is presented in [4, 34]. It has been found that SVMs perform better than neural networks for intrusion classifications. Liu et al. describe the design and performance of an anomaly detection system using the ART neural network [40]. Unlike the previously mentioned neural network based systems, this system can be trained in real-time operations. Valdes uses a competitive neural network to cluster unlabeled data into groups [3]. This overcomes the issue of having labeled normal and abnormal traffic for training. Sarasamma et al. propose a hierarchical Kohonen network model for detecting anomalies and outliers in connection events database [35]. Each level is a simple winner-take-all neural network that identifies features in the problem space. In this way, a clustering of the data is obtained which is then labeled using confidence measures.

Fuzzy sets and fuzzy logic have been used successfully for problems involving vagueness and imprecision including NID. The FIRE detection system, which is based on fuzzy techniques, uses the fuzzy cognitive map for decision making [17]. In the system design, several features in TCP, UDP, and ICMP protocol headers are evaluated by modeling some of them as fuzzy variables and visualizing them to ascertain their 'fuzziness'. Lee and Mikhailov propose a fuzzy classification system for intrusion detection based on numeric features [19], while Shah et al. describe a fuzzy clustering approach applied to statistics obtained from low level kernel system and network data [13]. A hybrid neuro-fuzzy intrusion detection system is described in [32].

Evolutionary algorithms (EAs) provide robust solutions to many problems by adopting a process of selection and evolution similar to the natural process of evolution. For NID, evolutionary algorithms have proven useful for signature generation and feature selection. Pillai et al. describe a genetic algorithm (GA) for generating high quality rules from the DARPA intrusion detection database [23]. The generation of fuzzy rules or signatures from anomaly traffic is described in [9]. These rules serve as negative selectors in an immunity-based intrusion detection system. Florez et al. present a fuzzy association rule mining approach for generating fuzzy rule sets [11]. A threshold on the similarity between different (fuzzy) sets of rules is used to detect intrusions. Middlemiss and Dick use GA to rank features by evolving feature weights over multiple generations with a nearest neighbor rule for fitness function [22]. An evolutionary algorithm has also been used to optimally design a radial-basis function neural network for intrusion detection [1]. Song et al. present an efficient algorithm for training from large data sets using linear structured genetic programming (GP) [7]. The GP algorithm is able to find features from the KDDCup-99 data set that contains 0.5 million records in 15 minutes. The design, implementation, and evaluation of GA-based misuse detection system is described in [31]. A recent study of the suitability of a fitness function for NID is presented in [28].

3 Machine Learning

Several machine learning and data mining techniques have been proposed for NID. Markov models have been used for capturing the sequence of events in network traffic and determining the probabilities of significant changes from the norm. Ye et al. study the performance of a Markov-chain model of normal events for detecting abnormal ones by applying it to UNIX system logs [25], while Anming and Chunfu investigate hidden Markov models for network intrusion detection [39]. Zhang and Zhu integrate hidden Markov models and neural networks for NID [36]. In general, Markov based models are efficient but often less accurate because of the wide diversity of normal behaviors. Decision tree models have been employed to analyze protocol headers for feature selection and intrusion detection [33], while a comparison of naïve Bayes and decision trees is provided in [26]. Cho utilizes both soft and hard computing approaches by integrating a hidden Markov model of normal traffic with neural network and fuzzy logic [33]. Feature selection and dimensionality reduction is done with a SOM, construction of a database of normal sequences is performed with a Markov model, and a fuzzy inference system is used for decision making. A NIDS based on a probabilistic data mining approach for learning rules of normal and abnormal traffic is presented in [24].

4 Artificial Immune Systems

The human immune system is an example of an efficient and effective intrusion defense system. It is capable of identifying beneficial (self) and harmful (non-self) elements in the body and taking action to expel or eradicate unwanted elements. This defense system has a direct analogy to the network intrusion detection and response system. It is therefore not surprising that immunity-based NIDS have been proposed since the 1990s. The recent works in this area have focused on developing detectors for self and non-self. An investigation of the parameters involved in the creation of non-self detectors is presented in [15], while in [18] EA is employed to create hyper-ellipsoid detectors for negative selection. Gomez et al. describe the generation of fuzzy rules that characterize the non-self of an immunity-based NIDS [14]. Hang and Dai test an immunity-based NIDS with both positive and negative selection classes [38]. A study of anomaly detection using different features of traffic and protocol header in the context of an immunity-based NIDS is presented in [10]. Esponda et al. present a formal framework for positive and negative selection in immunity-based systems [8]. A survey of immunity-based computer defense systems and discussion of future development trends is presented in [20].

5 Agent-Based Systems

Computer networks are distributed in nature. As such, distributed agent-based systems are commonly proposed for their security including intrusion detection. Miller and Inoue describe the performances of distributed NIDSs in which agents perform local feature extraction using SOMs and global decision making [27].

Dasgupta and Brian present a distributed architecture for network security using packet, process, system, and user information [6]. It combines profile-based anomaly detection and parametric pattern matching in an agent-based system motivated from the human immune system [5]. Specific agents implement neural network classification and fuzzy inference for decision making at a central location. Siraj et al. describe their intelligent intrusion detection system which involves distributed information collection and central information processing and decision making using fuzzy cognitive maps [2]. An intelligent agent based distributed architecture is presented in [29], while a cooperative/collaborative architecture is presented in [41]. Zhou et al. propose a cooperative model for network security in which different elements of the security chain – firewalls, intrusion detection, VPNs – interact among themselves while maintaining a certain degree of independence [21].

6 Conclusion

Computational intelligence has figured prominently in many proposed solutions to the network intrusion detection problem in the recent past. In these solutions, computational intelligence provides functionalities such as feature selection, signature generation, anomaly detection, and decision making. This paper reviews these solutions categorized into the sub-areas of soft computing, machine learning and data mining, artificial immune systems, and agent-based systems. It is observed that computational intelligence holds great promise for an effective and practical solution to the network intrusion detection problem.

Acknowledgment

This work was supported by a research grant from PTCL R&D Fund, Pakistan.

References

1. A. Hofmann, T. Horeis, and B. Sick: Feature Selection for Intrusion Detection: An Evolutionary Wrapper Approach. Proc., International Joint Conference on Neural Networks (IJCNN '04) (2004) 1563-1568
2. A. Siraj, R.B. Vaughn, and S.M. Bridges: Intrusion Sensor Data Fusion in an Intelligent Intrusion Detection System Architecture. Proc. Hawaii International Conference on System Sciences (2004) 902-911
3. A. Valdes: Detecting Novel Scans Through Pattern Anomaly Detection. Proc. DARPA Information Survivability Conference and Exhibition (DICEX '03) (2003) 140-151
4. A.H. Sung and S. Mukkamala: Identifying Important Features for Intrusion Detection Using Support Vector Machines and Neural Networks. Proc. Symposium on Applications and the Internet (SAINT '03) (2003) 209-216
5. D. Dasgupta and F. Gonzalez: An Immunity-based Technique to Characterize Intrusions in Computer Networks. IEEE Transactions on Evolutionary Computing, Vol. 6, No. 3 (2002) 281-291

6. D. Dasgupta and H. Brian: Mobile Security Agents for Network Traffic Analysis. Proc. DARPA Information Survivability Conference and Exhibition (2001) 332-340
7. D. Song, M.I. Haywood, A.N. Zincir-Heywood: Training Genetic Programming on Half a Million Patterns: An Example from Anomaly Detection. IEEE Transactions on Evolutionary Computation, Vol. 9, No. 3 (2005) 225-239
8. F. Esponda, S. Forrest, and P. Helman: A Formal Framework for Positive and Negative Decision Schemes. IEEE Transactions on Systems, Man, and Cybernetics – Part B (Cybernetics), Vol. 34, No. 1 (2004) 357-373
9. F. Gonzalez, J. Gomez, M. Kaniganti, and D. Dasgupta: An Evolutionary Approach to Generate Anomaly (Attack) Signatures. Proc. IEEE International Workshop on Information Assurance (IWIA '03) (2003) 251-259
10. F. Serezynski: Some Issues in Solving the Anomaly Detection Problem Using the Immunological Approach. Proc. IEEE International Parallel and Distributed Processing Symposium (IPDPS '05) (2005) 188-195
11. G. Florez, S.M. Bridges, and R.B. Vaughn: An Improved Algorithm for Fuzzy Data Mining for Intrusion Detection. Proc. North American Fuzzy Processing Society (2002) 457-462
12. H. Adeli and A. Karim: Wavelets in Intelligent Transportation Systems. John Wiley & Sons UK (2005)
13. H. Shah, J. Undercoffer, and A. Joshi: Fuzzy Clustering for Intrusion Detection. Proc. IEEE International Conference on Fuzzy Systems (2003) 1274-1278
14. J. Gomez, F. Gonzalez, and D. Dasgupta: An Immuno-Fuzzy Approach to Intrusion Detection. Proc. IEEE International Conference on Fuzzy Systems (2003) 1219-1224
15. J. Kim and P.J. Bentley: Towards an Artificial Immune System for Network Intrusion Detection: An Investigation of Clonal Selection with a Negative Selection Operator. Proc. Congress on Evolutionary Computing (2001) 1244-1252
16. J. Li and C. Manikopoulos. Early Statistical Anomaly Intrusion Detection of DoS Attacks Using MIB Traffic Parameters. Proc. IEEE International Workshop on Information Assurance (IWIA '03) (2003) 53-59
17. J. Xin, J.E. Dickerson, and J.A. Dickerson. Fuzzy Feature Extraction and Visualization for Intrusion Detection. Proc. IEEE International Conference on Fuzzy Systems (2003) 1249-1254
18. J.M. Shapiro, G.B. Lamont, and G.L. Peterson: An Evolutionary Algorithm to Generate Hyper-Ellipsoid Detectors for Negative Selection. Proc. GECCO '05 (2005) 337-344
19. K. Lee and L. Mikhailov: Intelligent Intrusion Detection System. Proc. IEEE International Conference on Intelligent Systems (2004) 497-502
20. K.P. Anchor, P.D. Williams, G.H. Gunsch, and G.B. Lamont: The Computer Defense Immune System: Current and Future Research in Intrusion Detection. Proc. Congress on Evolutionary Computing (2002) 1027-1032
21. L. Zhou, F. Liu, and J. Wu: Research on Co-operative Computer Network Security Technologies. Proc. IEEE International Conference on Systems, Man, and Cybernetics (2004) 1164-1168
22. M.J. Middlemiss and G. Dick: Weighted Feature Extraction Using a Genetic Algorithm for Intrusion Detection. Proc. Congress on Evolutionary Computing (2003) 1669-1675
23. M.M. Pillai, J.H.P. Eloff, and H.S. Venter: An Approach to Implement an Intrusion Detection System Using Genetic Algorithms. Proc. SAICSIT '04 (2004) 228-235
24. M.V. Mahoney and P.K. Chan: Learning Rules for Anomaly Detection of Hostile Network Traffic. Proc. IEEE International Conference on Data Mining (ICDM '03) (2003) 601-604

25. N. Ye, Y. Zhang, and C.M. Borrer: Robustness of the Markov Chain Model for Cyber-Attack Detection. *IEEE Transactions on Reliability*, Vol. 53, No. 1 (2004) 116-123
26. N.B. Amor, S. Benferhat, and Z. Elouedi: Naïve Bayes vs Decision Trees in Intrusion Detectin Systems. *Proc. SAC '04* (2004) 420-424
27. P. Miller and A. Inoue: Collaborative Intrusion Detection System. *Proc. North American Fuzzy Information Processing Society* (2003) 519-524
28. P.A. Diaz-Gomez and D.F. Hougen: Analysis and Mathematical Justification of a Fitness Function Used in an Intrusion Detection System. *Proc. GECCO '05* (2005) 1591-1592
29. Q. Xue, L. Guo, and J. Sun: The Design of a Distributed Network Intrusion Detection System IA-NIDS. *Proc. International Conference on Machine Learning and Cybernetics* (2003) 2305-2308
30. R.A. Kemmerer and G. Vigna: Intrusion Detection: A Brief History and Overview. *IEEE Computer* (2002) 27-30
31. R.H. Gong, M. Zulkernine, and P. Abolmaesumi: A Software Implementation of a Genetic Algorithm Based Approach to Network Intrusion Detection. *Proc. International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel Distributed Computing* (2005) 246-253
32. S. Chavan, K. Shah, N. Dave, and S. Mukherjee: Adaptive Neuro-Fuzzy Intrusion Detection Systems. *Proc. International Conference on Information Technology: Coding and Computing (ITCC '04)* (2004) 70-74
33. S. Cho: Incorporating Soft Computing Techniques into a Probabilistic Intrusion Detection System. *IEEE Transactions on Systems, Man, and Cybernetics – Part C (Applications and Reviews)*, Vol. 32, No. 2 (2002) 154-160
34. S. Makkamala and A.H. Sung: Detecting Denial of Service Attacks Using Support Vector Machines. *Proc. IEEE International Conference on Fuzzy Systems* (2003) 1231-1236
35. S.T. Sarasamma, Q.A. Zhu, and J. Huff: Hierarchical Kohonen Net for Anomaly Detection in Network Security. *IEEE Transactions on Systems, Man, and Cybernetics – Part B (Cybernetics)*, Vol. 35, No. 2 (2005) 302-312
36. T. Abbes, A. A. Bouhoula, and M. Rusinowitch: Protocol Analysis in Intrusion Detection using Decision Tree. *Proc. International Conference on Information Technology, Coding, and Computing (ITCC '04)* (2004) 404-408
37. W. Ng, R. Chang, and D. Yeung: Dimensionality Reduction for Denial of Service Detection Problems Using RBFNN Output Sensitivity. *Proc. International Conference on Machine Learning and Cybernetics* (2003) 1293-1298
38. X. Hang and H. Dai: Applying Both Positive and Negative Selection to Supervised Learning for Anomaly Detection. *Proc. GECCO '05* (2005) 345-352
39. X. Zhang and Z. Zhu: Combining the HMM and the Neural Network Models to Recognize Intrusions. *Proc. International Conference on Machine Learning and Cybernetics* (2004) 956-961
40. Y. Liu, D. Tian, and A. Wang: ANNIDS: Intrusion Detection System Based on Artificial Neural Network. *Proc. International Conference on Machine Learning and Cybernetics* (2003) 1337-1342
41. Y. Xiaoping and D. Yu: An Auto-Configuration Cooperative Distributed Intrusion Detection System. *Proc. World Congress on Intelligent Control and Automation* (2004) 279-283
42. Z. Anming and J. Chunfu: Study on the Applications of Hidden Markov Models to Computer Intrusion Detection. *Proc. World Congress on Intelligent Control and Automation* (2004) 256-260

Design of a Switching PID Controller Using Advanced Genetic Algorithm for a Nonlinear System

Jung-Shik Kong¹, Bo-Hee Lee², and Jin-Geol Kim³

¹Dept. of Automation Eng., Inha University, YongHyun-Dong, Nam-Gu, Incheon, Korea
tempus@dreamwiz.com

²School of Electrical Engineering, Semyung University, ShinWal-Dong, Chechon, Korea
bhlee@semyung.ac.kr

³School of Electrical Eng., Inha University, YongHyun-Dong, Nam-Gu, Incheon, Korea
john@inha.ac.kr

Abstract. This paper deals with a switching PID controller using a genetic algorithm in a multi-nonlinear system. In controlling the nonlinear element of the system, there are some problems such as the limit cycle. In this study, a switching PID controller was proposed to solve problems caused by nonlinearities of system. The PID is a well-known robust controller. But, in a motor system case, it may have a limit cycle when proportional gains exceed limit. However, in other case, if the PID gain is relatively small, its torque characteristics can be too weak. In this case, the suggested switching PID controller was found to be a good approach for solving these problems despite there being difficulties in determining its boundary and gains at each boundary. In this paper, an improved genetic algorithm was used for identifying a motor system and to determining each gain for the controller. In particular, new type of crossover and mutation using a sigmoid function is applied to improve the searching ability based on the proposed improved genetic algorithm. All the processes are investigated through simulations and are verified experimentally in a real motor system.

1 Introduction

In most fields of control engineering, some problems are often caused by the system nonlinearity. This nonlinearity can generate the limit cycle of a system and often make the system unstable. In control engineering, the nonlinearity of a motor system is the representative area to be controlled. This nonlinearity can be classified by internal and external nonlinear elements. Internal nonlinear elements exist between a motor and a controller such as a saturation and Coulomb friction. In contrast, the external nonlinear elements also exist in the external structure combined with driving the motor. Backlash is a good example of external nonlinearities. In addition, most of the motor systems include those nonlinearities as a combined form, which lead to be difficult to eliminate the effect of nonlinearity efficiently.

Many studies have been performed to reduce these problems [1, 2]. E. J. Davison [3] introduced a Describing Function as a method of frequency analysis and S. Komada[4] suggested a disturbance observer to reduce the limit cycle by a nonlinear

parameter. K.T. Woo and C.W. Tao [5, 6] proposed a method using an AI algorithm. The AI algorithm is very useful for determining an adaptable solution when the system information was not known accurately. A genetic algorithm especially has an excellent capability for searching a method within a boundary. A genetic algorithm has been used for searching optimal control parameters based on artificial intelligence [7].

In this paper, a switching PID controller is applied to control a motor with multi-nonlinearities. Two methods are performed to search for the optimal switching PID gain. One is the identifying the motor by a genetic algorithm, which is needed to determine the accurate motor parameter because the optimal switching PID gain is based on an accurate motor model. Most motor identification processes do not consider the effect from Coulomb friction during the simulation time. However, there is often a difference between the simulation model and the real model and the exact modeling is needed. Therefore, an accurate motor model including Coulomb friction is needed by computational methods such as a genetic algorithm. The other genetic algorithm is employed to search the optimal controller gain. These optimal gains lead to control system without any limit cycles. This paper also suggests a simple compensator to reduce the external nonlinear elements. All the processes are performed using simulation programs and are verified on an actual motor control system.

2 Manuscript Preparation

Most of the motor systems have a linear and nonlinear part at the same time. The motor including the mechanical structure is regarded as a linear part, while the voltage saturation and backlash from the reduction gear is considered to be a nonlinear part. This nonlinearity might cause the limit cycle and often have a negative influence on the system.

A motor transfer function between the voltage (V) of the motor and the angle (θ) of the joint can be defined as follows:

$$\frac{\theta(s)}{V(s)} = \frac{K_a}{L_a J_m s^3 + (L_a f_m + R_a J_m) s^2 + (R_a f_m + K_a K_b) s} \quad (1)$$

where $V(s)$, $\theta(s)$, L_a , R_a , J_m , f_m , K_a and K_b are the input voltage, output angle, armature inductance, armature resistance, moment of inertia of the motor, viscous-friction coefficient of the motor, the motor-torque proportional constant, and a proportional constant from back electromotive force, respectively.

The saturation represents the maximum voltage from the hardware driver. This is caused by the maximum driving capability of the motor power driver. The motor system also has backlash when the motor is mounted on an external structure. So, in this paper, the nonlinearities are regarded as saturation and backlash, which make the system design difficult to model with unique functions.

The motor system has several nonlinear parameters, which are static friction and kinetic friction. Fig. 1 shows a block diagram of a real motor with some frictions.

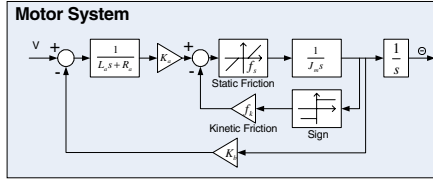


Fig. 1. Detailed block diagram of a real motor model

where f_s and f_k are the static friction and kinetic friction, respectively. The friction appears proportionally to the velocity. It is difficult to express a deterministic equation for the real motor model due to its nonlinearities. In this case, the parameters of a motor model can be identified using a computational algorithm such as a genetic algorithm (GA). These AI algorithms have been proposed previously [8, 9].

3 Genetic Algorithm

3.1 Improved GA

GA is considered as one of the searching algorithms in a global area. Finding the optimal solution without a differential equation is the main advantage of the GA.

In this paper, an improved type of genetic algorithm is suggested. The previous method was separated into two sections. One was a Binary-coded Genetic Algorithm (BGA) and the other was a Decimal-Convex Genetic Algorithm (DGA). Fig. 2 shows the crossover and mutation processes of the BGA and DGA, respectively.

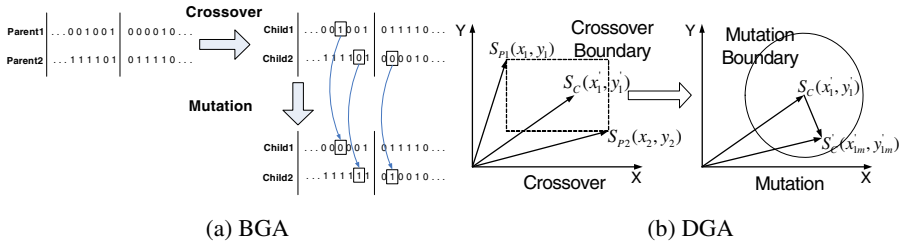


Fig. 2. Crossover and mutation of the BGA and DGA

In Fig. 2(b), $S_{p1}(x_1, y_1)$ and $S_{p2}(x_2, y_2)$ are a set of parents, $S_c(x'_1, y'_1)$ is the result of crossover and $S'_c(x'_1, y'_1)$ is the output of the mutation. Here, x and y represent the parameters to be determined using GA. BGA has the advantage of being able to apply genetic concepts easily. However, the algorithms' own accuracy limits the searching precision for an optimal solution. In addition, DGA can search for a more precise optimal solution than the BGA because it uses real values. However, it settles down with increasing time, because this algorithm is dictated by time with the number of generations.

In this study, a more advanced algorithm was proposed to improve the problems associated with the previous algorithms. This method basically includes a DGA. As

shown in Fig. 2(b), the DGA has the maximum crossover boundary between two parents during the crossover. In a conventional decimal-coded genetic algorithm, the fitness function is settled after passing the generation count and the optimal data is searched for at this time. The data, which remains close to the optimal value, may not be the desired data. If crossover and mutation boundary is reduced while the generation count has passed, a more optimal solution can be searched for. In order to solve this problem, a sigmoid function was used in crossover and mutation. Equations (2-3) show the proposed crossover and mutation. Here, λ is the random number ($\lambda \in (0,1)$). Since the parents influence the child's crossover, the child value should be within the crossover boundary. In addition, mutation d denotes the maximum boundary altered by the mutation, and α is a random number.

$$\begin{aligned} x'_1 &= x_1 + \delta(t) \times (1 - \lambda_1) x_2, & x'_2 &= x_2 + \delta(t) \times (1 - \lambda_2) x_1 \end{aligned} \tag{2}$$

$$y'_1 = y_1 + \delta(t) \times (1 - \lambda_3) y_2, \quad x'_2 = y_2 + \delta(t) \times (1 - \lambda_4) y_1$$

$$x'_{1M} = x'_1 + \delta(t) \times \text{sgn}(\alpha) \times d \times \lambda_m, \quad y'_{1M} = y'_1 + \delta(t) \times \text{sgn}(\alpha) \times d \times \lambda_m \tag{3}$$

where $\delta(t) = \left(1 - 1/1 + e^{-(10t+5)}\right)$, t : generation count.

By using $\delta(t)$, relative boundary of the crossover and mutation is reduced according to the time. It will be selected the proper value to search more optimal solution. Fig. 3 shows the comparison average and maximum of the fitness values.

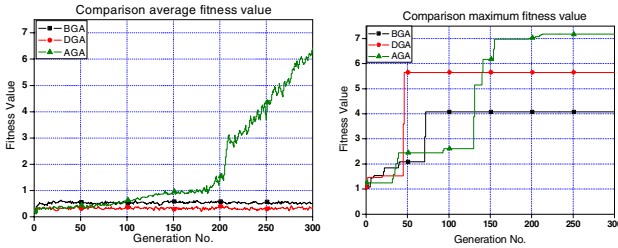


Fig. 3. Comparison of average and maximum fitness

Here, AGA is an acronym for an advanced decimal-coded genetic algorithm. AGA uses sigmoid function when the genetic algorithm is performed. It can improve the possibility for crossover result that can be closer than the DGA. From Fig. 4, the average fitness from the proposed GA has a relatively high fitness value. As the identification results show, a larger fitness value indicates a larger similarity to the real motor models.

3.2 Motor Identification Using GA

Motor identification is needed to search for accurate information on the motor. An accurate motor model can assist in determining the optimal controller gains. However, most motor models have nonlinear parameters, which make it difficult to determine a mathematical model. In this case, the GA is good to approach model that has no accurate parameters. Table 1 shows the parameters of the GA for a motor identification.

Here P_R is the position value of a real system and P_M is the position value of a mathematical model.

Table 1. Parameters of the GA for a motor identification

Parameters	Value	Parameters	Value
Generation No.	300	Crossover Rate.	0.6
Population No.	100	Mutation Rate	0.1
Fitness function	$fit = 1 / \sum P_R - P_M $		

3.3 Controller Design Using GA

A new controller is proposed to solve the combined nonlinearity efficiently, as shown in Fig. 4.

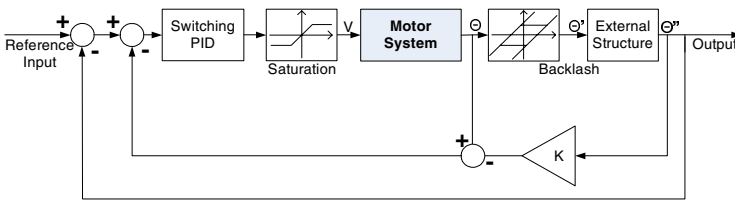


Fig. 4. Block diagram of proposed system

Two types of control blocks are inserted in order to control the nonlinearity. One is a new switching PID controller and the other is a compensator that can be represented as a gain K. The switching PID controller works as a main nonlinear controller, and the compensator is used for reducing the nonlinearity caused by the external structure.

System nonlinearities exist in a mixed structure as shown in the figure. The controller alters its proportional gains according to the tracking error. The error bound is determined as the square area within a uniform distance. In this study, error boundary is limited to 20 pulses for the motor encoder. The GA is used for searching for the optimal gains when a switching PID is applied. Actually, a nonlinear system is very sensitive to PID gains. If the proportional gain is too small, the torque characteristic will be bad, while the motor system has a limit cycle if PID gain is large. Therefore, a suitable method for changing the gain is required. However, the gain selection of an optimal switching PID is also difficult to search because of the sensitivity by nonlinearity. As a result, the GA was applied to search for the optimal switching gains. This is only possible if the motor model is accurate. Table 2 shows the genetic parameters applied to search for the switching PID gains.

Table 2. GA parameters of the switching PID gains

Parameters	Value	Parameters	Value
Generation No.	300	Crossover Rate	0.7
Population No.	100	Mutation Rate	0.2
Fitness function	$fit_s = (l_{max} - l_{min} + 1) / \sum (R_t - Y_t)$		

where l_{\max} , l_{\min} , R_t and Y_t are the maximum level of the limit cycle, the minimum level of the limit cycle, the reference value and the output value at each time t , respectively. The limit cycle is considered in the fitness function of the switching PID.

4 Result of Simulation and Experiment

In order to verify this suggestion, some simulations are performed using the motor control system, which was developed for our humanoid robot system. Table 3 details the parameters used in the motor model and Table 4 shows the tuning results of the switching PID gains by the GA.

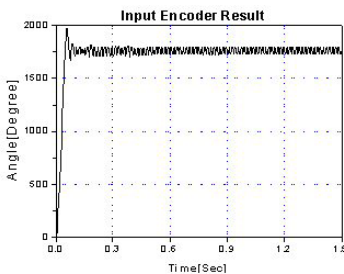
Using these gains, the limit cycle can successfully settled down. Fig. 5 shows the real experimental results for the general PID controller and the proposed switching PID one. A proportional term, an integrating term, and differentiating term of a

Table 3. Detailed parameters of a motor

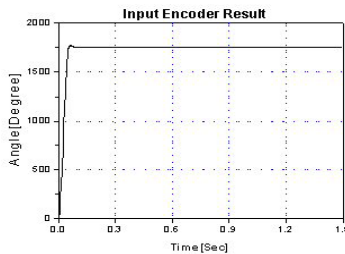
Sym.	Value	Sym.	Value	Sym.	Value	Symbol	Value
L_a	0.00034	J_m	8.09×10^{-7}	K_a	0.01933	f_s	0.00194
R_a	6.09146	f_m	1.51×10^{-5}	K_b	0.01688	f_k	0.00657

Table 4. Switching PID gain

Error Boundary	Kp	Ki	Kd
$ E(t) < 20$	238.346	0.312577	28.3408
$20 \leq E(t) < 40$	83.4081	0.483074	13.2854
$40 \leq E(t) < 60$	21.06	0.759974	10.3165
$60 \leq E(t) < 80$	5.3621	0.0331574	0.639023
$ E(t) \geq 80$	83.5079	0.707829	8.56871



(a) General PID controller



(b) Switching PID controller

Fig. 5. Comparison between the general PID and the Switching PID

general PID controller are 153.744, 0, and 15.47, respectively. The gains of a general PID controller were optimized by using the GA. Nevertheless, there are some limit cycles as a result of the saturation effect. After using the switching algorithm, they are all removed, as shown in Fig 5(b).

Fig. 6 shows the influence of saturation after applying the suggested switching PID controller, which indicates only the rejection of internal nonlinearities. An output compensator is necessary to solve the external nonlinearities. An output compensator compares the angular position of a motor with the angular position of the external structure. Fig. 6 shows a comparison of the results using an output compensator and not using an output compensator.

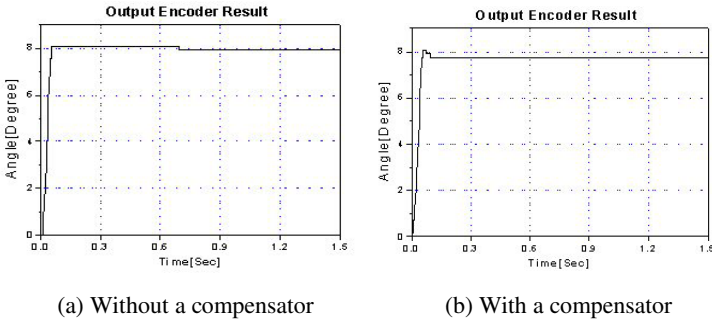


Fig. 6. Comparison of the results with and without a compensator

As known in Fig. 6, when an output compensator is used as a nonlinear controller, the influence of external nonlinearities can be dramatically reduced. Without the compensator, the steady state error was 4.07%, while the steady state error was 1.7% using the output compensator.

5 Conclusion

In this paper, two types of GA were applied to the motor system with nonlinearities. The first was used for obtaining the motor parameters in order to identify the motor system. The second was employed to obtain the optimal switching PID gains. This switching PID and output compensator could efficiently remove the internal and external nonlinearities such as the saturation, backlash, and some friction. In order to reduce these nonlinearities, an accurate motor model was derived using the improved GA, and the switching PID gains were determined by GA without any tedious transformations. In addition, a simple compensator was suggested to reduce the effect of external nonlinear elements. Through these processes, the limit cycle caused by the nonlinearities could be efficiently reduced or eliminated. As a result, the suggested switching PID controller was very efficient to control the motor system with nonlinearities. This algorithm was verified by simulations for the case of the step responses and a real experiment through the controller implementation.

In future, more advanced genetic algorithms will be considered. The current algorithm has a limitation when used in a real time system as a result of its computational

time. In order to apply a genetic algorithm to a real time system, the computational mechanism has to be changed into a simpler form. Furthermore, more advanced algorithms for multi-nonlinear processes will also be needed to provide a more accurate and faster control operation. A fuzzy algorithm will be a good candidate to compensate for the current switching PID controller.

Acknowledgement

This work was supported by Grant No. R01-2003-000-10364-0 from Korea Science & Engineering Foundation.

References

1. Nizar J. Ahmad and Farshad Khorrami: Adaptive Control of Systems with Backlash Hysteresis at the Input, Proceedings of the American Control Conference (1999), 3018-3022
2. Serkan T., Impram and Neil Munro: Limit Cycle Analysis of Uncertain Control Systems with Multiple Nonlinearities, Conf. on Decision and Control (2001), 3423-3428
3. E.J. Davison: Application of the Describing Function Technique in a Single-Loop System with Two Nonlinearities, IEEE Trans. On Automatic Control (1968), 168-170
4. Satoshi Komada, Noriyoshi Machii and Takamasa Hori: Control of Redundant Manipulators Considering Order of Disturbance Observer, IEEE Trans. On Industrial Electronics, Vol. 47, No. 2 (2000), 413-420
5. K.T. Woo, Li-Xin Wang, F.L. Lewis, and Z.X. Li: A Fuzzy System Compensator for Backlash, IEEE Int. Conf. on Robotics and Automation (1998), 181-186
6. C. W. Tao: Fuzzy Control for Linear Plants with Uncertain Output Backlashes, IEEE Trans. On systems, Man and Cybernetics, Vol. 32, No. 3 (2002), 373-380
7. Ali A. Jamshidifar and Caro Lucas: Genetic Algorithm Based Fuzzy Controller for Nonlinear Systems, Int. Conf. on Intelligent Systems (2004), 43-47
8. Vachkov, G. and Fukuda, T.: Identification and Control of Dynamical Systems Based on Cause-effect Fuzzy Models, Int. Conf. IFSA Vol. 4 (2001) 2072-2077
9. Akramizadeh, A., Farjami, A.A. and Khaloozadeh, H.: Nonlinear Hammerstein Model Identification Using Genetic Algorithm, Int. Conf. ICAIS (2002) 351-356

Preference Bi-objective Evolutionary Algorithm for Constrained Optimization*

Yuping Wang¹, Dalian Liu², and Yiu-Ming Cheung³

¹ Faculty of Computer Science and Technology,
Xidian University, Xi'an 710071, China
ywang@xidian.edu.cn

² Department of Basic Course Teaching, Beijing Union University, Beijing, China
ldlluck@sina.com

³ Department of Computer Science,
Hong Kong Baptist University, Hong Kong, China
ymc@comp.hkbu.edu.hk

Abstract. In this paper, we propose a new constraint handling approach that transforms constrained optimization problem of any number of constraints into a two objective preference optimization problem. We design a new crossover operator based on uniform design methods ([8]), a new mutation operator using local search and preference, and a new selection operator based on the preference of the two objectives. The simulation results indicate the proposed algorithm is effective.

1 Introduction

In the last two decades, evolutionary algorithms (EAs) have received much attention regarding their potential as global optimization techniques. Many researchers have applied EAs to constrained optimization problems (e.g., [1]~[6]). Among these algorithms, the most common approaches adopted in the literature are penalty function methods. In these methods, a penalty term is added to the objective function, the penalty increasing with the degree of constraint violation (static penalty) or the degree of constraint violation and the generation number (dynamic penalty). However, penalty functions have some well-known limitations ([4], [7]), from which the most remarkable is the difficulty to define good penalty parameters. These penalty parameters are normally generated by trial and error, and may severely affect the performance of evolutionary algorithms.

Recently, some researchers have suggested the use of multiobjective optimization concepts to handle constraints in EAs (e.g., [2], [5], [6]). The main idea behind these algorithms is to transform each constraint as an objective function, and redefine the constrained optimization problem as a multiobjective optimization problem with $(m+1)$ objectives, where m is the total number of constraints.

* This work was supported by the National Natural Science Foundation of China (60374063), by the Research Grant Council of Hong Kong SAR under Project HKBU 2156/04E.

Then use Pareto dominance as a selection criterion (e.g., [2], [5]), or use Pareto ranking to assign fitness in such a way that nondominated individuals are assigned a higher fitness value (e.g., [1]), or split the population in subpopulations that are evaluated either by objective function or by a single constraint function (e.g., [6]). Although there are some drawbacks such as they can not produce competitive results with respect to the state-of-art constraint handling techniques in evolutionary optimization, some results are still encouraging.

In this paper, we propose a new constraint handling approach that transforms constrained optimization problem of any number of constraints into a two objective preference optimization problem. We design a new crossover operator using uniform design methods ([8]), and a new selection operator based on the preference of the two objectives. The simulation results indicate the proposed algorithm is effective.

2 Transformation of Constrained Optimization Problem

We consider the following constrained optimization problem

$$\begin{cases} \min f(x) \\ \text{s.t. } g_i(x) \leq 0, i = 1 \sim m \end{cases} \quad (1)$$

Where $x \in R^n$. Note that the equality constraint, for example, $h(x) = 0$, is equivalent to the inequality constraint $h(x)^2 \leq 0$, so any problem with equality constraints can be transformed into the form of problem (1). If a point satisfies all constraints of problem (1), then it is called a feasible solution. The set

$$X = \{x \mid g_i(x) \leq 0, i = 1 \sim m\}$$

is called the set of feasible solutions of problem (1) (simply the feasible set).

Define functions

$$f_1(x) = f(x) \quad (2)$$

and

$$f_2(x) = \max\{0, g_i(x), i = 1 \sim m\} \quad (3)$$

Obviously, for any $x \in R^n$, we have $f_2(x) \geq 0$, and $f_2(x) = 0$ if and only if $x \in X$. Thus $f_2(x)$ attains the minimum value at x if and only if $x \in X$. Therefore, problem (1) can be transformed into the following two objective optimization problem

$$\min\{f_1(x), f_2(x)\} \quad (4)$$

Optimizing $f_1(x)$ means searching optimal solution for objective function for problem (1), and optimizing $f_2(x)$ means searching for the feasible solution for problem (1). Thus simultaneously optimizing both $f_1(x)$ and $f_2(x)$ means looking for not only a feasible solution for problem (1), but also a solution minimizing the objective function of problem (1), i.e., an optimal solution of problem (1).

However, due to the specific structure of problem (4), the evolution process for it in the proposed algorithm is different from that for usual multiobjective

optimization problems. For different solutions (feasible or infeasible solutions for problem (1)), we adopt different evolution schemes. For feasible solution, the key issue in evolution is put on to optimize the first objective function for the transformed problem (i.e., objective function for problem (1)) so that we can get the optimal solution for problem (1) as soon as possible. In this case, we will use some evolution scheme to improve the first objective function $f_2(x)$. While for infeasible solution, the key issue is to optimize the second objective function for the transformed problem (i.e., constraint functions for problem (1)) so that we can get feasible solution for problem (1) as soon as possible. In this case, we will use some evolution scheme to improve the second objective function $f_2(x)$.

In order to illustrate the relationship of problem (1) and (4), we introduce the concept of Pareto optimal solution as follows.

Definition 1. For $x^* \in R^n$, if there exists no point $x \in R^n$ such that $f_i(x) \leq f_i(x^*)$ for $i = 1, 2$ and $f_j(x) < f_j(x^*)$ for some $j = 1$ or 2 , then x^* is called a Pareto optimal solution of problem (4).

Now, we have the following result.

Lemma 1. Suppose that there exists at least one optimal solution of problem (1), then x^* is an optimal solution of problem (1) if and only if it is a Pareto optimal solution of problem (4) and $x^* \in X$.

Proof. If x^* is an optimal solution of problem (1), then $x^* \in X$. Thus $f_2(x)$ attains its minimum value at x^* , and $f_1(x)$ attains the minimum value at x^* in X . Therefore, x^* is a Pareto optimal solution of problem (4).

If $x^* \in X$ is a Pareto optimal solution of problem (4), suppose that x^* is not an optimal solution of problem (1), then there exists some $x \in X$ such that $f_1(x) < f_1(x^*)$. Note that $f_2(x) = f_2(x^*) = 0$. This contradicts the fact that x^* is a Pareto optimal solution of problem (4). The proof is completed.

3 Genetic Operators

3.1 New Crossover Operator

We use the uniform design methods ([8]) to design a new crossover operator. This operator can effectively explore the search space near parents taken part in the crossover. The detail is as follows.

Denote

$$C_n = \{(x_1, x_2, \dots, x_n) \mid 0 \leq x_1, \dots, x_n \leq 1\} \tag{5}$$

$$B_n(a, \delta) = \{(x_1, x_2, \dots, x_n) \mid (x_1 - a_1)^2 + \dots + (x_n - a_n)^2 \leq \delta^2\} \tag{6}$$

where $a = (a_1, a_2, \dots, a_n) \in R^n$ and δ is a real number. Let $[\theta]$ denote the integer part of a real number θ , and $\{\theta\}$ its decimal part. In the following, we will explain how to generate a set of uniformly distributed points in $B_n(a, \delta)$. Three widely used methods to generate q uniformly distributed points in C_n are as follows ([8]), where q is a positive integer.

- Denote $(\gamma_1, \gamma_2, \dots, \gamma_n) = (\sqrt{p_1}, \sqrt{p_2}, \dots, \sqrt{p_n})$, where p_1, \dots, p_n are the first n positive primes. Then

$$\{(\{k\gamma_1\}, \dots, \{k\gamma_n\}) \mid k = 1 \sim q\} \tag{7}$$

is a set of q uniformly distributed points in C_n .

- Let p be a positive prime. Denote $\lambda = 1/(n + 1)$ and $(\gamma_1, \gamma_2, \dots, \gamma_n) = (p^\lambda, p^{2\lambda}, \dots, p^{n\lambda})$, then

$$\{(\{k\gamma_1\}, \dots, \{k\gamma_n\}) \mid k = 1 \sim q\} \tag{8}$$

is a set of q uniformly distributed points in C_n .

- Let p be a positive prime satisfying $p \geq 2n + 3$. Denote $\omega = 2\pi/p$ and $(\gamma_1, \gamma_2, \dots, \gamma_n) = 2(\cos(\omega), \cos(2\omega), \dots, \cos(n\omega))$, then

$$\{(\{k\gamma_1\}, \dots, \{k\gamma_n\}) \mid k = 1 \sim q\} \tag{9}$$

is a set of q uniformly distributed points in C_n .

Now we design a simple method to generate r approximately uniformly distributed points in $B_n(a, \delta)$, where r is a positive integer with $r < q$.

Algorithm 1: (Generate uniformly distributed points in $B_n(a, \delta)$)

1. Generate q uniformly distributed points in C_n by one of formulas (7), (8) and (9), and denote the set of these points by

$$\{(c_{k1}, \dots, c_{kn}) \mid k = 1 \sim q\} = \{(\{k\gamma_1\}, \dots, \{k\gamma_n\}) \mid k = 1 \sim q\}$$

In simulation we use formulation (7).

2. Generate q uniformly distributed points in *hypercube*

$$[a - \delta, a + \delta] = \{(x_1, \dots, x_n) \mid a_i - \delta \leq x_i \leq a_i + \delta, i = 1 \sim n\},$$

which contains $B_n(a, \delta)$, by

$$B = \{(b_{k1}, \dots, b_{kn}) \mid b_{kj} = (a_j - \delta) + 2\delta c_{kj}, j = 1 \sim n, k = 1 \sim q\}$$

3. Generate r approximately uniformly distributed points in $B_n(a, \delta)$ by

$$P = \{(b_{k1}, \dots, b_{kn}) \mid (b_{k1}, \dots, b_{kn}) \in B \cap B_n(a, \delta)\}$$

where r is the number of elements in P .

Algorithm 2: (Crossover operator)

1. Suppose that x and y are two parents taken part in crossover. Let $a = (x + y)/2$, $\delta = \sqrt{\sum_{i=1}^n (a_i - x_i)^2}$. Define $B_n(a, \delta)$ by formula (6).
2. Determine the set P by algorithm 1. Then the points in P are offspring of x and y .

3.2 Mutation Operator with Preference

Suppose that x is a solution undergoing mutation. There are two cases to generate its offspring \bar{x} .

Case 1. If $f_2(x) = 0$, then $\bar{x} = x + \Delta x$, where

$$\Delta x \sim N(O, \sigma^2) = (N(0, \sigma_1^2), \dots, N(0, \sigma_n^2)),$$

i.e., Δx is an n -dimensional random variable obeying n -dimensional Gaussian distribution with mean $O = (0, 0, \dots, 0)$ and variance $\sigma^2 = (\sigma_1^2, \dots, \sigma_n^2)$.

Case 2. If $f_2(x) > 0$, define an approximate gradients of $f_2(x)$ at x , denoted by $d = (d_1, d_2, \dots, d_n)$, by

$$d_i = [f_2(x + \alpha e_i) - f_2(x)]/\alpha, \quad i = 1 \sim n,$$

where e_i is an n -dimensional unit vector with its i -th element one and other elements zeros, and $\alpha > 0$ is a small constant. Then \bar{x} is obtained by a line search method ([9]) starting at x along the search direction $-d$.

In case 2, the mutation will force x to approach to a feasible solution. In this sense, the mutation will prefer to evolve x to enter the feasible region.

3.3 Selection Operator with Preference

The selection in the proposed algorithm does not use the Pareto ranking or the concept of Pareto dominance. Instead, we use the following selection scheme.

- If two solutions have the same second objective values, we prefer to select the one with the smaller first objective value.
- If the second objective value of one solution is zero, and that of the other is nonzero, we prefer to choose the one with the zero second objective value.
- If the second objective values of two solutions are both nonzero, we prefer to select the one with the smaller second objective value.

4 Proposed Algorithm

Algorithm 3: (Preference bi-objective evolutionary algorithm)

1. (Initialization) Given population size N . Generate initial population $P(0)$ randomly, given crossover and mutation probabilities p_c and p_m . Set $t = 0$.
2. (Crossover) Select parents from $P(t)$ for crossover with probability p_c . Randomly match every two parents. For each pair of parents, generate r offspring by crossover operator (algorithm 2). The set of all offspring generated is denoted as $O1$.
3. (Mutation) Select parents from $O1$ for mutation with probability p_m . Each selected parent generates an offspring by mutation operator. The set of all these offspring is denoted as $O2$.
4. (Selection) Select the next population $P(t + 1)$ among $P(t) \cup O1 \cup O2$ by selection operator. Let $t = t + 1$.
5. (Stop criterion) If stop criterion is satisfied, then stop; otherwise, go to Step 2.

5 Simulation Results

Six widely used test functions are chosen from ([2]) and the proposed algorithm (denoted as PBEA for short) is executed for these functions.

F01:

$$\begin{cases} \max f(x) = \left| (\sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i)) / \sqrt{\sum_{i=1}^n i x_i^2} \right| \\ \text{s.t.} & 0.75 - \prod_{i=1}^n x_i \leq 0, \sum_{i=1}^n x_i - 0.75n \leq 0 \\ & 0 \leq x_i \leq 10 (i = 1 \sim n), n = 20 \end{cases}$$

F02:

$$\begin{cases} \min f(x) = (x_1 - 10)^3 + (x_2 - 20)^3 \\ \text{s.t.} & -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0, 13 \leq x_1 \leq 100, 0 \leq x_2 \leq 100 \\ & (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0 \end{cases}$$

F03:

$$\begin{cases} \min f(x) = [\sin^3(2\pi x_1) \sin(2\pi x_2)] / [x_1^3(x_1 + x_2)] \\ \text{s.t.} & x_1^2 - x_2 + 1 \leq 0, 1 - x_1 + (x_2 - 4)^2 \leq 0 \\ & 0 \leq x_i \leq 10, i = 1, 2 \end{cases}$$

F04:

$$\begin{cases} \min f(x) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 \\ & + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \\ \text{s.t.} & -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0 \\ & -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0 \\ & -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0 \\ & 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0 \\ & -10 \leq x_i \leq 10, i = 1 \sim 7 \end{cases}$$

F05:

$$\begin{cases} \min f(x) = x_1^2 + (x_2 - 1)^2 \\ \text{s.t.} & x_1^2 - x_2 = 0, -1 \leq x_i \leq 1, i = 1, 2 \end{cases}$$

F06:

$$\begin{cases} \min f(x) = e^{x_1 x_2 x_3 x_4 x_5} \\ \text{s.t.} & \sum_{i=1}^5 x_i^2 = 10, x_2 x_3 - 5x_4 x_5 = 0 \\ & -2.3 \leq x_1, x_2 \leq 2.3, -3.2 \leq x_3, x_4, x_5 \leq 3.2 \end{cases}$$

In simulation, we take the following parameters: $N = 100$, $p_c = 0.8$, $p_m = 0.1$, $q = 5$. We execute the proposed algorithm (PBEA) 30 independent runs for each test function and record the following data: Best objective value (Best), mean best objective value (Mean) and the worst objective value (Worst) on 30 runs. For each run the algorithm 1 stops if the best solution found in 20 successive generations can not be improved. We compare the results with the existing ones obtained in [2] by algorithms in [2] (IS-PAES), [5] (CHEA), [6] (TCEA) and [10](COMOGA) in Table 1. It can be seen from Table 1 that for each test function the performance of PBEA is better than that of any of CHEA, TCEA and COMOGA. For $F04$ and $F05$, the performance of PBEA is a little bit

Table 1. Comparison of best, mean and worst solutions among PBEA and those in [2], [5], [6] and [10]. N.F represents not feasible solution.

TF	Methods	Best	Mean	Worst	Optimal
F01	CHEA	0.68087	0.58474	0.4992	0.803619
	TCEA	0.000212	0.000077	0.000008	
	COMOGA	0.02172	0.01641	0.00781	
	IS-PAES	0.803376	0.793281	0.768291	
	PBEA	0.8036121	0.7986843	0.7859865	
F02	CHEA	-6957.9506	-6903.7747	-6845.4321	-6961.814
	TCEA	-6941.9321	-6873.1396	-6743.4951	
	COMOGA	-6622.2803	-6058.8651	-4859.3310	
	IS-PAES	-6961.814	-6961.813	-6961.810	
	PBEA	-6961.81396	-6961.81394	-6961.81389	
F03	CHEA	0.095825	0.095825	0.095825	0.095825
	TCEA	0.095825	0.095825	0.095825	
	COMOGA	0.095813	0.095447	0.093345	
	IS-PAES	0.095825	0.095825	0.095825	
	PBEA	0.09582507	0.09582506	0.09582504	
F04	CHEA	681.3240	681.1016	700.5506	680.630
	TCEA	693.2523	717.0234	744.0844	
	COMOGA	723.8549	873.9179	995.9818	
	IS-PAES	680.630	680.631	680.634	
	PBEA	680.632431	680.663242	680.702575	
F05	CHEA	0.7490	0.7491	0.7493	0.750
	TCEA	0.74904	0.7601	0.8119	
	COMOGA	0.7490	0.7493	0.7499	
	IS-PAES	0.750	0.750	0.751	
	PBEA	0.74990001	0.75262803	0.75354421	
F06	CHEA	N.F	N.F	N.F	0.053950
	TCEA	N.F	N.F	N.F	
	COMOGA	N.F	N.F	N.F	
	IS-PAES	0.05517	0.28184	0.5471	
	PBEA	0.05395	0.19632	0.35847	

worse than that of IS-PAES, but for other four test functions, the performance of PBEA is better than that of IS-PAES. Also note that the best and mean solutions found by PBEA for all test functions are very close to the true optimal solutions. It indicates PBEA is effective.

References

1. Carlos A. Coello Coello.: A Simple Multi-Membered Evolution Strategy to Solve Constrained Optimization Problems. *IEEE Trans. Evolutionary Computation* **9** (2005) 1-17
2. A.H. Aguirre, S.B. Rionda, Carlos A. Coello Coello, et al.: Handling Constraints Using Multiobjective Optimization Concepts. *International J. for Numerical Methods in Engineering* **59** (2004) 1989-2017

3. R. Farmani, J.A. Wright.: Self-Adaptive Fitness Formulation for Constrained Optimization. *IEEE Trans. Evolutionary Computation* **7** (2003) 445-455
4. M.J. Tahk, B.C. Sun.: Coevolutionary Augmented Lagrangian Methods for Constrained Optimization. *IEEE Trans. Evol. Comput* **4** (2000) 114-124.
5. Carlos A. Coello Coello.: Constraint Handling Using An Evolutionary Multiobjective Optimization Technique. *Civil Engineering Systems* **17** (2000) 319-346
6. Carlos A. Coello Coello.: Treating Constraints as Objectives for Single Objective Evolutionary Optimization. *Engineering Optimization* **32** (2000) 275-308
7. Z. Michalewicz, M. Schoenauer.: Evolutionary Algorithms for Constrained Parameter Optimization Problems. *Evolutionary Computation* **4** (1996) 1-32
8. F.T.Fang, Y.Wang. *Number-theoretic methods in statistics*. London, UK, Chapman & Hall (1994)
9. R. Fletcher. *Practical Methods of Optimization*. John Wiley and Sons, Chichester (1987)
10. Patrick D. Surry, Nicholas J. Radcliffe.: The COMOGA Method: Constrained Optimization by Multiobjective Genetic Algorithms. *Control and Cybernetics* **26** (1997) 391-412

Self-adaptive Differential Evolution

Mahamed G.H. Omran¹, Ayed Salman², and Andries P. Engelbrecht³

¹ Faculty of Computing & IT, Arab Open University, Kuwait
mjomran@engineer.com

² Computer Engineering Department, Kuwait University, Kuwait
ayed@eng.kuniv.edu.kw

³ Department of Computer Science, University of Pretoria, Pretoria, South Africa
engel@cs.up.ac.za

Abstract. Differential Evolution (DE) is generally considered as a reliable, accurate, robust and fast optimization technique. DE has been successfully applied to solve a wide range of numerical optimization problems. However, the user is required to set the values of the control parameters of DE for each problem. Such parameter tuning is a time consuming task. In this paper, a self-adaptive DE (SDE) is proposed where parameter tuning is not required. The performance of SDE is investigated and compared with other versions of DE. The experiments conducted show that SDE outperformed the other DE versions in all the benchmark functions.

1 Introduction

Evolutionary algorithms (EAs) are general-purpose stochastic search methods simulating natural selection and evolution in the biological world. EAs differ from other optimization methods, such as Hill-Climbing [1] and Simulated Annealing [2], in the fact that EAs maintain a population of potential (or candidate) solutions to a problem, and not just one solution [3].

Due to its population-based nature, EAs can avoid being trapped in a local optimum and consequently can often find global optimal solutions. Thus, EAs can be viewed as global optimization algorithms. However, it should be noted that EAs may fail to converge to a global optimum [4].

Recently, Storn and Price [5] proposed a new EA called Differential Evolution (DE). DE is similar to Genetic Algorithms (GAs) [6]. The main difference between GAs and DE is that, in GAs, mutation is the result of small perturbations to the genes while in DE mutation is caused by arithmetic combinations of individuals [6]. At the beginning of the evolution, the mutation operator of DE favors exploration. Then as the evolution progresses the mutation operator favors exploitation. Hence, DE automatically adapts the mutation increments (i.e. search step) to the correct value during the evolution process. Another difference between GAs and DE is that simple GAs use a binary representation while DE uses a floating-point representation [7].

DE is easy to implement, requires little parameter tuning [8] and exhibits fast convergence [9]. However, according to Krink *et al.* [10], noise may adversely affect the performance of DE due to its greedy nature.

DE has been successfully applied to solve a wide range of optimization problems. DE is now generally considered as a reliable, accurate, robust and fast optimization technique. However, the user needs to find the best values for the control parameters of DE for each problem. Finding the best values for the control parameters for each problem is a time consuming task. This paper proposes a new version of DE where the control parameters are self-adaptive. The new version is called *Self-adaptive Differential Evolution* (SDE). The results of the experiments conducted are shown and compared with the versions of DE proposed by Price and Storn [11].

The remainder of the paper is organized as follows: Section 2 provides an overview of DE. The proposed version of DE is given in Section 3. Benchmark functions to measure the performance of DE are provided in Section 4. Results of the experiments are presented in Section 5. Finally, Section 6 concludes the paper.

2 Differential Evolution

Differential evolution does not make use of a mutation operator that depends on some probability distribution function, but introduces a new arithmetic operator which depends on the differences between randomly selected pairs of individuals.

For each parent, $\mathbf{x}_i(t)$, of generation t , an offspring, $\mathbf{x}'_i(t)$, is created in the following way: Randomly select three individuals from the current population, namely $\mathbf{x}_{i_1}(t)$, $\mathbf{x}_{i_2}(t)$ and $\mathbf{x}_{i_3}(t)$, with $i_1 \neq i_2 \neq i_3 \neq i$ and $i_1, i_2, i_3 \sim U(1, \dots, s)$, where s is the population size. Select a random number $r \sim U(1, \dots, N_d)$, where N_d is the number of genes (parameters) of a single chromosome. Then, for all parameters $j = 1, \dots, N_d$, if $U(0, 1) < P_r$, or if $j = r$, let

$$\mathbf{x}'_{i,j}(t) = x_{i_3,j}(t) + F(x_{i_1,j}(t) - x_{i_2,j}(t)) \quad (1)$$

otherwise, let

$$\mathbf{x}'_{i,j}(t) = x_{i,j}(t) \quad (2)$$

In the above, P_r is the probability of reproduction (with $P_r \in [0, 1]$), F is a scaling factor with $F \in (0, \infty)$, and $\mathbf{x}'_{i,j}(t)$ and $x_{i,j}(t)$ indicate respectively the j -th parameter of the offspring and the parent.

Thus, each offspring consists of a linear combination of three randomly chosen individuals when $U(0, 1) < P_r$; otherwise the offspring inherits directly from the parent. Even when $P_r = 0$, at least one of the parameters of the offspring will differ from the parent (forced by the condition $j = r$).

The mutation process above requires that the population consists of more than three individuals.

After completion of the mutation process, the next step is to select the new generation. For each parent of the current population, the parent is replaced with its offspring if the fitness of the offspring is better, otherwise the parent is carried over to the next generation.

Price and Storn [11] proposed ten different strategies for DE based on the individual being perturbed (i.e. $\mathbf{x}_{i3,j}(t)$), number of individuals used in the mutation process and the type of crossover used. The strategy shown in this section is known as DE/rand/1. This strategy is considered to be the most widely used and the most successful strategy [7].

The performance of DE is sensitive to the choice of control parameters [5, 12, 13, 14]. Recently, there were several attempts to control the parameters of DE. Liu and Lampinen [13] proposed a fuzzy DE where the values of the control parameters (i.e. F and P_r) are adapted using fuzzy logic. Human knowledge and previous experience are used to establish the fuzzy rules and membership functions. This is not very objective and depends on how good the knowledge of the expert is.

Abbas [15] proposed the Self-adaptive Pareto DE (SPDE), a self-adaptive approach to DE for multi-objective optimization problems. In SPDE, the parameter F is generated for each variable from a normal distribution, $N(0,1)$. Each individual, i , has its own probability of reproduction, P_i . The parameter P_i is first initialized for each individual in the population from a uniform distribution between 0 and 1. Then, P_r is adapted according to the following equation:

$$P_i(t) = P_{i1}(t) + N(0,1) \times (P_{i2}(t) - P_{i3}(t)) \quad (3)$$

where $i1, i2, i3 \sim U(1, \dots, s)$.

According to Abbas [15], the proposed approach performed well compared to other evolutionary multi-objective optimization approaches.

Zaharie [12] theoretically studied the behavior of DE and proposed an approach of adapting the control parameters of DE that is guided by the evolution of population diversity.

More recently, Bui *et al.* [16] proposed to use DE/rand/1 that generates F from a uniform distribution between 0 and 1. According to Bui *et al.* [16], this approach performed better than the conventional DE/rand/1 using a fixed value of F .

3 Self-adaptive Differential Evolution (SDE)

Similar to SPDE, SDE uses self-adaptation to adapt the control parameters of DE. However, SDE self-adapts F , contrary to SPDE which uses a normal distribution $N(0,1)$ for F . Furthermore, in SDE, P_r is generated for each individual from a normal distribution while SPDE self-adapts P_r according to equation (3). In addition, SPDE uses a normal distribution with mean zero and standard deviation one (i.e. $N(0,1)$). On the other hand, SDE uses normal distributions of different means and standard deviations.

SDE works as follows: For each parent, $\mathbf{x}_i(t)$, of generation t , an offspring, $\mathbf{x}'_i(t)$, is created in the following way: Randomly select three individuals from the current population, namely $\mathbf{x}_{i1}(t)$, $\mathbf{x}_{i2}(t)$ and $\mathbf{x}_{i3}(t)$, with $i1 \neq i2 \neq i3 \neq i$ and $i1, i2, i3 \sim U(1, \dots, s)$, where s is the population size. Select a random number $r \sim U(1, \dots, N_d)$, where N_d is the number of genes (parameters) of a single chromosome. Then, for all parameters $j = 1, \dots, N_d$, if $U(0, 1) < N(0.5, 0.15)$, or if $j = r$, let

$$x'_{i,j}(t) = x_{i3,j}(t) + F_i(t)(x_{i1,j}(t) - x_{i2,j}(t)) \quad (4)$$

otherwise, let

$$x'_{i,j}(t) = x_{i,j}(t)$$

where

$$F_i(t) = F_{i4}(t) + N(0,0.5) \times (F_{i5}(t) - F_{i6}(t)) \quad (5)$$

with $i4, i5, i6 \sim U(1, \dots, s)$.

Thus, each individual i has its own scaling factor F_i which is a function for the scaling factor of randomly selected individuals. The parameter F_i is first initialized for each individual in the population from a normal distribution $N(0.5, 0.15)$ generating values which fits well within the range $(0, 1]$.

The rationale behind using a normal distribution $N(0.5, 0.15)$ for P_r is that $N(0.5, 0.15)$ will generate values in the range of $[0.5 - 3 \times 0.15, 0.5 + 3 \times 0.15]$ which covers the P_r boundary giving more probability to values surrounding 0.5. The reason for preferring values surrounding 0.5 is that $P_r = 0.5$ represents a uniform crossover (i.e. there is an equal probability that the new offspring will be chosen from either the perturbed individual or from the original individual). Hence, using $N(0.5, 0.15)$ will provide a relatively fair chance for both the perturbed individual and the original individual to be selected as the new offspring. According to our preliminary experiments (not shown because of space limit), using a normal distribution $N(0.5, 0.15)$ for P_r generally performed better than using equation (3) on the tested functions.

The normal distribution $N(0, 0.5)$ in equation (5) is used instead of $N(0, 1)$ on the basis of the assumption that $N(0, 0.5)$ will generate values in the range of $[-1.5, 1.5]$ which covers the F boundary (remember that F is usually $\in [0.5, 1]$) better than using $N(0, 1)$. $N(0, 1)$ will generate values in the range $[-3, 3]$. Hence, $N(0, 1)$ may increase the probability of generating values of F outside the range $(0, 1]$. If the value of F_i is not within $(0, 1]$, F_i will be repaired according to the repair rule used by Abbas [15]. The repair rule will truncate the constant part of the value (e.g. the value 1.4 will be adjusted to 0.4).

4 Benchmark Functions

The benchmark functions in this section provide a balance of unimodal and multimodal functions. These standard test functions, taken from the literature of evolutionary computation, have been used in various DE studies [6, 10].

For each of these functions, the goal is to find the global minimizer. Formally speaking:

Given $f: \mathfrak{R}^{N_d} \rightarrow \mathfrak{R}$

find $\mathbf{x}^* \in \mathfrak{R}^{N_d}$ such that $f(\mathbf{x}^*) \leq f(\mathbf{x}), \forall \mathbf{x} \in \mathfrak{R}^{N_d}$

Spherical: $\mathbf{x}^* = \mathbf{0}$, with $f(\mathbf{x}^*) = 0$.

$$f(\mathbf{x}) = \sum_{i=1}^{N_d} x_i^2 \quad (6)$$

with $-100 \leq x_i \leq 100$.

Rosenbrock: $\mathbf{x}^* = (1,1,\dots,1)$, with $f(\mathbf{x}^*) = 0$.

$$f(\mathbf{x}) = \sum_{i=1}^{N_d/2} \left(100(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2 \right) \quad (7)$$

with $-2.048 \leq x_i \leq 2.048$.

Rastrigin: $\mathbf{x}^* = \mathbf{0}$, with $f(\mathbf{x}^*) = 0$.

$$f(\mathbf{x}) = \sum_{i=1}^{N_d} \left(x_i^2 - 10 \cos(2\pi x_i) + 10 \right) \quad (8)$$

with $-5.12 \leq x_i \leq 5.12$.

Griewank: $\mathbf{x}^* = \mathbf{0}$, with $f(\mathbf{x}^*) = 0$.

$$f(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^{N_d} x_i^2 - \prod_{i=1}^{N_d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (9)$$

with $-600 \leq x_i \leq 600$.

Spherical and Rosenbrock are unimodal while Rastrigin and Griewank are multimodal functions.

5 Experimental Results

In this section, SDE is compared with the other strategies of DE proposed by Price and Storn [11]. The control parameters for these strategies were set as follows: $F = 0.5$ and $P_r = 0.9$. No attempt was made to tune the DE parameters to each problem. The rationale behind this decision is the fact that in real-world applications the evaluation time is significant and as such parameter tuning is usually a time consuming process [10]. For all the algorithms used in this section $s = 40$. Furthermore, all functions were implemented in 50 dimensions.

The results reported in this section are averages and standard deviations over 30 simulations. Each simulation was allowed to run for 20 000 evaluations of the objective function.

Table 1 shows the results of the experiments. Examining the results, SDE significantly outperformed the other strategies in all the test functions. This shows the

efficiency of the SDE scheme in the unimodal and multimodal functions. The improvement is even more significant for the Rastrigin function. The Rastrigin function has many good local optima. Hence, it is known to be a difficult function to optimize. SDE performed very well in this function suggesting that SDE works well with difficult multimodal functions. Furthermore, examining the standard deviations of all the algorithms, it can be seen that SDE has the smallest standard deviation. Thus, one can conclude that SDE is more stable and thus more robust than the other versions of DE.

Figures 1, 2, 3 and 4 show the DE/best/1 and DE/rand-to-best/1 getting trapped in a local optimum very early during the search. This behavior is expected because of the low diversity in both strategies. Figures 1, 2 and 4 show that SDE has a faster convergence rate than DE/rand/2. On the other hand, the three figures show that DE/rand/1 and DE/best/2 converge faster than SDE. However, SDE converges to a better solution than both DE/rand/1 and DE/best/2. For the Rastrigin function shown in Figure 3, it can be observed that SDE converges to a better solution faster than DE/rand/1, DE/rand/2 and DE/best/2.

Table 1. Mean and standard deviation (\pm SD) of the function optimization results ($N_d=50$)

	Sphere	Rosenbrock	Rastrigin	Griewank
DE/rand/1	0.004225 \pm 0.008481	135.352596 \pm 41.405378	367.286133 \pm 28.252943	0.810391 \pm 0.491085
DE/best/1	9.10373 \pm 2.299977	20704.1313 \pm 10067.8147	291.457902 \pm 38.724004	213.568685 \pm 53.005959
DE/best/2	0.000767 \pm 0.001640	165.565743 \pm 58.467422	363.211698 \pm 125.392344	0.33766 \pm 0.228069
DE/rand/2	0.038096 \pm 0.022325	191.239558 \pm 55.30455	405.489141 \pm 18.75283	1.540165 \pm 0.184424
DE/rand-to-best/1	4.294976 \pm 0.797361	6016.779462 \pm 2551.570461	162.186077 \pm 28.675567	94.336715 \pm 20.46955
SDE	0.000034\pm 0.000059	108.599572\pm 37.453761	36.808762\pm 8.520613	0.070461\pm 0.067611

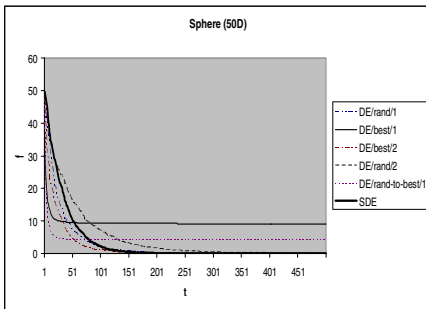


Fig. 1. DE strategies performance for the Sphere (50D)

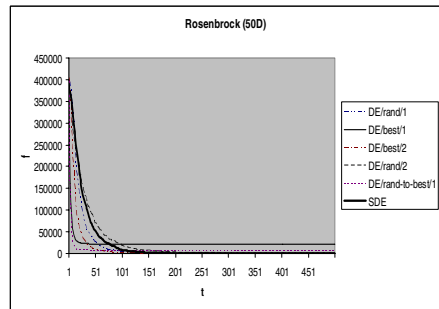


Fig. 2. DE strategies performance for the Rosenbrock (50D)

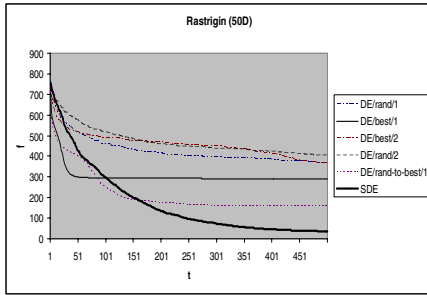


Fig. 3. DE strategies performance for the Rastrigin (50D)

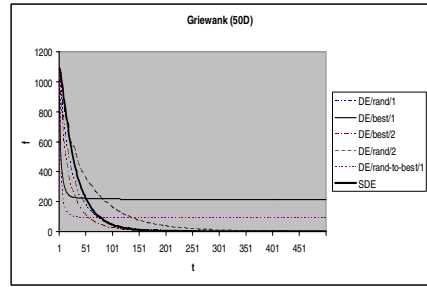


Fig. 4. DE strategies performance for the Griewank (50D)

6 Conclusion

The performance of DE is sensitive to the choice of control parameters. Finding the best values for these parameters for each problem is a time consuming task. This paper proposes a self-adaptive version of DE, called SDE. The approach was tested on four benchmark functions where it outperformed other well-known versions of DE.

Future research will investigate the contribution of each parameter in SDE. Furthermore, SDE will be tested on more functions and compared with other optimization techniques (e.g. SPDE, ES and EP). In addition, the effect of noise on the performance of SDE will be explored. The effect of the population size and the problem dimension will also be investigated.

References

1. Michalewicz, Z., Fogel, D.: *How to Solve It: Modern Heuristics*. Springer-Verlag, Berlin (2000)
2. Van Laarhoven, P., Aarts, E.: *Simulated Annealing: Theory and Applications*. Kluwer Academic Publishers (1987)
3. Engelbrecht, A.: *Computational Intelligence: An Introduction*. John Wiley and Sons (2002)
4. Gray, P., Hart, W., Painton, L., Phillips, C., Trahan, M., Wagner, J.: *A Survey of Global Optimization Methods*. Sandia National Laboratories, 1997, <http://www.cs.sandia.gov/opt/survey> (visited 2 July 2004) (1997)
5. Storn, R., Price, K.: *Differential Evolution – A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces*. Technical Report TR-95-012, International Computer Science Institute, Berkeley, CA (1995)
6. Feoktistov, V., Janaqi, S.: *Generalization of the Strategies in Differential Evolution*. In: the Proceedings of the 18th International Parallel and Distributed Processing Symposium (2004) 165-170
7. Babu, B., Jehan, M.: *Differential Evolution for Multi-Objective Optimization*. In: the 2003 Congress on Evolutionary Computation, Vol. 4 (2003) 2696-2703

8. Paterlini, S., Krink, T.: High Performance Clustering with Differential Evolution. In: the 2004 Congress on Evolutionary Computation, Vol. 2 (2004) 2004-2011
9. Karaboga, D., Okdem, S.: A Simple and Global Optimization Algorithm for Engineering Problems: Differential Evolution Algorithm. *Turk Journal of Electrical Engineering*, Vol. 12, No. 1 (2004) 53-60
10. Krink, T., Filipic, B., Fogel, G.: Noisy Optimization Problems – A Particular Challenge for Differential Evolution. In: the 2004 Congress on Evolutionary Computation (2004) 332-339
11. Price, K., Storn, R.: DE Web site, <http://www.ICSI.Berkeley.edu/~storn/code.html> (visited 9 July 2005) (2005)
12. Zaharie, D.: Parameter Adaptation in Differential Evolution by Controlling the Population Diversity. In: the 4th International Workshop on Symbolic and Numeric Algorithms for Scientific Computing, *Analele Univ. Timisoara*, Vol. XXXX, Issue 2 (2002)
13. Liu, J., Lampinen, J.: A Fuzzy Adaptive Differential Revolution Algorithm. In: Proceedings of the IEEE TENCON'02 (2002) 606-611
14. Wei, C., He, Z., Zhang, Y., Pei, W.: Swarm Directions Embedded in Fast Evolutionary Programming. In: Proceedings of the 2002 Congress on Evolutionary Computation (2002) 1278-1283
15. Abbas, H.: The Self-Adaptive Pareto Differential Evolution Algorithm. In: Proceedings of the 2002 Congress on Evolutionary Computation (2002) 831-836
16. Bui, L., Shan, Y., Qi, F., Abbas, H.: Comparing Two Versions of Differential Evolution in Real Parameter Optimization. Technical Report TR-ALAR-200504009, The Artificial Life and Adaptive Robotics Laboratory, School of IT and Electrical Engineering, University of New South Wales, Canberra, Australia (2005)

Steady-State Evolutionary Algorithm for Multimodal Function Global Optimization

Ziyi Chen and Lishan Kang

State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, China
chenziyi@263.net, kang_wuhu@yahoo.com

Abstract. This paper presents a two-phase steady-state evolutionary algorithm (TSEA) for solving function optimization containing multiple global optima. The algorithm includes two phases: firstly, steady-state evolution algorithm is used to get sub-optimal solutions in the global search, it enables individual to draw closer to each optimal solution, thus population is divided into subpopulations automatically after the global search. Secondly, local search is carried in the neighborhood of the best individual of each subpopulation to obtain precise solutions. Comparing with other algorithms, it has the following advantages. (1) It designs a new multi-parent crossover operator with strong direction which can accelerate the convergence. (2) A novel replacement strategy is proposed to maintain the diversity of population. This strategy is very simple and effective with little computational cost. (3) Proposed algorithm needs no additional control parameter which depends on a special problem. The experiment results show that TSEA is very efficient for the optimization of multi-modal functions.

1 Introduction

Interest in the multimodal function optimization is expanding rapidly since real-world optimization problems often require the location of multiple optima in the search space. DeJong (1975) in his doctoral dissertation used a crowding model to introduce diversity among solutions in a GA population [1]. The main idea of DeJong's study was the suggestion of replacing one solution by a similar solution in maintaining optimum solutions in an evolving population. Mahfoud improved standard crowding of De Jong by introducing competition between children and parents of identical niches [2]. After crossover and eventually mutation, each child replaces the nearest parent if it has a higher fitness. Other researchers have developed some different crowding schemes, for example, Restricted Tournament Selection (RTS) [3], adapts tournament selection for multimodal optimization, Mengshoel et al. [4] proposed *probabilistic crowding* as a probabilistic extension of the original *Deterministic crowding* (DC) method. Goldberg and Richardson proposed another revolutionary concept, sharing function model [5], where instead of replacing a solution by a similar solution, the focus was more on degrading the fitness of similar solutions. Most subsequent GA studies have used this model in solving multimodal optimization problems [6]. However, fitness sharing is computationally expensive because it requires the comparison of each couple of individuals to measure their distance at each generation.

Many other people's researches fall into parallel subpopulation method [7]. One important class of parallel subpopulation method is island model parallel GAs (IMGAs)[8]. Species conservation proposed by Jian-ping Li et al[7] also is a technique of parallel subpopulation method. The main drawback lies in these algorithms is that some additional control parameters have been introduced that need careful selection to ensure good algorithm performance. If we lack of the background of the multimodal function optimization problems, choosing suitable parameters is a difficult task.

This paper presents a novel two-phase steady-state evolutionary algorithm for solving multimodal optimization problems. The algorithm includes two phases: firstly, the steady-state evolutionary algorithm is used to get sub-optimal solution in the global search, it enables individual to draw closer to each optimal solution, accordingly population is divided into subpopulations automatically after the global search. Secondly, local search is carried in each subpopulation which focuses on the neighborhood of each optimal solution to gain higher precision solutions. The experiment results show that TSEA is very efficient for the optimization of multi-modal functions, usually it can obtain all the global optimal solutions by running once of the algorithm.

The rest of this paper is organized as follows: Section 2 gives the structure of TSEA and describes the implementation of the algorithm. The experiments and comparisons with other algorithms are illustrated by solving several benchmark problems in section 3. Finally, section 4 concludes with some brief remarks.

2 Description of Two-Phase Steady-State Evolutionary Algorithm(TSEA)

The algorithm(TSEA) includes two phases: global search phase and local search phase. The steady-state evolution algorithm is used to divided the population into subpopulations in global search phase, then local search is carried subsequently in the each subpopulation to obtain precise solutions. TSEA uses real-valued encoding without a mutation operator. The algorithmic description of TSEA is as follows:

```

Procedure TSEA algorithm
  Begin
    Randomly initialize population P(0);
    Evaluate P(0);
    t=0;
    {Global Search Phase}
    while(the termination condition for global search is
      not satisfied) do
      Begin
        select three parents from P(t) randomly;
        generate a child through Multi-Parent crossover
          (Three Parent);
        replace the worst individual of P(t) with the
          child according to the replacement strategy of
          global search ;
        t= t+1;
      end while;
  end while;

```

Determine the best individuals of each subpopulation (C_1, C_2, \dots, C_k) ;

{Local Search Phase}

For $i:=1$ to k do begin

Search the neighborhood of C_i to gain a optimal solution S_i ;

End For;

Output the optima (S_1, S_2, \dots, S_k) ;

End.

2.1 Multi-parent Crossover Operator (MPCO)

Multi-parent crossover operators, which mean that more than two parents are involved when generating offspring, are a more flexible version, generalizing the traditional two-parent crossover of nature. Much attention has been paying on them in the field of evolutionary computation. Numerical Experiments show that multi-parent crossover can keep a good balance between exploration and exploitation. So we design a Multi-parent crossover operator which is adopted in our algorithm. Fig.1 shows how to generate a new individual using Multi-Parent crossover operator.

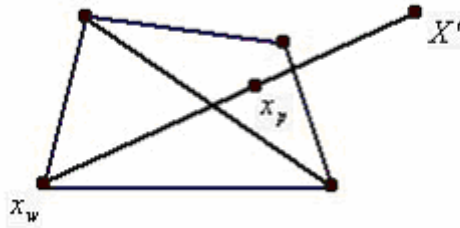


Fig. 1. Multi-Parent Crossover Operator

The flow of generating a new child is as follows:

1. M individuals $x_j, j=1,2,\dots,M$ are randomly selected in the population.
2. find the worst individual x_w in these M individuals (by comparing with the value of their fitness)
3. Denote the center of the remanent $M-1$ individuals as x_p :

$$x_p = \frac{\sum_{j=1}^M x_j - x_w}{M - 1} \tag{1}$$

4. find the reflecting point X' of x_w corresponding to x_p :

$$X' = 2 \times x_w - x_p \quad (2)$$

The most important advantage of this Multi-Parent crossover operator is that the new offspring generated will near the region where individuals may have higher fitness value, so there is rather larger probability that the new individual has higher fitness value than that of the worst individual in the population. Thus, this crossover operator can accelerate the constringency because of its strong direction. In fact, it can instruct its search direction using statistical information.

We use Three-Parent crossover operator in global search phase, meanwhile, Four-Parent crossover operator is adopted in local search phase, because the experiments show that the Three-Parent crossover is very suitable for searching global space but Four-Parent crossover has higher performance when searching the local neighborhood space.

2.2 Global Search Phase

The key elements in global search phase are multi-parent crossover operator and the replacement strategy. In global search phase, the algorithm should be get a good balance between finding better solutions and keep the diversity of population.

2.2.1 Replacement Strategy of Global Search

A widely used combination is to replace the worst individual only if the new individual is better. Meanwhile there are other different replacement strategies that try to enforce population diversity. Most of them are instances of the *crowding* methods, for example, *Restricted Tournament Selection*, *Worst Among Most Similar Replacement*, *Family Competition Replacement Schemes* and *CD/RW strategy*[9]. The strategy of replacing the worst is simple with little computational cost, but it is not suitable for solving multimodal function optimization problems, because the algorithm can only find partial optimal solutions if such strategy is adopted. The *CD/RW strategy* is a promised method, but it is computationally expensive. So we design a novel replacement strategy.

Denote the new individual and the worst individual respectively by X_{new} and X_{worst} :

If $better(X_{new}, X_{worst})$ then begin

$$\text{If } abs\left(\frac{f(X_{new}) - f(X_{worst})}{f(X_{new})}\right) > \alpha \text{ then } X_{new} \text{ replace } X_{worst}; \quad (3)$$

End;

Where α is a small positive real value, we fix it as 0.01.

On the one hand, this strategy is similar with the strategy of replacing the worst when in the early phase of global search, because the fitness of each individual has much difference at this time. On the other hand, this strategy can maintain the diversity in the late phase of global search phase, the reason lies that all individuals

have already clustered to the neighborhood of each optimal solution in the late phase of global search, there is tiny difference among every individual's fitness, thus there is little probability for the best individual of one subpopulation being replaced by another individual from other subpopulations if such strategy adopted. In this way, the diversity of the population is maintained and the quality of the solutions becomes improved, simultaneously. Moreover, this strategy is very simple and it needs little computational cost comparing with other replacement strategies.

2.2.2 The Termination Condition for Global Search

Denote the best individual and the worst individual respectively by X_{best} and X_{worst} . The global search will be terminated when the following formula(4) is satisfied, or the population will remain unchanged in successive K generations. Where ε is a given allowable error.

$$abs(f(X_{best}) - f(X_{worst})) \leq \varepsilon \quad (4)$$

Generally, if the formula (4) is satisfied, it means that every individuals have converge to each optimal solution after global search phase, thus there is no need to execute the local search phase again. For example, the local search phase will not execute when applying TSEA for solving the example 1 and example 3.

2.3 Local Search

How to obtain the precise solutions is the main task in local search phase. The steady-state evolutionary algorithm is still adopted to search the neighborhood of the best individuals which obtained after the global search. We use Four-Parent crossover operator and real-coding without mutation operator, meanwhile, the strategy of replacing the worst is adopted. To limit the search in the neighborhood of the best individual of one subpopulation, denote the best individual by C_i , the following constraint should be added to the origin function optimization problem:

$$(X - C_i)^2 \leq r \quad (5)$$

where r is a small positive real value.

3 Numerical Experiments and Analysis

When testing the algorithm on well understood problems, there are two measures of performance:

1. Success rate: the ratio of found modals and actual modals.
2. The average number of objective function evaluations required to find these optima.

The algorithm SCGA proposed in [7] is also a evolutionary algorithm for solving multimodal function global optimization problems which proved to be very effective. Some test problems are selected to compare the performance of the proposed algorithm with the algorithm SCGA.

Example 1[10] Humpback function (the function has six local optimal solutions, two of which are global optimal solutions)

$$\min f(x_1, x_2) = (4 - 2.1x_1^2 + x_1^4 / 3)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2 \tag{6}$$

where $x_1 \in [-3,3], x_2 \in [-2,2]$

Example 2 [10]N-dimension Shubert function(when n=2, the function has 720 local optimal solutions, 18 of which are global optimal solutions. When n=3,the function has 81 global optimal solutions) .

$$\min f(x_1, x_2, \dots, x_n) = \prod_{i=1}^n \sum_{j=1}^5 j \cos((j+1)x_i + j) \tag{7}$$

where $x_i \in [-10,10], i = 1,2, \dots, n$.

Example3 two-peak trap [7]

The fitness function of the two-peak trap is defined by:

$$F(c) = \begin{cases} \frac{160}{10}c & \text{for } 0 \leq c < 10 \\ \frac{160}{5}(15 - c) & \text{for } 10 \leq c < 15 \\ \frac{200}{5}(c - 15) & \text{for } 15 \leq c \leq 20 \end{cases} \tag{8}$$

This function has a global maximum of 200 at $c = 20$, but it has a “central” false maximum of 160 for $c = 10$.

To compare the performance of the proposed algorithm with the algorithm SCGA in [7], we use the same population size adopted in SCGA for each test problems, and the other control parameters are set as follows: $k=500, r=0.1, \epsilon=10^{-8}$.Table 1 summarizes the experimental results we obtained by using TSEA and statistics for the 30 independent runs. Table 1 also gives the comparison between our results and the lasted results [7] that we can find in the literature.

From the Table 1, we can see that the experimental results using the new algorithm are surprisingly good. For the benchmark problems,TSEA can find all optimal solutions by running once of the algorithm, Meanwhile, the average number of function evaluations of TSEA is less than that of the algorithm SCGA for every test problems. Moreover, there is no additional control parameter in TSEA which is sensitive to a special problem, but the choice of the species distance has a significant effect on the performance of the algorithm SCGA.Note that the mechanism adopted in the our algorithm to maintain the diversity of population is very simple, therefore it needs little computational cost at every generation,however,the algorithm SCGA needs more computational cost at every generation because of its more complicated method to maintain the diversity of population.The results show that our algorithm is very efficient for the global optimization of multimodal functions.

Table 1. The comparison of this algorithm and the algorithms SCGA in the [7]

Example No.	Algorithm	Population size	Actual modals	Success rate	Average number of function evaluations
Example 1	TSEA	50	2	100%	1824
	SCGA	50	2	100%	1836
Example 2(n=2)	TSEA	1000	18	100%	35016
	SCGA	1000	18	100%	64178
Example 2(n=3)	TSEA	4000	81	100%	352486
	SCGA	4000	81	100%	850338
Example 3	TSEA	50	1	100%	522
	SCGA	50	1	100%	625

4 Conclusions

This paper presents a two-phase steady-state evolutionary algorithm (TSEA) for solving function optimization containing multiple global optima. Comparing with other algorithms, it has the following advantages.

- (1) It designs a new multi-parent crossover operator with strong direction which can accelerate the convergence.
- (2) A novel replacement strategy is proposed to maintain the diversity of population. This strategy is very simple and effective with little computational cost.
- (3) Proposed algorithm need no additional control parameter which depends on a special problem. This is a most important feature which shows the generality of our algorithm.

Acknowledgment

This research is supported by the National Natural Science Foundation of China (No. 60473081,60133010).

References

1. De Jong, K. A. (1975). *An Analysis of Behavior of a Class of Genetic Adaptive Systems*. Ph.D. thesis, University of Michigan, Ann Arbor, Michigan.
2. Mahfoud, S.W. (1995). Niching methods for genetic algorithms. IlliGAL Technical Report 95001, Illinois Genetic Algorithms Laboratory, University of Illinois, Urbana, Illinois.
3. G. Harik, Finding multimodal solutions using restricted tournament selection. *Proc. 6th Int. Conf. Genetic Algorithms*, L. J. Eshelman (Ed.), (Morgan Kaufmann, San Mateo, CA, 1995) 24-31.

4. O.J. Mengshoel, D.E. Goldberg. Probabilistic crowding: Deterministic crowding with probabilistic replacement. In W. Banzhaf et al. (Eds.) Proc. Of the Genetic and Evolutionary Computation Conference GECCO-99, pp. 409-416, Morgan Kaufmann Publishers, San Francisco, CA, 1999.
5. Goldberg D E, Richardson J. Genetic algorithms with sharing for multi-modal function optimization. In: Grefenstette eds. Proceedings of the second International Conference on Genetic Algorithms. NJ: Lawrence Erlbaum Associates, 1987, 41-49
6. Bellomo D, Naso D, B.Turchiano. Improving genetic algorithms: an approach Based on multi-elitism and lamarckian mutation. In: A. E. Kamel, K.Mellouli, P. Bome, eds. Proceedings of 2002 IEEE International Conference on Systems, Man and Cybernetics. NJ:IEEE Press, 2002, 4: 6
7. Jianping Li, Marton E. Balizs et al. A Species Conserving Genetic Algorithm for multimodal function optimization. *Evolutionary Computation*, 2002,10(3): 207-234
8. Gordon V S, Whitley D et al. Dataflow parallelism in genetic algorithms. In: Männer, R., Manderick, B. eds. *Parallel Problem Solving from Nature 2*. Amsterdam: Elsevier Science, 1992, 533-542
9. M. Lozano, F. Herrera, and J.R.Cano. Replacement strategies to preserve useful diversity in steady-state genetic algorithms. *In Press*, March 2004.
10. Michalewicz, Z. (1996). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer-Verlag, New York, New York.

Finding Optimal Addition Chains Using a Genetic Algorithm Approach

Nareli Cruz-Cortés, Francisco Rodríguez-Henríquez,
Raúl Juárez-Morales, and Carlos A. Coello Coello

Computer Science Section, Electrical Engineering Department,
Centro de Investigación y de Estudios Avanzados del IPN,
Av. Instituto Politécnico Nacional No. 2508, México D.F.
nareli@computacion.cs.cinvestav.mx,
{francisco, ccoello}@cs.cinvestav.mx

Abstract. Since most public key cryptosystem primitives require the computation of modular exponentiation as their main building block, the problem of performing modular exponentiation efficiently has received considerable attention over the years. It is known that optimal (shortest) addition chains are the key mathematical concept for accomplishing modular exponentiations optimally. However, finding an optimal addition chain of length r is an **NP**-hard problem whose search space size is comparable to $r!$. In this contribution we explore the usage of a Genetic Algorithm (GA) approach for the problem of finding optimal (shortest) addition chains. We explain our GA strategy in detail reporting several promising experimental results that suggest that evolutionary algorithms may be a viable alternative to solve this illustrious problem in a quasi optimal fashion.

1 Introduction

Arguably, the field or modular exponentiation is the most important single arithmetic operation in public key cryptosystems. The search for efficient algorithm solutions for this problem has a long history whose roots can be traced as far back as the ancient works of Indian mathematicians in 200 B.C [1]. In addition to its historical and theoretical relevance, field exponentiation has many important practical applications in the areas of error-correcting codes and cryptography. Modular exponentiation is used in several major public-key cryptosystems such as RSA, Diffie-Hellman and DSA [2]. For instance, the RSA crypto-scheme is based on the computation of $M^e \bmod n$, where e is a fixed number, M is an arbitrarily chosen numeric message and n is the product of two large primes, namely, $n = pq$. Typical bit-lengths for n used in commercial applications range from 1024 up to 4096 bits. In addition, modular exponentiation is also a major building block for several number theory problems including integer prime testing, integer factorization, field multiplicative inverse computation, etc.

Let α be an arbitrary integer in the range $[1, n - 1]$, and e an arbitrary positive integer. Then, we define modular exponentiation as the problem of finding the

unique integer $\beta \in [1, n - 1]$ that satisfies the equation

$$\beta = \alpha^e \pmod n \tag{1}$$

In order to improve legibility, in the rest of this paper we will drop the modular operator whenever it results unambiguous.

The problem of determining the correct sequence of multiplications required for performing a modular exponentiation can be elegantly formulated by using the concept of *addition chains*. Formally, An *addition chain* for e of length l is a sequence U of positive integers, $u_0 = 1, u_1 \dots, u_l = e$ such that for each $i > 1$, $u_i = u_j + u_k$ for some j and k with $0 \leq j \leq k < i$. Therefore, if U is an addition chain that computes e as mentioned above, then for any $\alpha \in [1, n - 1]$ we can find $\beta = \alpha^e \pmod n$ by successively computing: $\alpha, \alpha^{u_1}, \dots, \alpha^{u_{l-1}}, \alpha^e$.

Let $l(e)$ be the shortest length of any valid addition chain for a given positive integer e . Then the theoretical minimum number of field multiplications required for computing the modular exponentiation of (1) is precisely $l(e)$. Unfortunately, the problem of determining an addition chain for e with the shortest length $l(e)$ is an **NP**-hard problem [2].

Across the centuries, a vast amount of algorithms for computing modular exponentiation have been reported. Reported strategies include: binary, m-ary, adaptive m-ary, power tree, the factor method, etc. [3, 1, 2]. On the other hand, relatively few probabilistic heuristics have been reported so far for finding near optimal addition chains [4, 5]. In this paper, we present a Genetic Algorithm (GA) suited to optimize addition chains. The results obtained suggest that this approach is a very competitive alternative to the solution of the problem.

2 Problem Statement

The problem addressed in this work consists of finding the shortest addition chain for an exponent e . Formally, an addition chain can be defined as follows,

Definition. An *addition chain* U for a positive integer e of length l is a sequence of positive integers $U = \{u_0, u_1, \dots, u_l\}$, and an associated sequence of r pairs $V = \{v_1, v_2 \dots, v_l\}$ with $v_i = (i_1, i_2), 0 \leq i_2 \leq i_1 < i$, such that:

$$u_0 = 1 \text{ and } u_l = e; \quad \text{for each } u_i, 1 \leq i \leq l, u_i = u_{i_1} + u_{i_2}.$$

The search space for computing optimal addition chains increments its size at a factorial rate as there exist $r!$ different and valid addition chains with length r . Clearly, the problem of finding the shortest ones becomes more and more complicated as r grows larger.

3 Deterministic Heuristics for Modular Exponentiation

In this section, we briefly review some deterministic heuristics proposed in the literature for computing field exponentiation. For a complete description of these and other methods, interested readers are referred to [1, 6].

Let e be an arbitrary m -bit positive integer e , with a binary expansion representation given as, $e = (1e_{m-2} \dots e_1 e_0)_2 = 2^{m-1} + \sum_{i=0}^{m-2} 2^i e_i$. Then,

$$y = x^e = x^{\sum_{i=0}^{m-1} 2^i e_i} = \prod_{i=0}^{m-1} x^{2^i e_i} = \prod_{e_i \neq 0} x^{2^i} \quad (2)$$

Binary strategies evaluate equation (2) by scanning the bits of the exponent e one by one, either from left to right (MSB-first binary algorithm) or from right to left (LSB-first binary algorithm) applying Horner's rule. Both strategies require a total of $m - 1$ iterations. At each iteration a squaring operation is performed, and if the value of the scanned bit is one, a subsequent field multiplication is performed. Therefore, the binary strategy requires a total of $m - 1$ squarings and $H(e) - 1$ field multiplications, where $H(e)$ is the Hamming weight of the binary representation of e . The binary method can be generalized by scanning more than one bit at a time. Hence, the window method (first described in [1]) scans k bits at a time. The window method is based on a k -ary expansion of the exponent, where the bits of the exponent e are divided into k -bit words or digits. The resulting words of e are then scanned performing k consecutive squarings and a subsequent multiplication as needed. For $k = 1, 2, 3, 4$ the window method is called, respectively, *binary*, *quaternary*, *octary* and *hexa* MSB-first exponentiation method.

4 The Proposed Genetic Algorithm

In this work, we present a Genetic Algorithm (GA) approach suited for finding optimal addition chains. In the rest of this Section we describe how the design decisions were taken.

4.1 Representation

In this work, we adopt an integer encoding, using variable-length chromosomes. Each element from the addition chain is directly mapped on each gene in the chromosome. Then, in this case, the genotype and the phenotype are both the same.

For example, if we are minimizing the addition chain for the exponent $e = 6271$, one candidate solution could be $1 - 2 - 4 - 8 - 10 - 20 - 30 - 60 - 90 - 180 - 360 - 720 - 1440 - 2880 - 5760 - 5970 - 6150 - 6240 - 6270 - 6271$.

This integer sequence represents a chromosome or individual I , where each gene I_k corresponds to one step on the addition chain, for $0 \leq k \leq l$ with length $l = 19$, and $I_l = 6271$.

4.2 Fitness Function

Since we are looking for the minimal addition chain's length, then the individual's fitness is precisely the addition chain's length, or in other words, the

chromosome's length. The shorter the chromosome's length is, the better its fitness value, and vice versa.

If we consider the previous addition chain, the associated fitness of this chain is 19.

4.3 Crossover Operator

The crossover operator creates two children from two parents. In our GA, we adopt *one point crossover*. Some extra considerations must be taken, however, mainly because of two reasons: first, it is necessary to assure that the resulting children are valid addition chains (i.e., feasible ones) and second, the chromosomes are of variable length. The way this operator produces offspring is illustrated in Figure 1. The genes before the crossover point are copied to the children as values (the alleles are copied); meanwhile from the genes after the crossover point only the rules are copied (rules indicate the positions from the chains which are selected to be added). See the following pseudocode:

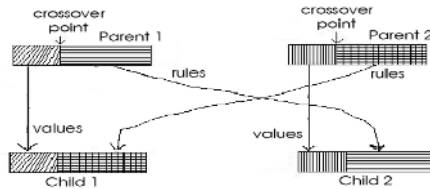


Fig. 1. Crossover Operator

Begin Function Crossover

For a pair of parents ($P1$ and $P2$) do:

1. Select a randomly selected crossover point p such that $(2 \leq p \leq l - 2)$ where l is the chromosome length.
2. Create child ($C1$), copy the $P1$'s values to the $C1$ starting from 0 until p is reached, hence,
 - For $(k = 0)$ to $(k = p)$ $C1_k \leftarrow P1_k$
 - From the point p until e is reached, complete the child $C1$ following the rules by which $P2$ was created, in the following way,
 - For $(k = p + 1)$ to $(k = length)$
 - Look for a and b values such that, $P2_k = P2_a + P2_b$
 - Set $C1_k \leftarrow C1_a + C1_b$
 - EndFor
3. Create child ($C2$):
 - For $(k = 0)$ to $(k = p)$
 - $C2_k \leftarrow P2_k$
 - EndFor
 - For $(k = p + 1)$ to $(k = length)$
 - Look for a and b values such that, $P1_k = P1_a + P1_b$
 - Set $C2_k \leftarrow C2_a + C2_b$
 - EndFor

End Function Crossover

We point out that the crossover operator is applied in the way described above only when that data manipulation does not produce values exceeding the exponent e . In case the value indicated by the crossover operator exceeds e , then the value assigned would be the maximum allowable.

4.4 Mutation Operator

It is noted that our definition of this operator allows us to introduce random changes into the chromosome, while preserving addition chains' validity.

BEGIN Function Mutation

For each child (C) do:

1. Randomly select a mutation point i and a random number j such that $2 \leq j < i < (l - 2)$, where l is the chromosome length.
2. The new value of the child at the mutation point C_{i+1} will be $C_{i+1} = C_i + C_j$
3. Repair the upper part of the chromosome $\{C_{k>i+1}\}$, using the following criterion:
 - For $k = i + 2$ to l , with $C_l = e$ do
 - If (*Flip*(Z)) then use the doubling rule whenever is possible, i.e, $C_k = 2C_{k-1}$
 - Else if (*flip*(0.5)) set $C_k = C_{k-1} + C_{k-2}$
 - Else set $C_k = C_m + C_n$, where m and n are two randomly selected integers such that $0 \leq m, n < l$.

END Function Mutation

Flip(Z) is a function that receives an input parameter Z such that $0 \leq Z \leq 1$. It returns *true* with probability Z , or *false* in other case.

General GA

Having defined the main Genetic Algorithm primitives, we proceed to put them together into the skeleton structure of the GA strategy outlined below,

BEGIN-General-GA

1. Randomly create an initial population size N .
2. Repeat:
 - 2.a. Compute individuals' fitness.
 - 2.b Select the N parents to be reproduced.
 - 2.c With probability Pc , apply crossover operator to the N parents.
 - 2.d Apply the mutation operator to the children with a probability Pm .
 - 2.e. Children will form the next generation population.
3. Go to step 2.a. until a predetermined number of *Generations* is reached.
4. Report the fittest individual.

END-General-GA.**5 Experiments and Results**

In order to validate the GA approach described in the previous Section, we conducted a series of experiments, with the aim of comparing our GA's experimental results against the ones obtained by using several traditional deterministic methods.

Table 1. Accumulated addition chain lengths for exponents $e \in [1, 1000]$

Optimal value=10808			
Strategy	Total length	Strategy	Total length
Dyadic [6]	10837	Quaternary	11479
Total [6]	10821	Genetic Algorithm	Best: 10818
Fermat [6]	10927		Average: 10824.07
Dichotomic [6]	11064		Median: 10824
Factor [6]	11088		Worst: 10830
Binary	11925		Std.Dev.: 2.59

The first set of experiments consisted on finding the accumulated addition chain lengths for all exponents e in a given interval as it was done in [6]. Then, as a second test, we applied our genetic algorithm to a special class of exponents whose optimal addition chains are particularly hard to find. All our experiments were performed by applying the following GA’s parameters: Population size $N = 100$, Number of Generations = 300, Crossover Rate $Pc = 0.6$, Mutation Rate $Pm = 0.5$, Probability $Z = 0.7$ (used in the mutation operator), Selection = Binary Tournament. All the statistical results shown here were produced from 30 independent runs of the algorithm with different and independent random seeds (adopting a uniform distribution).

Using our genetic algorithm approach, we computed the accumulated addition chain lengths for all the first 1000 exponents, i.e. $e \in [1, 1000]$. The accumulated value so obtained was then compared against the accumulated values reported in [6] by applying the following deterministic methods: Dyadic, Total, Fermat, Dichotomic, Factor, Quaternary and Binary [1, 6]. All results found are shown in Table 1. It can be seen that in the best case, our GA approach obtained better results than all the other six methods. In average, the GA approach was ranked in second place, only behind the Total method. It is noted that none of the features strategies was able to find the optimal value that was found by performing an exhaustive search.

Furthermore, we computed the accumulated addition chain lengths for all the exponents in the ranges $e \in [1, 512]$, $e \in [1, 2000]$ and $e \in [1, 4096]$. The methods used were the GA strategy, the binary method and the quaternary method. Once again and for comparative purposes, we computed the corresponding optimal values (obtained by enumeration). Those results are shown in Table 2.

Clearly, the results obtained by the GA strategy outperformed both, the binary and the quaternary method, even in the worst case. We can observe that for the three cases considered (i.e., 512, 2000 and 4096), the GA obtained a reasonably good approximation of the optimal value.

A Special Class of Exponents Hard to Optimize

Let $e = c(r)$ be the smallest exponent that can be reached using an addition chain of length r . Solutions for that class of exponents are known up to $r = 30$ and a compilation of them can be found in [7]. Interesting enough, the computational difficulty of finding shortest addition chains for those exponents seems to be among the hardest if not the hardest one from the studied exponent families [1]. We show the solutions found by the GA for the class of exponents shown in Table 3.

Table 2. Accumulated addition chain lengths for 512, 2000 and 4096

for all $e \in [1, 512]$	for all $e \in [1, 2000]$	for all $e \in [1, 4096]$
Optimal: 4924	Optimal: 24063	Optimal: 54425
Binary: 5388	Binary: 26834	Binary: 61455
Quaternary: 5226	Quaternary: 25923	Quaternary: 58678
Genetic Algorithm	Genetic Algorithm	Genetic Algorithm
Best: 4925	Best: 24124	Best: 54648
Average: 4927.7	Average: 24135.17	Average: 54684.13
Median: 4927	Median: 24136	Median: 54685
Worst: 4952	Worst: 24144	Worst: 54709
Std.Dev.: 4.74	Std.Dev.: 5.65	Std.Dev.: 13.55

Table 3. Shortest addition chains for a special class of exponents

exponent $e = c(r)$	Addition Chain	Length r
1	1	0
2	1 - 2	1
3	1 - 2 - 3	2
5	1 - 2 - 3 - 5	3
7	1 - 2 - 3 - 5 - 7	4
11	1 - 2 - 3 - 5 - 8 - 11	5
19	1 - 2 - 3 - 5 - 7 - 12 - 19	6
29	1 - 2 - 3 - 4 - 7 - 11 - 18 - 29	7
47	1 - 2 - 3 - 5 - 10 - 20 - 40 - 45 - 47	8
71	1 - 2 - 3 - 5 - 7 - 12 - 17 - 34 - 68 - 71	9
127	1 - 2 - 4 - 6 - 12 - 24 - 48 - 72 - 120 - 126 - 127	10
191	1 - 2 - 3 - 5 - 10 - 20 - 21 - 42 - 63 - 126 - 189 - 191	11
379	1 - 2 - 4 - 5 - 10 - 15 - 25 - 50 - 75 - 150 - 300 - 375 - 379	12
607	1 - 2 - 4 - 6 - 12 - 24 - 48 - 96 - 192 - 384 - 576 - 600 - 606 - 607	13
1087	1 - 2 - 3 - 6 - 12 - 18 - 36 - 74 - 144 - 216 - 432 - 864 - 865 - 1081 - 1087	14
1903	1 - 2 - 3 - 5 - 10 - 13 - 26 - 52 - 104 - 105 - 210 - 420 - 840 - 1680 - 1890 1903	15
3583	1 - 2 - 3 - 6 - 12 - 18 - 36 - 72 - 108 - 216 - 432 - 864 - 1728 - 3456 - 3564 - 3582 - 3583	16
6271	1 - 2 - 3 - 6 - 12 - 24 - 48 - 96 - 192 - 384 - 768 - 1536 - 3072 - 6144 - 6240 - 6264 - 6270 - 6271	17
11231	1 - 2 - 3 - 6 - 12 - 24 - 25 - 50 - 100 - 200 - 400 - 800 - 1600 - 3200 - 6400 - 9600 - 11200 - 11225 - 11231	18
18287	1 - 2 - 3 - 6 - 9 - 15 - 30 - 45 - 47 - 94 - 188 - 190 - 380 - 760 - 1520 - 3040 - 6080 - 12160 - 18240 - 18287	19
34303	1 - 2 - 3 - 6 - 12 - 14 - 28 - 56 - 112 - 224 - 448 - 504 - 1008 - 2016 - 4032 - 8064 - 16128 - 32256 - 34272 - 34300 - 34303	20
65131	1 - 2 - 3 - 6 - 12 - 24 - 48 - 72 - 144 - 288 - 576 - 1152 - 2304 - 4608 - 4611 - 9222 - 18444 - 27666 - 55332 - 55908 - 65130 - 65131	21
110591	1 - 2 - 4 - 5 - 10 - 20 - 40 - 80 - 160 - 320 - 640 - 1280 - 2560 - 2570 5140 - 7710 - 12850 - 25700 - 51400 - 102800 - 110510 - 110590 - 110591	22
196591	1 - 2 - 3 - 6 - 12 - 15 - 30 - 60 - 120 - 240 - 480 - 720 - 1440 - 2880 - 5760 11520 - 23040 - 46080 - 92160 - 184320 - 19584 - 196560 - 196590 - 196591	23
357887	1 - 2 - 3 - 4 - 8 - 16 - 32 - 64 - 128 - 256 - 257 - 514 - 771 - 11542 - 3084 - 6168 - 12336 - 24672 - 49344 - 49347 - 98691 - 148038 - 296076 - 345423 - 357759 - 357887	24
685951	1 - 2 - 4 - 6 - 7 - 14 - 21 - 42 - 84 - 168 - 336 - 504 - 840 - 1680 - 3360 - 6720 - 13440 - 26880 - 53760 - 57120 - 114240 - 228480 - 342720 - 685440 - 685944 - 685951	25
1176431	1 - 2 - 4 - 5 - 10 - 15 - 19 - 38 - 76 - 152 - 304 - 608 - 612 - 1224 - 2448 - 4896 - 9792 - 19584 - 29376 - 58752 - 117504 - 235008 - 352512 - 587520 - 1175040 - 1176264 - 1176416 - 1176431	27
2211837	1 - 2 - 3 - 6 - 9 - 15 - 30 - 60 - 120 - 126 - 252 - 504 - 1008 - 2016 - 4032 - 8062 - 16128 - 16143 - 32286 - 64572 - 129144 - 258288 - 516576 - 1033152 - 2066304 - 2195448 - 2211591 - 2211717 - 2211837	28
4169527	1 - 2 - 3 - 6 - 12 - 24 - 48 - 96 - 192 - 384 - 768 - 1536 - 2304 - 4608 - 9216 - 18432 - 36864 - 73728 - 147456 - 294912 - 589824 - 589825 - 1179650 - 1769475 - 3538950 - 4128775 - 4169527 - 414167943 - 4169479 - 4169527	29
7624319	1 - 2 - 3 - 6 - 12 - 18 - 36 - 72 - 144 - 288 - 576 - 1152 - 1224 - 2448 - 4896 - 9792 - 19584 - 39168 - 78336 - 156672 - 313344 - 626688 - 1253376 - 1254600 - 1274184 - 1274185 - 2548370 - 5096740 - 6370925 - 7624301 - 7624319	30
14143037	1 - 2 - 3 - 6 - 12 - 18 - 30 - 60 - 120 - 240 - 480 - 960 - 961 - 1922 - 3844 - 7688 - 11532 - 23064 - 46128 - 92256 - 184512 - 369024 - 461280 - 830304 - 1660608 - 3321216 - 6642432 - 13284864 - 14115168 - 14138232 - 14142076 - 14143037	31

It is noted that in 24 out of 30 exponents, the GA approach was able to find the shortest addition chain. However, for the 6 remaining exponents (namely, 357887, 1176431, 2211837, 4169527, 7624319 and 14143037), our GA strategy found addition chains that were one unit above the optimal.

6 Conclusions and Future Work

In this paper we described how a genetic algorithm strategy can be applied to the problem of finding shortest addition chains for optimal field exponentiation computations. The GA heuristic presented in this work was capable of finding almost all the optimal addition chains for any given fixed exponent e with $e < 4096$. Taking into account the optimal value (which was found by enumeration) the percentage error of our GA strategy was within 0.4% from the optimal for all cases considered. In other words, for any given fixed exponent e with $e < 4096$, our strategy was able to find the requested shortest addition chain in at least 99.6% of the cases. In a second experiment for assessing the actual power of the GA strategy as a search engine, we tested it for generating the shortest addition chains of a class of exponents particularly hard to optimize, whose optimal lengths happen to be known for the first 30 members of the family. In most cases considered, the GA strategy was able to find the optimal values.

Acknowledgments

The first and second authors acknowledge support from CONACyT through the CONACyT project number 45306. The third and fourth authors acknowledge support from CONACyT through the NSF-CONACyT project number 42435-Y.

References

1. Knuth, D.: The Art of Computer Programming. 3rd. edn. Addison-Wesley, Reading, Massachusetts (1997)
2. Menezes, A. J., van Oorschot, P., Vanstone, S.: Handbook of Applied Cryptography. CRC Press, Boca Raton, Florida (1996)
3. Gordon, D. M.: A Survey of Fast Exponentiation Methods. *Journal of Algorithms*, Vol. 27 No.1 (1998) 129–146
4. Cruz-Cortés, N., Rodríguez-Henríquez, F., Coello, C.: On the Optimal Computation of Finite Field Exponentiation. In: Lemaître, C., Reyes, C. and González, J. (eds.): *Advances in Artificial Intelligence - IBERAMIA 2004: 9th Ibero-American Conference on AI. Lecture Notes in Computer Science*, Vol. 3315. Springer-Verlag, (2004) 747–756
5. Bos, J., Coster, M.: Addition Chain Heuristics. In: Brassard, G. (ed.): *Advances in Cryptology —CRYPTO 89. Lecture Notes in Computer Science*, Vol. 435. Springer-Verlag, (1989) 400–407
6. Bergeron, F., Berstel, J., Brlek, S.: Efficient Computation of Addition Chains. *Journal de thorie des nombres de Bordeaux*, Vol. 6 (1994) 21–38
7. Bleinchenbacher, D., Flammenkamp, A.: An Efficient Algorithm for Computing Shortest Addition Chains. Available at: <http://www.uni-bielefeld.de/~achim> (1997)

Using Reconfigurable Architecture-Based Intrinsic Incremental Evolution to Evolve a Character Classification System

Jin Wang, Je Kyo Jung, Yong-min Lee, and Chong Ho Lee

Department of Information Technology & Telecommunication,
Inha University, Incheon, Korea
wangjin_liips@yahoo.com.cn

Abstract. Evolvable hardware (EHW) has been employed in the circuit design automation domain, as an alternative to traditional human being designer. However, limited by the scalability of EHW, at present the scales of all the evolved circuits are smaller than the circuits designed by traditional method. In this paper, a character classification system for recognizing 16 characters was evolved by a novel evolution scheme: reconfigurable architecture-based intrinsic incremental evolution. The entire EHW system is implemented on one Xilinx Virtex xcv2000E FPGA that is fitted in the Celoxica RC1000 board. Hardware evolutionary result proved that the new method could bring us a scalable approach to EHW by efficiently limiting the chromosome string length and reducing the time complexity of evolutionary algorithm (EA).

1 Introduction

As an alternative solution to digital electronic circuit design, EHW attracted increasing attentions after the early 1990's. EHW is a hardware which is based on programmable logic device (e.g. PLDs and FPGAs) and its inner architecture and function can be reconfigured and optimized by using EA to search a large number of candidate solutions in a certain criterion. Using EA which is a robust search tool, EHW may find a solution which is untouchable by traditional design scheme or bring us a better result circuit. On the other hand, the evolutionary design approach does not required a priori knowledge of any particular design domain. We just need to illuminate the relation of the target system's input and expected output, and then EHW automatically finishes other jobs in circuit design process.

The previous features of EHW give us a promise but there is still a long way to go before EHW is employed as a real-world application of circuit design automation. One of the major obstacles is the scale of the evolved circuit. EHW is proved that it could evolve different circuits successfully, but for our best knowledge, most of the evolved circuits are on a tiny scale [1, 2]. The importance of scalability issue of EHW has been recognized by several researchers [3, 4, 5]. Various schemes for approaching the scalability problem of EHW have been undertaken. These methods include variable length chromosome GA [6], function

level evolution [7], parallel genetic algorithm [8] and increased complexity evolution [3] etc. In this paper, a novel scheme: reconfigurable architecture-based intrinsic incremental evolution is introduced to conquer the scalability problem of EHW.

This paper is organized as follows. Section 2 describes some basic concepts about the scalability of EHW and the reconfigurable architecture-based incremental evolution. In Section 3, hardware implement is discussed. The experimental setup and results are presented in section 4. Section 5 concludes the paper.

2 Scalability of EHW and a Novel Approach

Generally, FPGAs (Field Programmable Gate Array) are the most popular commercial programmable logic device which is used for digital EHW. A general issue of EHW is that EHW is only limited in designing small scale circuits, although the FPGA devices themselves, can implement complex circuits. This issue is defined as the scalability problem of EHW. X. Yao [4] divide the scalability of EHW into two parts: firstly, the scalability of the chromosome representation of electronic circuits, and secondly, the scalability of the time complexity of EA.

The first problem is how we can reduce the chromosome string length of EA. A generally path to approach EHW usually directly maps the configuration bit-streams of FPGA as the chromosome of EA. Obviously, when the scale of target circuit increases, the length of chromosomes string also grows remarkably. Large chromosome string length often makes the search space of EA too huge to be able to find an anticipant result system.

Moreover, the scalability related to the time complexity of EA is a more pivotal issue. Generally, as the complexity of the EHW increased, both of the search space of EA and the difficulty for EA to converge an optimized solution mount up. These features induce the rapid increase of the numbers of populations and generations in EA. On the other hand, complex EHW application usually leads a large period to evaluate the fitness value which can be considered as the most time-consuming part in individual evaluation process. All of the above features lead a long evolution period of EA. However, it is impossible to run an EA for several days. Several EHW experiments [3, 5] proved the time complexity has been established. According to the above discussions, time complexity could be determined as:

$$Time = ngen \times n \times (t_e + t_g) \quad (1)$$

In the Eq. (1), we define the fitness value evaluating time of each individual as t_e , t_g is the time for the EA to perform genetic operations to an individual, $ngen$ indicates the number of generations in evolution process, n is the size of the population, $Time$ is the total evolution period.

Faced with the scalability issue of EHW, a novel scheme: reconfigurable architecture-based intrinsic incremental evolution is introduced in this paper. A simple 16 characters classification system was evolved to show how the new evolution scheme could bring us a scalable approach to EHW. The target character classification system is expected to identify 16 characters from A to P as

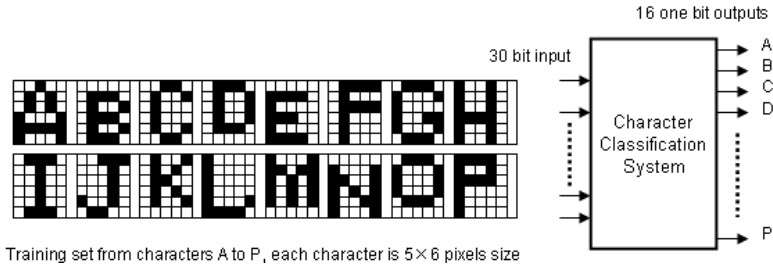


Fig. 1. Diagram of character classification system for patterns A to P identification

shown in Fig. 1. To process the input characters of 5×6 pixels, where each pixel can be 0 or 1, the character classification system includes a 30 bits input port. Each 1 bit input responds to one of pixel in the input characters. The size of the system output is equal to the amount of the characters need to be identified and is 16 bits. Each bit of output corresponds to one character. During the characters distinguishing process, the output port corresponding to the input character should be 1, synchronously the other outputs should be 0.

2.1 Incremental Evolution

The basic ideal of incremental evolution was first introduced by Torresen as a divide-and-conquer approach to EHW [9]. The main motive for this work is to allow for evolving complex digital systems rather than size-limited circuits. Compared to evolving the intact system in a single run, incremental evolution process is first undertaken individually on a set of sub-systems, and then the evolved sub-systems are employed as the building blocks which are used to further evolution or structure of a larger and more complex system.

To approach the incremental evolution, in this paper, the entire function of result system is divided into several sub-functions. As shown in Fig.2, in this scenario, a simple four characters classification system is evolved from two different strategies. In the left case, an EHW system is directly evolved. However, in the right case, the top system is divided into two sub-systems. In the evolution process, each sub-system inputs all characters from A to D as training set but only employs two output ports. In the evolved sub-system, the output port corresponding to the input character should be 1, synchronously the other output should be 0. Apparently, each of the evolved sub-systems could just individually classify two corresponding characters. When other two characters are input, the subsystem is evolved to 0 on each output. Thus, the final systems still has the same classification ability as the system evolved in one operation.

By dividing the top-function into several simpler sub-functions, incremental evolution technique can efficiently make the search space of EA simpler than directly evolution. On the other hand, we can spend smaller size of chromosome to present each sub-systems, it bring us a smaller search space of EA. The parameter of $negn \times n$ which is defined in Eq. (1) can be expected to be reduced because of the simpler and smaller search space of EA. The hardware

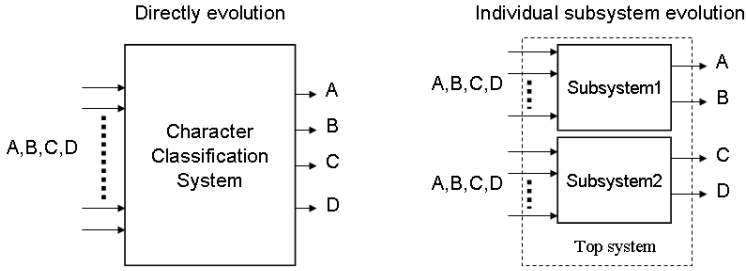


Fig. 2. Employing incremental evolution in four characters identification

experiments in section 4 will exhibit the improvement of $negn \times n$ by employing the incremental evolution approach.

2.2 FPGA Based Reconfigurable Architecture

Reconfigurable architecture which is inspired by the Cartesian Genetic Programming [10] was successfully introduced to the digital circuit evolutionary design field as a novel rapid reconfigurable hardware platform utilizing conventional FPGAs [11, 12]. Fig.3 presents a typical reconfigurable platform which is implemented in a FPGA. For applying incremental evolution, according to Cartesian Genetic Programming, a $N \times M$ array of function elements (FE) builds a reconfigurable component which executes a sub-function of EHW as shown in Fig.4. A reconfigurable unit which presents top-function of EHW includes several reconfigurable components. Each input ports of FEs can be connected to one of the outputs from the previous L columns of the FEs array. Each FE supports some predefined functions which are applied to process the inputs of FE. On contrast to directly map the outside configuration bit-streams of FPGA as chromosome, reconfigurable architecture based EHW employs configuration bits as chromosome which is generated by inner EA unit. By continuously changing the configuration bits which confirm the interconnection of FEs array and the functions implemented in each FE, the reconfigurable unit can be evolved.

The applications of local interactions of FEs and internal self-reconfiguration feature of EHW make local learning possible which is proved could bring us more powerful computation ability than traditional computer [3]. The powerful computation ability of reconfigurable architecture gives us a promise to efficiently reduce t_e and t_g which are defined in Eq.(1). Otherwise, the scale of FEs is flexible and can be manually defined according to the requirement of a given application. This feature gives us a potential path to control the chromosome length of a non-trivial combinational circuit.

3 Hardware Execution

Our EHW system is designed by using VHDL code and is implemented by a Xilinx Virtex XCV2000E FPGA chip which is fitting in the Celoxica RC1000

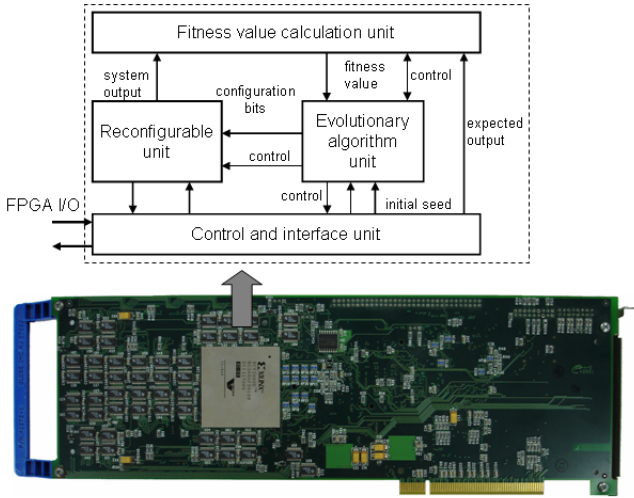


Fig. 3. Reconfigurable architecture-based EHW implemented by a Virtex XCV2000E FPGA in the RC1000 board

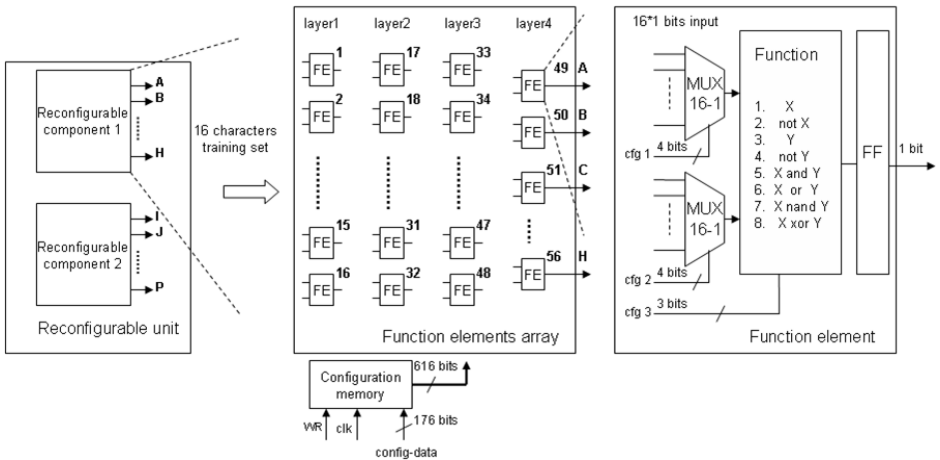


Fig. 4. Hardware architecture of reconfigurable unit

board. RC1000 board is shown in Fig. 3. In this paper, our task is to recognize 16 different characters. As our previous discussion, each reconfigurable component must input all 16 characters as training set. In Fig.4, the reconfigurable unit is divided into two reconfigurable components and each one owns 8 single bit outputs to classify 8 corresponding characters respectively.

Each reconfigurable component consists of a configuration memory and a FEs array. By the former experiments, the number of layers in the FEs array is defined as 4. Except for the last layer in the array, 16 uniform FEs are placed in each layer. The amount of FEs in the last layer is equal to the number of

sub-system outputs. Each FE in layer 2,3,4 can be reprogrammed to perform one of eight logic functions that are evident from Fig. 4. In the layer 1, only two logic operations of buffer X and inverter Y are available for each FE. Each FE has two 1 bit input port. Any input ports of FE could be connected to one of the reconfigurable component inputs (FEs in layer 1) or to one of the outputs of the FEs at the previous layer (FEs in layer 2, 3, 4). In layer 1, for processing the 30 bits input, two 32-input multiplexers are equipped by FEs on contrast to employ two 16-input multiplexers which are equipped by the FEs in the other layers. Two redundant input ports of 32-input multiplexer are directly defined as binary number 1 and 0. Each FEs needs 11 configuration bits which are uploaded from corresponding configuration memory to determine its active function and input connection (In the first layer, $5 + 5 + 1$ bits, in the other layers, $4 + 4 + 3$ bits).

The evolutionary operations which are implemented by the EA unit only include selection and mutation, crossover is not used. The initial population of four individuals is generated randomly. New populations are produced as follow. The mutated versions of the four initial individuals are downloaded to the reconfigurable unit as configuration bits. Depending on fitness values of the four individuals, the individual who has the highest fitness is chosen as the parent. Its four mutated versions provide the new population for the next generation. If one of the new individual has the better fitness than its parent, it substitutes its parent and is chosen as the new parent to generate the next generation. Based on experiments, we decided to mutate about 9 bits per intact individual on average. Fitness value of each individual is calculated in the fitness value calculation unit. The fitness function is defined as equal to the sum of the number of correct outputs in the system output vectors set. Control and interface unit executes the function of communicating with the outside environment.

4 Experiments and Results

The major objective of our experiment is to demonstrate: reconfigurable architecture based intrinsic incremental evolution could approach scalability issue of EHW from the fields of limiting both chromosome string length and time complexity of EHW.

According to the synthesis report, a single reconfigurable component uses 5% (about 1000 slices) of the Virtex XCV2000E FPGA which totally owns 19200 slices, the EA unit takes 4011 slices. The design can operate at 97.513MHz based on the synthesis report. However, the actual hardware experiment is run at 30 MHz because of easier synchronization with PCI interface. Considering the scale of our device, just 4 sub-systems, 2 sub-systems based incremental evolution is executed. Directly evolving the top-system is also executed as a contrast to the incremental evolution approach.

In incremental evolution process, for example dividing the top-system into 2 sub-systems, the reconfigurable component for the characters A to H classification is evolved first. This evolution process runs until a correct solution is found or the predefined number of generations is exhausted (the value of predefined

Table 1. The hardware evolving result of 16 characters classification system

Type of sub-system	Number of generations (avg.)	Total evolution time (avg.)	Chromosome length (bit)	device cost (slice)
16 characters (A to P)	45521490	97.11 sec	704	5309
8 characters (A to H)	173241	0.96 sec	616	6247
8 characters (I to P)	275067			
4 characters (A to D)	4055			
4 characters (E to H)	4499	0.04 sec	572	8123
4 characters (I to L)	3948			
4 characters (M to P)	5549			

number is 2^{32}). After this phase, the second reconfigurable component for the characters I to P classification is evolved too.

Table 1 shows statistics for 3 different evolution strategies. All average results are obtained from 20 runs. An interesting observation is that the difference of the incremental evolution schemes induces the diversity of total number of generations to evolve a system to classify 16 characters. For example the sum of number of generations (18051) required for evolving a 16 characters classification system which is divided into 4 sub-systems is 25 times less than the number of generations (448308) required for evolving the same system which is divided into 2 sub-systems. By introducing the increment evolution, the chromosome string length of EA is also reduced. Higher device cost of EHW also appears as the number of sub-systems increase. However the main scalability limitations of EHW aren't the hardware device scale, but the chromosome length and time complexity.

On the other hand, according to Torresen's report [3, 9], limited by the time complexity of EHW, evolving the same 16 characters classification system directly is unsuccessful in his experiment. However, in our experiments, with the introducing of the reconfigurable architecture-based intrinsic evolution technique, even directly evolving the 16 characters classification system, we can obtain the result in an acceptable time (97.11 sec). Reconfigurable architecture based EHW exhibits us its powerful computation ability to conquer the time complexity of EHW.

5 Conclusions

The work reported in this paper describes a scalable approach to digital circuit evolutionary design: reconfigurable architecture-based intrinsic incremental evolution. Incremental evolution can efficiently reduce the time complexity of EA by bringing a simpler and smaller search space of EA. With a powerful computation platform which is built on the reconfigurable architecture, the EHW system is utilized to evolve a 16 characters classification system in a very short time.

Acknowledgment

This work was supported by INHA University Research Grant.

References

1. Higuchi, T. et al.: Real-World Applications of Analog and Digital Evolvable Hardware. *IEEE Transactions on Evolutionary Computation* 3 (1999) 220-235
2. Hollingworth, G. et al.: The Intrinsic Evolution of Virtex Devices Through Internet Reconfigurable Logic. In: *Proc. Of the 3rd International Conference on Evolvable Systems: From Biology to Hardware ICES'00*, LNCS 1801, Springer-Verlag, Berlin (2000) 72-79
3. Torresen, J.: A scalable approach to evolvable hardware. *Journal of Genetic Programming and Evolvable Machines* 3 (2002) 259-282
4. Yao, X., Higuchi, T.: Promises and challenges of evolvable hardware. *IEEE Transactions on Systems, Man, and Cybernetics* 29 (1999) 87-97
5. Hereford, J., Gwaltney, D.: Design Space Issues for Intrinsic Evolvable Hardware. In: *Proc. of the 2004 NASA/DoD Conference on the evolvable Hardware*, IEEE Computer Society, Los Alamitos (2004) 231-234
6. Kajitani, I. et al.: Variable Length Chromosome GA for Evolvable Hardware. In *Proc. 3rd Int. Conf. on Evolutionary Computation, ICEC96* (1996) 443-447
7. Murakawa, M. et al.: Hardware Evolution at Function Level. In : *Proc. Of the Parallel Problem Solving from Nature PPSN IV*, LNCS 1141, Springer-Verlag , Berlin (1996) 62-72
8. Cantu-Paz, E.: A survey of parallel genetic algorithms. *Calculateurs Paralleles, Reseaux et Systems Repartis* 10 (1998) 141-171
9. Torresen, J.: A divide-and-conquer approach to evolvable hardware. in *Evolvable Systems: From Biology to Hardware. Second International Conference, ICES 98*, LNCS 1478, Springer-Verlag (1998) 57-65
10. Miller, J., Thomson, P.: Cartesian Genetic Programming In: *Proc. Of the Third European Conference on Genetic Programming*, LNCS 1802, Springer-Verlag (2000) 121-132
11. Sekanina, L., Friedl, S.: On Routine Implementation of Virtual Evolvable Devices Using COMBO6, In: *Proc. of the 2004 NASA/DoD Conference on Evolvable Hardware*, Los Alamitos, US, ICSP (2004) 63-70
12. Wang, J. et al.: Evolutionary Design of Image Filter Using The Celoxica Rc1000 Board. *International conference on control, automation and systems, ICCAS2005*, Korea (2005) 1355-1360

On the Relevance of Using Gene Expression Programming in Destination-Based Traffic Engineering

Antoine B. Bagula and Hong F. Wang

Department of Computer Science, University of Stellenbosch, 7600 Stellenbosch, South Africa
bagula@cs.sun.ac.za
<http://www.cs.sun.ac.za/~bagula>

Abstract. This paper revisits the problem of Traffic Engineering (TE) to assess the relevance of using Gene Expression Programming (*GEP*) as a new fine-tuning algorithm in destination-based TE. We present a new TE scheme where link weights are computed using *GEP* and used as fine-tuning parameters in destination-based path selection. We apply the newly proposed TE scheme to compute the routing paths for the traffic offered to a 23- and 30-node test networks under different traffic conditions and differentiated services situations. We evaluate the performance achieved by the *GEP* algorithm compared to a memetic and the Open Shortest Path First (*OSPF*) algorithms in a simulated routing environment using the NS packet level simulator. Preliminary results reveal the relative efficiency of *GEP* compared to the memetic algorithm and *OSPF* routing.

1 Introduction

Current generation IGP protocols such as Open Shortest Path First (*OSPF*) [1] are poorly equipped to support QoS and substantial increases of traffic load without the need for Traffic Engineering (TE); a network management technique that achieves QoS agreements between the available resources and the current and expected traffic offered to a network. These protocols are based on destination-based routing; a packet forwarding paradigm where IP forwarding uses solely the destination address specified in the packet header. Despite its scalability which contributed to the large expansion of the Internet, destination-based routing leads to opportunistic bandwidth sharing overloading some portions of the network while leaving some others unused.

The *Link weight optimisation approach* [2] was proposed in the context of destination-based TE to improve *OSPF* routing performance by using a smart selection of link weights (metrics) to overcome the limitations of destination-based routing. This approach has the advantages of (1) simplicity (2) capability of using diverse performance constraints and (3) compatibility with traditional IGPs. However finding link metrics which minimise the maximum link utilisation is NP-hard. Heuristic solutions using analogies with natural and social systems have been proposed to optimise IGP routing [3]. These solutions referred to as Memetic Algorithms (*MAs*) belong to a class of evolutionary algorithms (*EAs*) which use a hybrid optimisation model where a local search is used to complement classical global search.

EAs can find acceptably good solutions to problems by examining and manipulating a set of possible solutions from a set of designs but are not guaranteed to find the

global solution to a problem. They are based on an evolutionary paradigm using concepts from real-world genetics to evolve solutions to problems. In *EAs* each iteration of an algorithm transforms one population of individuals into a new generation, using some pre-determined fitness measure for an individual with the expectation of achieving the survival of the fittest populations. The strong correlation between the performance and costs of the algorithms and the characteristics of the initial populations is another limitative factor of some of the evolutionary algorithms such as *MA*s. Randomly generated initial populations are simple and less time consuming but are not easy to use by *MA*s due to structural constraints and the kind of genetic operators available to create genetic modifications. Gene Expression Programming (*GEP*) [4] is another subset of *EAs* that overcomes these limitations by using randomly generated initial populations. However to the best of our knowledge *GEP* has never been used to address IP routing issues.

This paper revisits the problem of traffic engineering to assess the relevance of using *GEP* as a new paradigm to improve IGP routing. The main contributions of this paper are

- **Evolutionary optimisation.** We consider a new routing optimisation model where optimal link weights computed using *GEP* are used as fine-tuning parameter in *OSPF* routing. Whereas *MA*s have been used in previous works for fine-tuning *OSPF* routing, *GEP* is not yet well-known and to the best of our knowledge has never been used to achieve destination-based TE.

- **Performance evaluation.** We applied *GEP* to find the routing paths for the traffic offered to a 23- and 30-node test networks and used the NS [5] simulator to compare the performance achieved by (1) *GEP*, (2) a memetic algorithm (*MA*) and (3) *OSPF* routing under different traffic profiles and service conditions. The results revealed the efficiency of *GEP* compared to the other routing schemes.

This paper is organised as follows. Section 2 presents the routing optimisation model and describes the routing algorithms considered. An application of the different algorithms to compute paths for the traffic offered to a 23- and 30-node networks is presented in section 3. We draw some preliminary conclusions and present guidelines for future work in section 4.

2 The Routing Optimisation Model

Our routing optimisation problem consists of finding a set of link weights $\mathcal{W} = \{w_1, \dots, w_L\}$ minimising the maximum link utilization of a network. These weights are used as routing metrics in *OSPF* path selection to find the routes followed by the traffic flows. The resulting *OSPF* routing metric is expressed by

$$L_p(OSPF) = \sum_{\ell \in p} w_\ell^\alpha / C_\ell^{1-\alpha} \quad (1)$$

where $L_p(x)$ and $L_\ell(x)$ are respectively the path and link metrics associated with the routing algorithm x for a link ℓ and a path p traversing ℓ , α is a calibration parameter

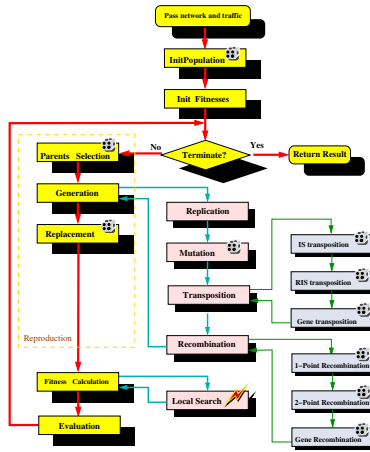


Fig. 1. “The GEP algorithm”

expressing the balance between *OSPF* routing and link weight optimisation and $w_\ell \in \mathcal{W}$ is the weight computed for the link ℓ . Note that the routing metric (1) yields (a) classical *OSPF* routing for $\alpha = 0$, (b) link weight optimisation for $\alpha = 1$ and (c) a balance between the two for $0 < \alpha < 1$.

We consider an evolutionary optimisation model where a *population* is expressed by a set of links represented by a vector \mathcal{W} ; each position ℓ holding a link weight w_ℓ . The **fitness function** is expressed by $(1/\max_\ell \mu_\ell)^P$ where μ_ℓ is the utilization of link ℓ , the power value is increased when the fitness is improved or decreased if the fitness is decreased. The **population acceptance probability** is increased upon fitness improvement or decreased when the fitness has worsened to ensure the survival of good populations over generations.

Our *GEP* algorithm uses a different genome representation than classical *MAs* and more operators than classical *GAs* as illustrated by Figure 1. *Fitness Calculation* uses translated link weights patterns to compute the fitness. *LocalSearch* deals with the TE computation to improve the genetic individuals fitness through hill-climbing. *Evaluation* evaluates the fitness function and modifies the probabilities and powers as suggested above. *ParentsSelection* randomly selects two parents in the population set and generates two offsprings. The *Transposition* procedure calls the *ISTransposition*, *RISTransposition*, and *GeneTransposition* procedures. These procedures achieve respectively (1) *IS transposition* with IS length of three elements, (2) *Root IS Transposition* with IS length of three elements, and (3) *Gene Transposition*. *Recombination* plays the same role as the crossover operator in *MAs* by gene crossover on generating offsprings. Three recombination methods were involved in our application: 1-point recombination, 2-point recombination, and Gene recombination. The *Replacement* procedure selects a population from the offsprings and original populations based on its fitness to give birth to a new generation. The *Mutation* procedure uses chromosome mutation to perform 1-point mutation on a chromosome assigning a randomly selected terminal to a randomly selected position of the chromosome. There are different other

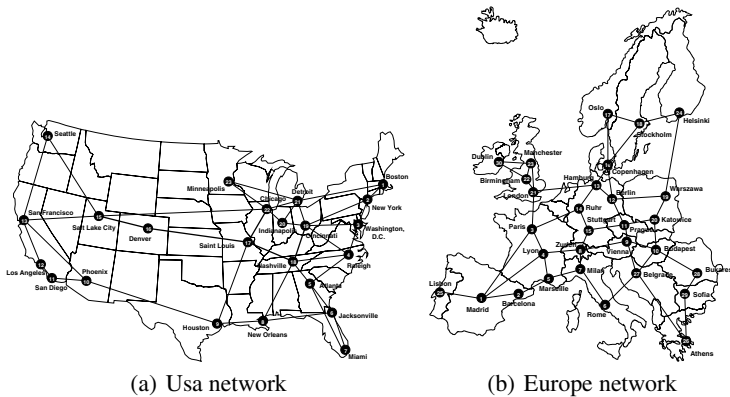


Fig. 2. “The test networks”

procedures associated with the *Fitness calculation* in *GEP* to allow the mapping of a population into a set of link weights pattern and translate a link weight into a chromosome consisting of a function of $+$ and terminals as usually expressed by the *Karva GEP* language [4]. These include (1) a procedure that translates a population set into a set of link weights (2) a procedure which translates a chromosome into a link weights pattern and (3) a procedure which reforms the link weights modified by *LocalSearch* into a chromosome.

3 An Implementation

This section presents simulation experiments conducted using a 23- and 30-node test networks to compare the performance of 3 different algorithms: (1) the *OSPF* algorithm herein referred to as *Normal* (2) a memetic algorithm (*MA*) and (3) the *GEP* algorithm. The 23- and 30-node networks illustrated by Figures 2 (a) and (b) are fictitious representations of a USA and a Europe networks where each node is considered as a potential ingress-egress pair. We considered several performance parameters some expressing the engineering efficiency in terms of bandwidth usage such as the throughput achieved by the network while others express the quality of service received by different flows in terms of routing delay.

We conducted simulation experiments to evaluate the efficiency of the different algorithms by combining in-house link weight calculation programs and NS packet-level simulation. The weight calculation process is performed by one of the two evolutionary algorithms (*MA* or *GEP*). In the NS simulation are performed (1) path selection (2) packet forwarding and (3) network monitoring to control congestion and recover from failure. We considered CBR traffic over the UDP protocol for all our NS simulation experiments.

- Efficiency under time-of-the day traffic fluctuation. We conducted a set of experiments to compare the throughput achieved by the different algorithms in a routing scenario where the traffic flows are randomly divided into ten groups, each group

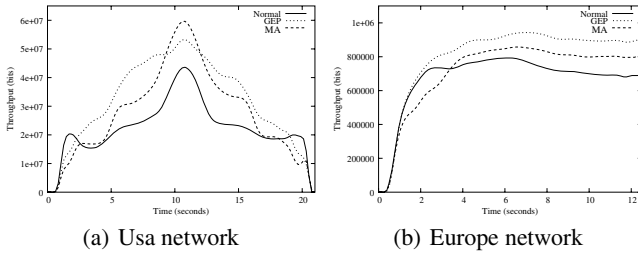


Fig. 3. “Time-of-the-day traffic fluctuation”

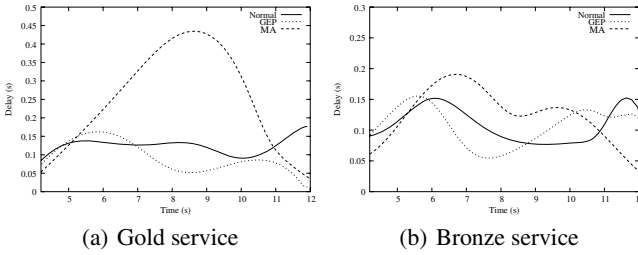


Fig. 4. “Bandwidth differentiated-services”

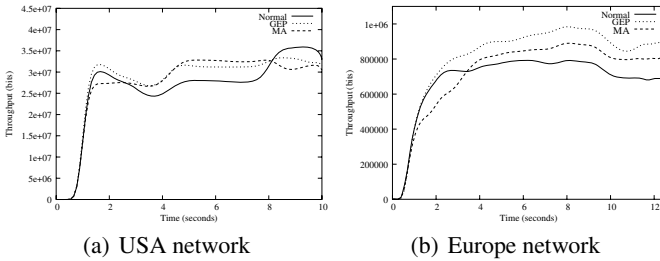


Fig. 5. “Recovery from failure”

being assigned to a time-zone with the expectation of simulating the time-of-the-day fluctuation of the traffic according to different zones. The total throughput received by all the traffic flows during the simulation period depicted by Figure 3 reveals that *GEP* achieves the highest throughput while *OSPF* achieves the least. The throughput achieved by *MA* lies between the two extremes. Per-group simulation revealed the same performance pattern but was not reported in this paper for space limitation.

- **Efficiency under bandwidth differentiated-services.** We conducted another set of experiments to evaluate the delay experienced by different flows in a bandwidth-differentiated services environment where different flows are allocated different grades of service depending on their bandwidth requirements: *Gold* service for high bandwidth demanding flows, *Silver* service for moderate bandwidth demanding flows and *Bronze* service for low bandwidth demanding flows. The results depicted by Figure 4 for *Gold* and *Bronze* services reveal that *GEP* and *OSPF* outperform *MA* in terms of delay.

Silver service results revealed the same performance pattern as *Gold* and *Bronze* but were not reported in this paper for space limitation.

- **Recovery capability upon failure.** We conducted another set of simulation experiments to evaluate the recovery capability of the three different algorithms when the highest capacity link (the fattest pipe) is failed for a short period of time and recovered later under two traffic conditions: (1) where the traffic is offered to the USA network from one time-zone and (2) where the traffic is offered to the Europe network from multiple time-zones. The results depicted by Figure 5 reveal that *GEP* outperforms *MA* and *OSPF* in terms of recovery from failure under the two failure scenarios by exhibiting higher throughput. Note that in Figure 5 (a), the fattest pipe of the USA network was failed from time $t = 2s$ to $t = 8s$ while the Europe's fattest pipe was failed from time $t = 3s$ to $t = 12s$.

4 Conclusion

This paper revisits the problem of fine-tuning IGP routing with the objective of finding link weights that maximise a network bandwidth usage. For the first time, we apply *GEP* to find a set of link weights that minimise the maximum link utilisation of a network and compare its performance with a memetic algorithm and *OSPF* routing. Preliminary results using packet-level simulation reveal that *GEP* outperforms both *OSPF* and *MA* in terms of throughput and delay under different traffic profiles and service differentiation scenarios. The evaluation of the performance achieved by *GEP* when using different performance parameters under different network topologies and traffic profiles is a direction for future research work. The assessment of the *GEP* paradigm to achieve different other TE needs such as flow-based TE is another direction for future research.

References

1. J. Moy, "OSPF Version 2", *Request For Comments*, <http://www.ietf.org/rfc/rfc1583.txt>, March 1994.
2. B. Fortz, M. Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights", *Proceedings of IEEE INFOCOM*, March 2000.
3. A. Riedl, "A Hybrid Genetic Algorithm for Routing Optimization in IP Networks Utilizing Bandwidth and Delay Metrics", *Proceedings of the IPOM2002 Conference*, October 2002.
4. C. Ferreira, "Gene Expression Programming: A New Adaptive Algorithm for solving problems." *Complex Systems*, 13(2):87-129, 2001.
5. *Network Simulator*, <http://www.isi.edu/nsnam/ns/>

Model and Convergence for the Combination of Genetic Algorithm and Ant Algorithm

Jianli Ding^{1,2}, Wansheng Tang¹, and Yufu Ning^{1,3}

¹ Institute of Systems Engineering, Tianjin University, Tianjin 300072, China
jianliding@yahoo.com.cn, {tang, ning}@tju.edu.cn

² Tianjin Key Lab for Advanced Signal Processing,
Civil Aviation University of China, Tianjin 300300, China

³ Department of Computer Science, Dezhou University, Dezhou 253023, China

Abstract. Although genetic algorithm (GA) has the ability to do quick and stochastic global search, it can't efficiently use the output information for systems. Ant algorithm (AA), on the other hand, is a parallel-proceed and distributive-forward system with a relatively slow speed for carrying out its solution. Incorporating GA and AA can improve their merits one for another. In this paper, the model and the method from the combination of GA and AA are proposed. The convergence of the method based on the Markov theory is analyzed. The experiment and analysis are conducted on the NP-hard problems for the cases of TSP30 (Travel Salesman Problem 30 cities) and CHN144 (China 144 cities). This work proves that the satisfactory solution sequence is monotonically decreasing and convergent. The results of simulations show that not only this combined algorithm is a step-by-step convergent process, but also its speed and effect of solving are quite satisfactory.

1 Introduction

The merits of GA are: the ability of searching in wide ranges, and independence of the problem fields; starting search from a group, and has the latent parallelism, comparison for several values, strong robustness; enlightening by evaluating functions during searching, simple process; iterating by the regulation of probability, and has the random city; flexibility, i.e., it is easy to combine with other algorithms.

However, GA has the following disadvantages [1]. Feedback information in the system is used insufficiently. A mass of redundant iterative is inevitable when the process solution is in a certain scope. Low efficiencies often occur as pursuing for exact solution. Ant algorithm is called ant colony optimization (ACO), it is also called ant system (AS) [2]. The merits of the ant system are: Firstly, its theory is a regulation of the positive feedback or an increasing-learning system, and through information elements updating continuously, it can converge to the optimal path. Secondly, it is a currently stochastic-optimization method. Thirdly, it is a distributing optimal algorithm. Finally, it is a globally optimal algorithm. However, the disadvantages of AA are: it lacks information pheromones at the initial stages, and the speed for initial solution is slow.

The combination of GA and AA unifies the fast speed of GA for global solutions and good precision of AA for solutions by the feedback information. The advantages reinforce one for another and expect to obtain dual successes.

2 The Combinative Model of GA and AA

2.1 Idea to Combine GA and AA

The basic idea of genetic algorithm-ant algorithm (GAAA) is to assimilate the merits of these two algorithms, to overcome their shortcomings, and to complement their predominance. It is excelled to AA in timing efficiency, and excelled to GA in solving efficiency. It is a new illuminative method in timing and solving efficiencies. Its basic idea is to adopt GA in the former process, and use sufficiently the rapidity, random city and global convergence of GA to bring a distribution on the problem of initial pheromone. The latter process adopts AA to use sufficiently the specialties of juxtaposition, feedback and solving efficiency of AA under the distribution condition of a certain initial pheromone.

2.2 Global Frame of GAAA

GAAA defines objective function and fitness-value function based on certain problems first; the genetic algorithm is carried out secondly to bring several optimal solutions; thirdly, based on the distribution of the initial pheromone brought by the optimal solution, ant algorithm is carried out; the output is the best solution finally. The global frame is shown in Fig. 1.

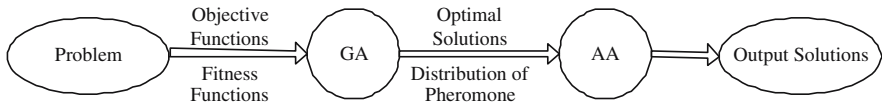


Fig. 1. Global frame of GAAA

2.3 Selection and Connection for the Model of GAAA

The details for selection and connection for the model of GAAA are described as follows:

- 1) The selection and definition for the model of GA in GAAA. The GA in GAAA is based on the theory and definition of elitist selecting GA. Its obvious merit is that the individual is selected as the one whose fitness value is the smallest in the processes of selection, crossover and mutation. Therefore, the eminent specialty of being monotone and non-increasing in colony sequential adaptive value, and it makes the process of inheritance evolving. That is: decimal code is used to confirm the fitness function of corresponding object; a dual of mating father chromosomes are selected based on the fitness-value function; Davis sequential-cross method is adopted; the method of reversion mutation is used, i.e., the fitness value increase and reserve after reversion, or the reversion is invalid.

- 2) The selection and improvement of AA in GAAA. The AA in GAAA is based on the model of ant loop and MMAS algorithm [3]; it is improved based on assimilation of the merits, that is:

The placement for the initial value of pheromone: the solving result of genetic algorithm is transformed into the initial value of pheromone.

Updating model of pheromone: pheromone is updated by using the model of ant loop, i.e., only the ant in shortest path within one loop can modify the increase of pheromone.

- 3) The connection for two algorithms in GAAA. The key of the connection of the two algorithms is the way to transform the result of GA into the pheromone of AA. MMAS sets the initial values of pheromones in the paths $\max \tau_{max}$. Here, certain pheromones in the path are obtained by GA. Therefore, the initial value of pheromone is set as:

$$\tau_s = \tau_c + \tau_g \tag{1}$$

where τ_c is a constant of pheromone given by the problem of solving solution, it is equivalent to τ_{min} in algorithm of MMAS, τ_g is the value of pheromone in the result of solving solution in GA.

3 Analysis for Simulations

GAAA is adapting to the experiments for the problems of the classical NP-hard problems of TSP30 case and CHN144 case. The iterative degrees are selected

Table 1. An approach process of GAAA optimization-solution

Process of GAAA	Optimal solution of TSP30			Optimal solution of CHN144		
	Max	Min	Average	Max	Min	Average
Initial random city	1500	1209	1318.8	88763	65739	76316
Selection operator	1198	1101	1139.1	65564	61278	63211
Crossover operator	1095	1005	1050.3	56405	54329	55323
Mutation operator	987	817	912.3	51994	51344	51508
Ant algorithm	452	424	433.7	31153	30351	30612

Table 2. A distribution of GAAA 30 optimization-solution value

The distribution of optimal value in TSP30							The distribution of optimal value in CHN144						
$(\alpha = 1, \beta = 2, \rho = 0.8, Q = 1000)$							$(\alpha = 2, \beta = 2, \rho = 0.9, Q = 100000)$						
436	430	431	439	426	437	433	30351	30585	30643	30730	30807	30928	31153
429	434	439	426	438	424	426	30810	30748	30617	30507	30358	30379	30357
425	446	449	426	424	443	434	30595	30551	30676	30760	30736	30788	30563
427	452	436	426	425	448	431	30354	30360	30431	30566	30739	30792	30743
440	430						30380	30356					

as 30 and 144, respectively. The initial values τ_c of pheromones in the paths in ant algorithm are set as 60 and 600, respectively. The values τ_g of pheromones of the result transformation of solving solution pass the path are added 2 and 20, respectively. The updates of locus are selected as $\rho = 0.8$, $Q = 1000$ and $\rho = 0.9$, $Q = 100000$, respectively. Tables 1-5 and Figs. 2-7 are the results of simulations.

Table 1 explains the process of approaching the optimal solution in GAAA from the initial optimal solution. The algorithm in this paper is a process of convergence step by step from respect of Max, Min, and Average.

Table 2 is the distribution of 30 optimal solutions in GAAA. It shows that the precision of solving solutions is very satisfactory. The GAAA can obtain the best solutions not only for the TSP problem of 30 cities ($d^* = 423.74$) but also for the problem of CHN144 in China ($d = 30351$). Moreover, a mass of satisfactory solutions approaching the best solution are also obtained. It is very important in practice.

Fig. 2 explains that the result of solving solution by GA in GAAA behaves in the placement of the initial values of pheromone. The above analysis shows that the algorithm in this paper is a process of convergence step by step. The average and the distribution indicate that the precision of solving solution is very good.

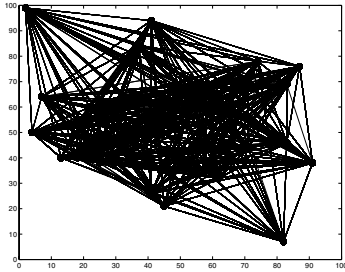


Fig. 2. The pheromone distribution brought by GA in GAAA (TSP30)

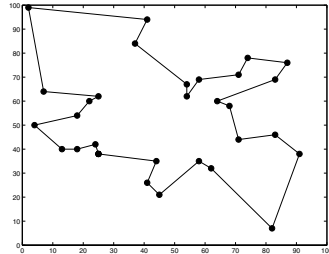


Fig. 3. The optimal route in GAAA (TSP30, $d^*=423.74$)

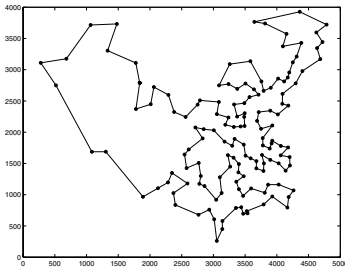


Fig. 4. The optimal route in GAAA (CHN144, $d = 30351$)

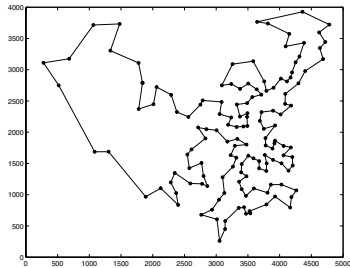


Fig. 5. The best result of stochastic iterative in GAAA (CHN144, $d = 30354$)

Table 3. Simulation results for GAAA

Shortest GAAA evolving generation								
α	β	ρ	route length	GA+AA	α	β	ρ	route length GA+AA
1	1	0.8	426.60	30+11	3	2	0.8	425.52 30+13
1	2	0.8	424.69	30+10	5	2	0.8	424.90 30+9
2	1	0.8	424.46	30+16	5	3	0.8	426.90 30+11
2	2	0.8	423.74	30+13	3	5	0.8	429.79 30+9
2	3	0.8	424.67	30+21	5	5	0.8	430.13 30+10
3	3	0.8	425.65	30+19				

Table 4. Simulation results for AA

α	β	ρ	Shortest-route generation	
2	2	0.5	424.8	350
2	2	0.9	427.	344
1	2	0.5	423.7	342
5	2	0.9	430.5	338
5	2	0.5	445.	347

Table 5. GA and SA

Algorithms	Average generation
GA	404
SA	1018
GASA	554
(Parallel P) SA	1012

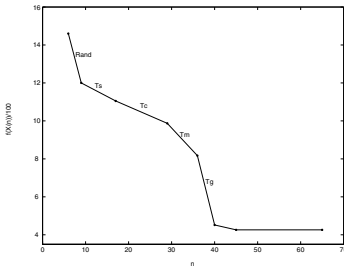


Fig. 6. GAAA solving process (TSP30)

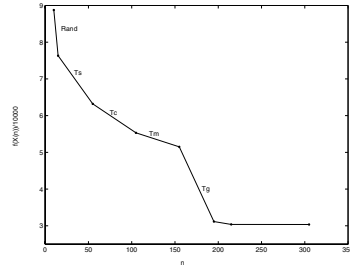


Fig. 7. GAAA solving process (CHN144)

Figs. 3 and 4 are the optimal solutions by using GAAA. They consist with the best solution obtained at present.

The solutions in Fig. 5 approach the optimal solution. They are the values readily getting into the local optimization. Since the algorithm in this paper adopts stochastic colony during inheriting, it can be avoided to getting into the local optimization.

Table 3 is the result of simulation in this paper. In contract to the basic AA in Table 4, it not only decreases the evolving generation, but also repeats the iterative, and it can avoid getting into local optimization. Accordingly, the precision of solving can be increased significantly. In contract to GA in Table 5 and simulation anneal, the efficiency of solving in GAAA is increased largely.

Figs. 6 and 7 are the processes of once time stochastic solving in GAAA to the problems of TSP30 and CHN144. They not only show the monotone decrease, but also describe distinctly the effect of the solving process in each performance.

Where, the initial-value phase of T_g is very abrupt. It is manifested that ant algorithm approaches rapidly to its optimal solution by using its pheromones after the performances of inheritance selection, crossover and mutation. This overcomes the shortcoming of slowly solving due to lack of initial pheromone in ant algorithm. From the iterative process, TSP30 can obtain the satisfactory solution in about 40 reiterations, and CHN144 can obtain the satisfactory solution in about 200 reiterations. Accordingly, a high efficiency of solving is shown.

4 Analysis of Convergence

Define $S^N = \{\vec{X} = (X_1, X_2, \dots, X_N), X_i \in S(i \leq N)\}$ as the colony (optimal solution) space, $S^2 = \{(X_1, X_2), X_1, X_2 \in S\}$ is called as father space; S is individual space; X_i is individual, its length is l called as chain length; N is colony (optimal solution) dimension, n is the iterative degree of superior selection of ant algorithm; m is the moving period of ant algorithm; P is distributive probability on S^N . Here, GAAA can be taken as the expansion of genetic algorithm with superior selection. It has four formulations of processes: selection operator T_s , crossover operator T_c , mutation operator T_m , and pheromone operator T_g . The basic definitions can be found in [4].

Theorem 1. *Optimal solution sequence $\{\vec{X}(n); n \geq 0\}$ in GAAA is a finite homogenous Markov chain.*

Proof. From lemma [4], the genetic algorithm of superior selection $\vec{X}(n + 1)$ is only correlative with $\vec{X}(n)$ and irrespective non-correlative with n . From lemma [5], ant system $\{\vec{\tau}(m + 1), \vec{X}(m + 1), f^*(m + 1)\}$ is only correlative with $\{\vec{\tau}(m), \vec{X}(m), f^*(m)\}$ and non-correlative with cycle period m . At the same time, from the model of ant loop, optimal ant loop can only be updated in once pheromone; it has the characteristic of genetic algorithm of superior selection. Therefore, we can select:

$$\vec{X}(n+1) = (X_i(n+1) = T_g^i \circ T_m^i \circ T_c^i \circ T_s^i(\vec{X}(n)); (i \leq N-1); X_N(n+1) = X_{i_0}(n)) \tag{2}$$

where $i_0 = \arg \min_j \{f(X_j(n))\}$ denotes that the individual is $X_j(n)$ under the condition of making $f(X_j(n))$ minimum, and transformation matrix is:

$$\begin{aligned}
 P \{ \vec{X}, \vec{Y} \} &= P \{ \vec{X}(n + 1) = \vec{Y} / X(n) = \vec{X} \} \\
 &= \begin{cases} \prod_{k=1}^N P \{ T(\vec{X}(n))_k = Y_k \}, & \exists i_0 \in M(\vec{X}), \text{ makes } Y_N = X_{i_0} \\ 0, & \text{else} \end{cases} \tag{3}
 \end{aligned}$$

Where, $M(\vec{X}) = \{i; f(X_i) = \min\{f(X_j)\}\}$, then

$$T(\vec{X}(n)) = T_g^i(T_m^i(T_c^i(T_s^i(\vec{X}(n)))))) = T_g^i \circ T_m^i \circ T_c^i \circ T_s^i(\vec{X}(n)). \tag{4}$$

It is manifested from the above equation that $\vec{X}(n+1)$ is only correlative with $\vec{X}(n)$ and non-correlative with n . Optimal solution sequence $\{\vec{X}(n); n \geq 0\}$ of GAAA is finite homogenous Markov chain.

Theorem 2. *The satisfactory value sequence of optimal solution of the sequence of Markov chain in GAAA is monotone non-increased, i.e., for an arbitrary $n \geq 0$, such that $F(\vec{X}(n)) \geq F(\vec{X}(n+1))$.*

Proof. At first, because genetic algorithm of superior selection is used in GAAA, $i_0 = \arg \min_j \{f(X_j(n))\}$, we have

$$X_N(n+1) = X_{i_0}(n), F(\vec{X}(n)) \geq F(X_N(n+1)) \geq F(\vec{X}(n+1)). \tag{5}$$

Secondly, the algorithm of ant loop is used in GAAA, $i_0 = \arg \min_j \{f(X_j(m))\}$, therefore, we can get

$$X_N(m+1) = X_{i_0}(m), F(\vec{X}(m)) \geq F(X_N(m+1)) \geq F(\vec{X}(m+1)). \tag{6}$$

The initial distribution of the algorithm of ant loop is the result of genetic algorithm of superior selection; it has the successor of superior. Therefore, for an arbitrary $n \geq 0$, such that $F(\vec{X}(n)) \geq F(\vec{X}(n+1))$, i.e., the satisfactory value sequence of optimal solution of Markov chain in GAAA is monotone non-increased.

Theorem 3. *Markov sequence of optimal solution in GAAA converges to the subset B_0^* of the set B of satisfactory solutions $B_0^* = \{\vec{Y} = (Y_1, Y_2, \dots, Y_N); Y_N \in B\}$, i.e.,*

$$\lim_{n \rightarrow \infty} P\{\vec{X}(n) \in B_0^* / \vec{X}(0) = \vec{X}_0\} = 1. \tag{7}$$

Proof. Suppose X is the only minimum solution of $f(X)$. From Theorem 1 and Lemma [4], we have:

- 1) When $\vec{X}, \vec{Y} \in B_0^*, P\{X, Y\} > 0, P\{Y, X\} > 0$, i.e., $\vec{X} \leftrightarrow \vec{Y}$;
- 2) When $\vec{X} \in B_0^*, \vec{Y} \notin B_0^*, P\{X, Y\} = 0$, i.e., $\vec{X} \rightarrow \vec{Y}$.

Therefore, B_0^* is the non-period divisible closed set, $S^N \setminus B_0^*$ is the state set.

$$\lim_{n \rightarrow \infty} P\{\vec{X}(n) = \vec{Y} / \vec{X}(0) = \vec{X}_0\} = \begin{cases} \pi(\vec{Y}), & \vec{Y} \in B_0^* \\ 0, & \vec{Y} \notin B_0^* \end{cases} \tag{8}$$

Then, we have

$$\lim_{n \rightarrow \infty} P\{\vec{X}(n) \in B_0^* / \vec{X}(0) = \vec{X}_0\} = 1. \tag{9}$$

Corollary 1. *If GAAA $\{\vec{X}(n); n > 0\}$ is convergent for an arbitrary set of satisfactory solution, it must converge to the set of global optimal solution.*

5 Conclusion

The combination of GA and AA can compliment for each other, and can obtain an excellent efficiency in capabilities of optimization and time. Through the analysis of simulation, the improved algorithm of ant loop approaches the optimal solution after forming pheromone distribution by GA of superior selection in GAAA. At the same time, a mass of sightless experiments can be avoided. The analysis of theory manifests that the sequence of satisfactory solution in GAAA is a monotone decreasing Markov process and must be convergent. This work demonstrates that the combination of methods based on intelligent optimization is one of the important methods of solving large-scale complicated NP problems in the future.

Acknowledgments

This work was supported by the National Natural Science Foundation of China Grant No. 70471049 and China Postdoctoral Science Foundation No. 2004036138.

References

1. Li, M., Xu, B., Kou, J.: On the combination of genetic algorithms and neural networks. *Systems-Engineering Theory & Practice* (in Chinese). **19** (1999) 65-69
2. Dorigo, M., Bonabeau, E., Theraulaz, G.: Ant algorithms and stigmergy. *Future Generation Computer System*. **16** (2000) 851-871
3. Stutzle, T., Hoos, H. H.: Max-min ant system. *Future Generation Computer System*. **16** (2000) 889-914
4. Zhang, W., Liang, Y.: *Mathematical Foundation of Genetic Algorithms* (in Chinese). Xi'an Jiantong University Press, Xi'an (2000)
5. Gutjahr, W. J.: A graph-based ant system and its convergence. *Future Generation Computer System*. **16** (2000) 873-888

Moving Block Sequence and Organizational Evolutionary Algorithm for General Floorplanning*

Jing Liu, Weicai Zhong, and Licheng Jiao

Institute of Intelligent Information Processing, Xidian University,
No.2 South TaiBai Road, Xi'an 710071, China
neouma@163.com

Abstract. A new nonslicing floorplan representation, moving block sequence (MBS), is first proposed. The MBS is suitable for evolutionary algorithms since no extra constraints are exerted on the solution space. Furthermore, an organizational evolutionary algorithm is designed with the intrinsic properties of MBS in mind, called MBS-OEA. In experiments, 21 benchmarks from MCNC and GSRC are used to test the performance of MBS-OEA. The number of blocks in these benchmarks varies from 9 to 300. Comparisons are also made between MBS-OEA and some well-designed existing algorithms. The experimental results show that MBS-OEA can find high quality solutions even for the problems with 300 blocks. Therefore, MBS-OEA is competent for solving large scale problems.

1 Introduction

Floorplanning is an essential step in the hierarchical physical design of deep sub-micron very large scale integration (VLSI) circuits. Stochastic optimization algorithms employ methods of perturbing the floorplan and searching for better solutions, and has become the most popular method in solving floorplanning. For floorplanning problems, the most of stochastic optimization algorithms in common use is simulated annealing (SA) and evolutionary algorithms (EAs). However, SA is much more popular than EAs. We think one reason for less application of EAs in the nonslicing floorplan is nonslicing floorplan representations are often very complex, and the solution spaces are often constrained. Therefore, it is important to devise new nonslicing floorplan representations that no extra constraints are exerted on the solution spaces. In this paper, we will present such a representation, the moving block sequence (MBS).

An MBS is composed of two tuples that denote the permutation of block names and the initial positions of blocks. Each block is placed on the chip in the order it occurs in the permutation. Because there are four choices for the initial position of each block, no extra constraints are exerted on the solution space, and it is easy to devise effective crossover operators for EAs on such a solution space. Furthermore, an organizational evolutionary algorithm is designed with the intrinsic properties of the MBS in mind, called MBS-OEA. Organization evolutionary algorithm is a new type

* This research is supported by the National Natural Science Foundation of China under Grant 60133010.

of EA proposed in our previous work, and has been successfully applied to classification problems [1]. In MBS-OEA, three new operators are designed. In the experiments, benchmarks with 9 to 300 blocks from MCNC and GSRC are used to test the performance of MBS-OEA, with good performances result.

2 Moving Block Sequence Representation

Given a floorplanning problem with n blocks, $B=\{B_0, B_1, \dots, B_{n-1}\}$. Let $F(B)$ stand for a floorplan of the n blocks and $\text{Cost}(F(B))$ be the predefined cost metric. In the following text, the bottom side and the left side of the chip are fixed to form the X-coordinate and Y-coordinate. All blocks can only move in the first quadrant. Before a block starts to move on the chip, it must be placed on an initial position.

Definition 1: The block B will move on the chip. Let (x^{LB}, y^{LB}) be the coordinate of the left-bottom corner of B . Let $width$ and $height$ be the width and the height of B . Let Box be the minimum bounding rectangle of the blocks having been placed on the chip and (x^{BoxRT}, y^{BoxRT}) the coordinate of the right-top corner of Box . The coordinate of the left-bottom corner of Box always is $(0, 0)$. Then there are four **initial positions** for B . The first one is placing B on the left-top side over Box , numbered as 0, the second one is placing B on the right-bottom side beside Box , numbered as 1, the third one is placing B on the top-right side over Box , numbered as 2, and the fourth one is placing B on the right-top side beside Box , numbered as 3. Namely,

$$\left\{ \begin{array}{ll} \text{initial position 0:} & x^{LB} \leftarrow 0, \quad y^{LB} \leftarrow y^{BoxRT} \\ \text{initial position 1:} & x^{LB} \leftarrow x^{BoxRT}, \quad y^{LB} \leftarrow 0 \\ \text{initial position 2:} & x^{LB} \leftarrow \max(0, x^{BoxRT} - width), \quad y^{LB} \leftarrow y^{BoxRT} \\ \text{initial position 3:} & x^{LB} \leftarrow x^{BoxRT}, \quad y^{LB} \leftarrow \max(0, y^{BoxRT} - height) \end{array} \right. \quad (1)$$

Definition 2: There are four **moving rules** corresponding to four initial positions. **Moving rule 0:** When a block starts to move on initial position 0, the block can only move to bottom vertically until cannot move any more; **Moving rule 1:** When a block starts to move on initial position 1, the block can only move to left horizontally until cannot move any more; **Moving rule 2:** When a block starts to move on initial position 2, the block can repeatedly move to bottom and to left until cannot move any more, and moving to bottom is prior to moving to left; **Moving rule 3:** When a block starts to move on initial position 3, the block can repeatedly move to left and to bottom until cannot move any more, and moving to left is prior to moving to bottom.

Definition 3: A **Moving Block Sequence** (MBS) is composed of two tuples. One denotes the permutation of all block names π , and another one denotes the initial positions for each block IP , that is,

$$MBS=(\pi, IP) \text{ and } \pi=(\pi_0, \pi_1, \dots, \pi_{n-1}) \text{ and } IP=(IP_1, IP_2, \dots, IP_{n-1}) \quad (2)$$

where π_i ($0 \leq i \leq n-1$) denotes a block among $\{B_0, B_1, \dots, B_{n-1}\}$, and $IP_j \in \{0, 1, 2, 3\}$ ($1 \leq j \leq n-1$) denotes the initial position for π_j . The first block, π_0 , is placed on the left-bottom corner of the first quadrant.

3 Algorithm Transforming an MBS to a Floorplan

The information of a rectangular block is recorded in the following structure,

B = record
 x^{LB}, y^{LB} : the X and Y-coordinate of the left-bottom corner of **B**;
width, height: the width and height of **B**;
end.

The following two structures, $e^{//X}$ and $e^{//Y}$, are designed to record the information of edges paralleling the X-coordinate and the Y-coordinate, respectively,

$e^{//X} = \text{record}$ $e^{//Y} = \text{record}$
 x^L : the X-coordinate of the left point; x : the X-coordinate of the edge;
 x^R : the X-coordinate of the right point; y^B : the Y-coordinate of the bottom point;
 y : the Y-coordinate of the edge; y^T : the Y-coordinate of the top point;
end. **end.**

Definition 4: Let $a^{//X}$ and $b^{//X}$ be two edges paralleling the X-coordinate. If $a^{//X}$ and $b^{//X}$ satisfy (3), $a^{//X}$ **top-overlaps** $b^{//X}$; otherwise, $a^{//X}$ **un-top-overlaps** $b^{//X}$. Let $a^{//Y}$ and $b^{//Y}$ be two edges paralleling the Y-coordinate. If $a^{//Y}$ and $b^{//Y}$ satisfy (4), $a^{//Y}$ **right-overlaps** $b^{//Y}$; otherwise, $a^{//Y}$ **un-right-overlaps** $b^{//Y}$.

$$(a^{//X}(y) \geq b^{//X}(y)) \text{ and } (a^{//X}(x^R) > b^{//X}(x^L)) \text{ and } (a^{//X}(x^L) < b^{//X}(x^R)) \quad (3)$$

$$(a^{//Y}(x) \geq b^{//Y}(x)) \text{ and } (a^{//Y}(y^T) > b^{//Y}(y^B)) \text{ and } (a^{//Y}(y^B) < b^{//Y}(y^T)) \quad (4)$$

Algorithm 1. Algorithm transforming an MBS to a floorplan

Input: **MBS**: $MBS = (\pi, IP) \in \mathcal{MBS}$, π is a permutation of $\{B_0, B_1, \dots, B_{n-1}\}$;

Output: $\mathcal{F}(\mathcal{B})$, where $\mathcal{B} = \{B_0, B_1, \dots, B_{n-1}\}$;

$BtoT^{//X}$ and $LtoR^{//Y}$ record the top edges from bottom to top and the right edges from left to right of the placed blocks. Let $BtoT_i^{//X}$ and $LtoR_i^{//Y}$ be the i th element in $BtoT^{//X}$ and $LtoR^{//Y}$. $e^{//X} \leftarrow (a, b, c)$ stands for assigning a , b , and c to $e^{//X}(x^L)$, $e^{//X}(x^R)$, and $e^{//X}(y)$, respectively, and the same to $e^{//Y} \leftarrow (a, b, c)$.

begin

$\pi_0(x^{LB}) := 0$; $\pi_0(y^{LB}) := 0$; $x^{BoxRT} := \pi_0(\text{width})$; $y^{BoxRT} := \pi_0(\text{height})$;

$BtoT_0^{//X} \leftarrow (0, \pi_0(\text{width}), \pi_0(\text{height}))$; $LtoR_0^{//Y} \leftarrow (\pi_0(\text{width}), 0, \pi_0(\text{height}))$;

for $i := 1$ **to** $n-1$ **do**

begin

case (IP_i) **of**

0: $\pi_i(x^{LB}) := 0$; $\pi_i(y^{LB}) := y^{BoxRT}$; $e^{//X} \leftarrow (0, \pi_i(\text{width}), y^{BoxRT})$; $j := |BtoT^{//X}|-1$;
while $(e^{//X}$ un-top-overlaps $BtoT_j^{//X})$ **do** $j := j-1$;

$\pi_i(y^{LB}) := BtoT_j^{//X}(y)$;

1: $\pi_i(x^{LB}) := x^{BoxRT}$; $\pi_i(y^{LB}) := 0$; $e^{//Y} \leftarrow (x^{BoxRT}, 0, \pi_i(\text{height}))$; $j := |LtoR^{//Y}|-1$;
while $(e^{//Y}$ un-right-overlaps $LtoR_j^{//Y})$ **do** $j := j-1$;

$\pi_i(x^{LB}) := LtoR_j^{lly}(x)$;
 2: $\pi_i(x^{LB}) := \max(0, x^{BoxRT} - \pi_i(width))$; $\pi_i(y^{LB}) := y^{BoxRT}$;
repeat
 $e^{lly} \leftarrow (\pi_i(x^{LB}), \pi_i(x^{LB}) + \pi_i(width), \pi_i(y^{LB}))$; $j := |BtoT^{lly}| - 1$;
 while $((j \geq 0) \text{ and } (e^{lly} \text{ un-top-overlaps } BtoT_j^{lly}))$ **do** $j := j - 1$;
 if $(j \geq 0)$ **then**
 begin
 $\pi_i(y^{LB}) := BtoT_j^{lly}(y)$; $CoverLeftX := BtoT_j^{lly}(x^L)$;
 $j := j - 1$;
 end
 else begin $\pi_i(y^{LB}) := 0$; $CoverLeftX := 0$; **end**;
 $e^{lly} \leftarrow (\pi_i(x^{LB}), \pi_i(x^{LB}) + \pi_i(width), \pi_i(y^{LB}))$;
 while $((j \geq 0) \text{ and } (BtoT_j^{lly}(y) = \pi_i(y^{LB})))$ **do**
 begin
 if $((e^{lly} \text{ top-overlaps } BtoT_j^{lly}) \text{ and } (BtoT_j^{lly}(x^L) < CoverLeftX))$ **then** $CoverLeftX := BtoT_j^{lly}(x^L)$;
 $j := j - 1$;
 end;
 $e^{lly} \leftarrow (\pi_i(x^{LB}), \pi_i(y^{LB}), \pi_i(y^{LB}) + \pi_i(height))$; $j := |LtoR^{lly}| - 1$;
 while $((j \geq 0) \text{ and } (e^{lly} \text{ un-right-overlaps } LtoR_j^{lly}))$ **do** $j := j - 1$;
 if $(j \geq 0)$ **then** $CoverRightX := LtoR_j^{lly}(x)$
 else $CoverRightX := 0$;
 if $(CoverLeftX - CoverRightX < \pi_i(width))$ **then**
 begin $\pi_i(x^{LB}) := CoverRightX$; $CanMove := false$; **end**
 else begin $\pi_i(x^{LB}) := CoverLeftX - \pi_i(width)$; $CanMove := true$;
 end;
 until $(\text{not } CanMove)$;
 3: This part is similar to Case 2;
end; //End of **case**
 Updating $x^{BoxRT}, y^{BoxRT}, |BtoT^{lly}|$, and $|LtoR^{lly}|$;
end; //End of **for i**
 $Cost(\mathcal{F}(\mathcal{B})) := x^{BoxRT} \times y^{BoxRT}$;
end.

4 Organizational Evolutionary Algorithm Based on Moving Block Sequence

4.1 Definition of Organization for Floorplanning

Since an organization is composed of members, members are first defined as follows,

Definition 5: A **Member** corresponds to an MBS, and is represented as,

Member = record

- MBS*: the MBS corresponding to this member;
- S*: used to determine the direction of each block;
- $\mathcal{F}(\mathcal{B})$: the floorplan determined by *S* and *MBS*;
- Cost*: $Cost(\mathcal{F}(\mathcal{B}))$;
- Trained*: used in evolutionary operators;

end.

Definition 6: *S* of each member is an ordered set, which the *i*th element in *S* corresponding to the *i*th block. Let $S=\{s_0, s_1, \dots, s_{n-1}\}$, then $s_i \in \{0, 1\}$, which represents the *i*th block needs to rotate 0 or 90 degrees clockwise, respectively.

Definition 7: An organization, *Org*, is an ordered set, which is composed of more than one member. The first member is best whereas the last is worst.

4.2 Evolutionary Operators for Organizations

Splitting operator: This operator divides a parent organization into two child organizations according to some conditions. The condition to split an organization is

$$(|Org| > Max_{OS}) \text{ or } \left\{ (2 \leq |Org| \leq Max_{OS}) \text{ and } \left(U(0, 1) < |Org| / N_{Member} \right) \right\} \quad (5)$$

Max_{OS} is the maximum number of members in an organization. $U(0, 1)$ is a random real number uniformly distributed in 0 to 1. N_{member} is the summation of the number of members of all organizations in a population. Let Org^P be a parent organization. If Org^P satisfies (5), then Org^P will be split into two child organizations as follows: $|Org^P|/3 - 2|Org^P|/3$ members are first selected from Org^P to form one child organization, Org^{C_1} , and the remainder forms another child organization, Org^{C_2} .

Annexing operator: Let two parent organizations are $Org^{P_1} = \{Member_0^{P_1}, Member_1^{P_1}, \dots, Member_{m_1-1}^{P_1}\}$ and $Org^{P_2} = \{Member_0^{P_2}, Member_1^{P_2}, \dots, Member_{m_2-1}^{P_2}\}$. $Member_0^{P_1} (Cost)$ is compared with $Member_0^{P_2} (Cost)$, and the better one wins.

Without loss of generality, let $Member_0^{P_1}$ win. Then Org^{P_2} is annexed by Org^{P_1} , which means Org^{P_1} increases m_2 new members, and Org^{P_2} is washed out from the population. There are two methods to generate the new members, labeled as Generator₁(*Member*₁, *Member*₂, *Member*_{new}) and Generator₂(*Member*, *Member*_{new}).

In Generator₁(*Member*₁, *Member*₂, *Member*_{new}), let $Member_1(MBS) = (\pi^1, IP^1)$, $Member_2(S) = \{s_0^1, s_1^1, \dots, s_{n-1}^1\}$, $Member_2(MBS) = (\pi^2, IP^2)$, $Member_2(S) = \{s_0^2, s_1^2, \dots, s_{n-1}^2\}$, $Member_{new}(MBS) = (\pi^{new}, IP^{new})$, and $Member_{new}(S) = \{s_0^{new}, s_1^{new}, \dots, s_{n-1}^{new}\}$. Then,

$$\pi^{new} = \pi^1 \text{ and } IP_i^{new} = \begin{cases} IP_m^1 & \text{if } (\pi_m^1 = \pi_l^2) \text{ and } U(2)=0 \\ IP_l^2 & \text{if } (\pi_m^1 = \pi_l^2) \text{ and } U(2)=1 \end{cases} \quad (6)$$

$$s_i^{new} = \begin{cases} s_i^1 & \text{if } U(2)=0 \\ s_i^2 & \text{if } U(2)=1 \end{cases} \quad (7)$$

Where $U(x)$ is uniformly distributed in 0 to $x-1$. Finally, performing (8),

if ($Member_2(Cost) < Member_{new}(Cost)$) then $Member_{new} = Member_2$ (8)

In $Generator_2(Member, Member_{new})$, let $Member(MBS)=(\pi, IP)$, $Member(S)=(s_0, s_1, \dots, s_{n-1})$, $Member_{new}(MBS)=(\pi^{new}, IP^{new})$, and $Member_{new}(S) = \{s_0^{new}, s_1^{new}, \dots, s_{n-1}^{new}\}$. First, assign $Member$ to $Member_{new}$, and then for each element in π^{new} , IP^{new} , and $Member_{new}(S)$, do the following operations if $U(0, 1) < 1/n$:

$$\begin{cases} \pi_i^{new} = \pi_j^{new} & \text{where } j = U(n) \text{ and } (i \neq j) \\ IP_i^{new} = U(4) & \text{where } U(4) \neq IP_i^{new} \\ s_i^{new} = 1 - s_i^{new} \end{cases} \quad (9)$$

The new m_2 members of Org^P are determined as follows,

$$\begin{cases} Generator_1(Member_0^P, Member_i^P, Member_i^C) & 0 \leq i \leq \lfloor m_2/2 \rfloor \\ Generator_2(Member_{i-\lfloor m_2/2 \rfloor \bmod m_1}^P, Member_i^C) & \lfloor m_2/2 \rfloor < i < m_2 \end{cases} \quad (10)$$

Finally, set $Member_0^C(Cost)$, $Member_1^C(Cost)$, ..., $Member_{m_2-1}^C(Cost)$ to *false*.

Training operator: Let this operator be performed on Org , and $Member$ is the member selected from Org . Then $Member$ is the first member of Org whose *Trained* is *false*. If *Trained* of all members is *true*, the first member of Org is assigned to $Member$. Let $Member(MBS)=(\pi, IP)$ and $Member(S) = \{s_0, s_1, \dots, s_{n-1}\}$. The process of training consists three parts. The three parts is performed on $Member$ randomly.

Changing the permutation of the block names: For π_i ($0 \leq i < n$), select another element in π to exchange with it, and such an exchange is repeated five times. After each exchange, $Member(Cost)$ is computed out. If it is better than before, then the exchange is accepted; otherwise, the exchange is given up.

Changing the initial positions: Assign another three initial positions to each IP_i ($1 \leq i < n$). If an initial position decreases $Member(Cost)$, then it is accepted.

Changing the directions of the blocks: For each blocks, assign another value to s_i , and compute $Member(Cost)$. If $Member(Cost)$ is better than before, this value is accepted; otherwise, it is given up.

After all three parts are performed, $Member(Trained)$ is set to *true*.

4.3 Implementation of Algorithm

The details are shown in Algorithm 2.

Algorithm 2. Organizational evolutionary algorithm based on moving block sequence

Input: Max_{OS}, N_{Member} : Predefined parameters;

Output: $\mathcal{F}(\mathcal{B})$, where $\mathcal{B} = \{B_0, B_1, \dots, B_{n-1}\}$;

Org_i^t is the i th organization in the population of the t th generation, P^t .

begin

 Generating N_{Member} members randomly;

 Computing the cost of each member, and setting its *Trained* to *false*;

 Using each member to form an organization, therefore, the initial population, P^0 , consists of N_{Member} organizations.

$t := 0$; $CanTrain := true$;

```

repeat
  for  $i := 0$  to  $|P^i|-1$  do
    begin
      if ( $Org_i^i$  satisfies (5)) then
        begin
          Splitting  $Org_i^i$  into two organizations,  $Org^{C_1}$  and  $Org^{C_2}$ , according to the splitting operator;
          Adding  $Org^{C_1}$  and  $Org^{C_2}$  into  $P^{i+1}$ ;
        end;
        Deleting  $Org_i^i$  from  $P^i$ ;
      end;
    repeat
      Selecting two organizations,  $Org_1$  and  $Org_2$ , from  $P^i$  randomly;
      Performing the annexing operator on  $Org_1$  and  $Org_2$  to generate  $Org^C$ ;
      Adding  $Org^C$  into  $P^{i+1}$ , and deleting  $Org_1$  and  $Org_2$  from  $P^i$ ;
    until ( $|P^i| < 2$ );
    if ( $|P^i|=1$ ) then Add  $Org_0^i$  into  $P^{i+1}$ ;
    if ( $CanTrain$ ) then
      begin
         $CanTrain := false$ ;
        for  $i := 0$  to  $|P^i|-1$  do
          begin
            Performing the training operator on  $Org_i^i$ ;
            if (the cost of the select member from  $Org_i^i$  decreased) then
               $CanTrain := true$ ;
            end;
          end;
        end;
         $t := t+1$ ;
      until (Stop conditions are satisfied);
      Setting  $\mathcal{F}(\mathcal{B})$  of the best members among all organizations in  $P^i$  to  $\mathcal{F}(\mathcal{B})$ ;
    end.

```

5 Experimental Studies

21 floorplanning problems come from MCNC and GSRC benchmarks [2] are used to test the performance of MBS-OEA. The parameters of MBS-OEA are set as follows: $Max_{OS}=20$ and $N_{Member}=1000$. The number of evolutionary generations is set to 100 when $n \leq 100$, 2000 when $n=200$, 3000 when $n=300$. MBS-OEA runs 10 times independently on each problem. The experimental results of MBS-OEA are shown in Table 1. MBS-OEA obtains a better performance in both the best and the average value of Area Ratio. Besides, MBS-OEA also finds high quality floorplans for the problems with 200 or 300 blocks, which illustrates MBS-OEA is competent for solving large scale problems.

Table 1. The experimental results of MBS-OEA

Problem name	Area		Area Ratio (%)		Average times (s)
	Best	Average	Best	Average	
apte	46.924848	46.924848	99.23	99.23	0.5
xerox	19.795608	19.879554	97.75	97.34	1.1
hp	8.947008	8.947008	98.70	98.70	1.7
ami33	1.175363	1.206013	98.39	95.91	45.0
ami49	36.350160	37.040805	97.51	95.70	173.2
n10a	22.6092	22.8031	98.05	97.22	0.9
n10b	22.5585	22.7450	98.05	97.24	1.2
n10c	23.1804	23.3698	98.69	97.89	1.1
n30a	21.4600	21.7734	97.20	95.81	25.6
n30b	20.1150	20.4226	98.33	96.85	29.4
n30c	22.8242	23.2908	97.49	95.55	25.5
n50a	20.4140	20.8561	97.28	95.22	123.2
n50b	20.9271	21.2932	97.03	95.37	121.0
n50c	20.7564	21.2032	97.08	95.05	137.0
n100a	18.9832	19.3208	94.56	92.91	1706.1
n100b	16.9158	17.3306	94.66	92.40	1065.4
n100c	18.2648	18.5742	94.15	92.59	1319.0
n200a	19.2384	19.5258	91.33	89.99	12114.1
n200b	19.0554	19.3817	91.62	90.09	14217.2
n200c	18.7600	18.9976	90.69	89.56	11210.7
n300a	30.2412	30.7492	90.33	88.84	80984.1

Table 2. The comparison between MBS-OEA and 9 existing algorithms

	apte		xerox		hp		ami33		ami49	
	Area	Time (s)	Area	Time (s)	Area	Time (s)	Area	Time (s)	Area	Time (s)
O-Tree	47.1	38	20.1	118	9.21	57	1.25	1430	37.6	7428
E O-Tree	46.92	11	20.21	38	9.16	19	1.24	118	37.73	406
B ⁻ -Tree	46.92	7	19.83	25	8.947	55	1.27	3417	36.80	4752
TCG	46.92	1	19.83	18	8.947	20	1.20	306	36.77	434
TCG-S	46.92	1	19.80	5	8.947	7	1.185	84	36.40	369
Fast-SP	46.92	1	19.80	14	8.947	6	1.205	20	36.5	31
E Q-seq	46.92	0.35	19.93	3.6	9.03	3.5	1.194	40	36.75	57
TBS	47.44	0.86	19.78	1.3	8.48 [*]	0.76	1.196	1.26	36.89	2.55
CS	46.92	1	19.83	54	8.947	6	1.18	530	36.28	851
MBS-OEA	46.92	0.5	19.80	1.1	8.947	1.7	1.175	45	36.35	173.2

* Smaller than the area of all blocks, a wrong solution

Table 2 compares the experimental results of MBS-OEA with those of 9 existing well-designed algorithms [3-11]. MBS-OEA obtains a better performance, especially for the two larger problems, ami33 and ami49.

References

1. Jiao, L., Liu, J., Zhong, W.: An Organizational Coevolutionary Algorithm for Classification. IEEE Trans. Evol. Comput. (2006) to be published
2. <http://www.cse.ucsc.edu/research/surf/GSRC/progress.html>.

3. Guo, P. N., Takahashi, T., Cheng, C. K., Yoshimura, T.: Floorplanning Using a Tree Representation. *IEEE Trans. CAD.* 20(2) (2001) 281–289
4. Pang, Y., Cheng, C. K., Yoshimura, T.: An Enhanced Perturbing Algorithm for Floorplan Design using the O -tree Representation. in *Proc. ACM Int. Physical Design Symposia.* (2000) 168–173
5. Chang, Y. C., Chang, Y. W., Wu, G. M., Wu, S. W.: B^* -Trees: a New Representation for Nonslicing Floorplans. in *Proc. ACM/IEEE Design Automation Conf.* (2000) 458–463
6. Lin, J. M., Chang, Y. W.: TCG: a Transitive Closure Graph-Based Representation for General Floorplans. *IEEE Trans. VLSI.* 13(2) (2005) 288–292
7. Lin, J. M., Chang, Y. W.: TCG-S: Orthogonal Coupling of P^* -Admissible Representations for General Floorplans. *IEEE Trans. CAD.* 23(6) (2004) 968–980
8. Tang, X., Wong, D. F.: FAST-SP: a Fast Algorithm for Block Placement based on Sequence Pair. in *Proc. IEEE Asia South Pacific Design Automation Conf.* (2001) 521–526
9. Zhuang, C., Sakanushi, K., Jin, L., Kajitani, Y.: An Enhanced Q-sequence Augmented with Empty-room-insertion and Parenthesis Trees. in *Proc. Design Automation Test Europe.* (2002) 61–68
10. Young, E. F. Y., Chu, C. C. N., Shen, Z. C.: Twin Binary Sequences: a Nonredundant Representation for General Nonslicing Floorplan. *IEEE Trans. CAD.* 22(4) (2003) 457–469
11. Lin, J. M., Chang, Y. W., Lin, S. P.: Corner Sequence - a P-Admissible Floorplan Representation With a Worst Case Linear-time Packing Scheme. *IEEE Trans. VLSI.* 11(4) (2003) 679–686

Integrating the Simplified Interpolation into the Genetic Algorithm for Constrained Optimization Problems*

Hong Li^{1,2}, Yong-Chang Jiao¹, and Yuping Wang³

¹ National Laboratory of Antennas and Microwave Technology,
Xidian University, Xi'an, Shaanxi 710071, China
{lihong, ychjiao}@xidian.edu.cn

² School of Science, Xidian University, Xi'an, Shaanxi 710071, China

³ School of Computer Science and Engineering,
Xidian University, Xi'an, Shaanxi 710071, China

Abstract. In this paper, a hybrid genetic algorithm for solving constrained optimization problems is addressed. First, a real-coded genetic algorithm is presented. The simplified quadratic interpolation method is then integrated into the genetic algorithm to improve its local search ability and the accuracy of the minimum function value. Simulation results on 13 benchmark problems show that the proposed hybrid algorithm is able to avoid the premature convergence and find much better solutions with high speed compared to other existing algorithms.

1 Introduction

Consider the general nonlinear programming problem:

$$\begin{cases} \text{minimize} & f(x) \\ \text{subject to} & g_j(x) \leq 0, j = 1, \dots, m, x \in B \end{cases} \quad (1)$$

where $x = (x_1, \dots, x_n)^\top$, $B = \{x = (x_1, \dots, x_n)^\top | l_i \leq x_i \leq u_i, i = 1, \dots, n\}$.

Denote the feasible region by $I = \{x \in B | g_j(x) \leq 0, j = 1, \dots, m\}$. The equality constraint $h(x) = 0$ is converted into the inequality constraint: $|h(x)| - \delta \leq 0$, where δ stands for the degree of violation [2].

As a class of complex optimization problems, problem (1) arises in almost every field such as science, engineering and business. Evolutionary algorithms (EAs), including genetic algorithm (GA), evolution strategy (ES) and evolutionary programming (EP), have been found to be popular tools for solving several types of optimization problems (e.g. [1]-[3] [5] [6]). For problem (1), Runarsson and Yao [2] proposed an algorithm named stochastic ranking (SR), and Mezura-Montes et al. [3] presented a simple multimembered evolution strategy (SMES).

* This work was supported by the National Natural Science Foundations of China (60171045, 60374063 and 60133010).

These two ES algorithms are capable of finding better results compared with other algorithms available.

Here we integrate the simplified quadratic interpolation (SQI) method into a real-coded genetic algorithm (RCGA), and propose a hybrid genetic algorithm (HGA) for problem (1). In RCGA, all the constraints are incorporated into the fitness function by using the exact penalty method. The combination of an arithmetical crossover and a heuristic crossover is taken as the crossover operator. The proposed mutation operator enhances the random search ability of the algorithm. We also adopt the ranking selection and introduce some foreign individuals into the population in order to maintain its diversity. The SQI method is used in [4] to conduct global searches. However, we integrate the SQI method into RCGA to improve its local search ability, and to make the algorithm escape from the local optima. The proposed HGA has been compared with SR and SMES by solving 13 benchmark problems described in [2] [3]. The simulation results show that HGA is efficient.

This paper is organized as follows. RCGA is presented in Section 2, and the SQI method is given in Section 3. HGA for problem (1) is then proposed in Section 4. Experimental results and discussion are presented in Section 5. We finally conclude our paper in Section 6.

2 A Real-Coded Genetic Algorithm

Vector $x = (x_1, \dots, x_n)^\top$ is used to express an individual, which represents a trial solution of problem (1).

2.1 Generation of Initial Population

We generate at random pop uniformly distributed initial individuals $x^i(0)$, $i = 1, \dots, pop$, using the Monte Carlo method. Here pop denotes the population size.

2.2 Fitness Function

Based on the exact penalty method, the fitness function $fit(x)$ is expressed as:

$$fit(x) = \begin{cases} f(x), & \text{if } x \in I \\ f(x) + M \sum_{j=1}^m \max\{g_j(x), 0\}, & \text{otherwise} \end{cases} \quad (2)$$

where M ($M > 0$) is the penalty parameter.

2.3 Crossover Operator

Let p_c be the crossover probability. Randomly select two parents with probability p_c : $x^1 = (x_1^1, \dots, x_n^1)^\top$ and $x^2 = (x_1^2, \dots, x_n^2)^\top$, and generate the offspring $y = (y_1, \dots, y_n)^\top$ according to

$$y = \begin{cases} x^1 + \lambda(x^1 - x^2), & \text{if } fit(x^1) \leq fit(x^2) \\ x^2 + \lambda(x^2 - x^1), & \text{otherwise} \end{cases} \quad (3)$$

where $\lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$, $\lambda_i \in (-1, 1)$, $\lambda_i \neq 0$, $i = 1, \dots, n$.

After the algorithm runs N_2 generations, we confine $\lambda_i \in (-0.5, 0.5)$, $\lambda_i \neq 0$, $i = 1, \dots, n$. By using the crossover operator (3), pop_c individuals are generated.

2.4 Mutation Operator

Let p_m be the mutation probability, and $best = (best_1, \dots, best_n)^\top$ be the best individual, an individual with minimal fitness function value among all individuals generated till now.

For every offspring $y = (y_1, \dots, y_n)^\top$ such that $\|best - y\| \geq \varepsilon_1$ ($\varepsilon_1 = 10^{-4}$), randomly generate a number $r \in [0, 1]$. If $r < p_m$, then generate offspring $z = (z_1, \dots, z_n)^\top$ according to

$$z = best + V \cdot |c|, \quad (4)$$

where $V = \text{diag}(best_1 - y_1, \dots, best_n - y_n)$, $c \in R^n$ is a random vector obeying an n -dimensional standard normal distribution, i.e., $c \sim (N(0, 1), \dots, N(0, 1))^\top$.

For every offspring $y = (y_1, \dots, y_n)^\top$ such that $\|best - y\| < \varepsilon_1$ ($\varepsilon_1 = 10^{-4}$), we also randomly generate a number $r \in [0, 1]$. If $r < p_m$, then generate offspring $z = (z_1, \dots, z_n)^\top$ according to

$$z = best + \Delta v, \quad (5)$$

where $\Delta v \sim N(0, \sigma^2) = (N(0, \sigma_1^2), \dots, N(0, \sigma_n^2))^\top$, an n -dimensional normal distribution with mean value $0 = (0, \dots, 0)^\top$ and deviation $\sigma^2 = (\sigma_1^2, \dots, \sigma_n^2)^\top$.

After the algorithm runs N_2 generations, both random vectors c and Δv are replaced by different n -dimensional uniform distribution, i.e., $c \sim (U(0, 1), \dots, U(0, 1))^\top$, $\Delta v \sim (U(-0.5, 0.5), \dots, U(-0.5, 0.5))^\top$. After the mutation operation, pop_m individuals are generated.

2.5 Selection Operator

We first compare the fitness values of all the individuals, including those in the current population and all the offspring generated by the crossover and mutation operators, order ascendingly and relabel them. We then divide the best pop individuals into three groups, and randomly generate N_3 individuals to replace those in the middle group. The resulting pop individuals constitute the next population. In addition, we retain the best individual of every generation.

2.6 Stopping Criterion

If the algorithm is executed to the maximal number of generations N , and the best solution in the population has no improvement in last successive N_0 generations, then stop. The best solution we found in the last population is then taken as the approximate global optimal solution.

3 The Simplified Quadratic Interpolation Method

Here, we introduce the SQI method, which will be integrated into RCGA. Denote three individuals by $x^a = (x_1^a, \dots, x_n^a)^\top$, $x^b = (x_1^b, \dots, x_n^b)^\top$, $x^c = (x_1^c, \dots, x_n^c)^\top$,

and calculate their fitness values $f_a = fit(x^a)$, $f_b = fit(x^b)$, $f_c = fit(x^c)$. Suppose that $f(x^a) > f(x^b)$ and $f(x^c) > f(x^b)$. Then, derived from the three-point quadratic interpolation, the approximate minimal point $\bar{x} = (\bar{x}_1, \dots, \bar{x}_n)^\top$ is calculated according to

$$\bar{x}_i = \frac{1}{2} \left\{ \frac{[(x_i^b)^2 - (x_i^c)^2]f_a + [(x_i^c)^2 - (x_i^a)^2]f_b + [(x_i^a)^2 - (x_i^b)^2]f_c}{(x_i^b - x_i^c)f_a + (x_i^c - x_i^a)f_b + (x_i^a - x_i^b)f_c} \right\}, \quad (6)$$

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

4 The Hybrid Genetic Algorithm

Now we present the proposed algorithm for solving problem (1) as follows.

Data. Choose population size pop , crossover probability p_c , mutation probability p_m , maximal number of generations N , and four suitable integers N_0 , N_1 , N_2 and N_3 such that $N_0 \ll N$, $N_1 < N_2 < N$, $N_3 < pop$.

Step 0. Set $k = 0$, generate randomly the initial population set $P(k) = \{x^1(k), \dots, x^{pop}(k)\}$ and compute the fitness values $fit(x^i(k))$, $i = 1, \dots, pop$.

Step 1. If $k < N_2$, then choose randomly $\lambda_i \in (-1, 1)$, $\lambda_i \neq 0$, $i = 1, \dots, n$. Otherwise, choose randomly $\lambda_i \in (-0.5, 0.5)$, $\lambda_i \neq 0$, $i = 1, \dots, n$.

Step 1.1. Generate at random a number $r_1 \in [0, 1]$ and two integers $r_2, r_3 \in [1, pop]$, $r_2 \neq r_3$. If $r_1 < p_c$, then take $x^{r_2}(k)$ and $x^{r_3}(k)$ as two parents, execute crossover operator (3) and produce their offspring $y(k)$.

Step 1.2. Repeat Step 1.1 pop times, produce pop_c offspring individuals, which form a temporary population set denoted by $P_c(k) = \{y^1(k), \dots, y^{pop_c}(k)\}$, and compute their fitness values $fit(y^i(k))$, $i = 1, \dots, pop_c$.

Step 2. If $k < N_2$, then generate randomly vectors $c \sim (N(0, 1), \dots, N(0, 1))^\top$, $\Delta v \sim N(0, \sigma^2) = (N(0, \sigma_1^2), \dots, N(0, \sigma_n^2))^\top$, respectively. Otherwise, generate randomly vectors $c \sim (U(0, 1), \dots, U(0, 1))^\top$ and $\Delta v \sim (U(-0.5, 0.5), \dots, U(-0.5, 0.5))^\top$, respectively.

Step 2.1. For $i = 1, \dots, pop_c$, if $\|best - y^i(k)\| \geq \varepsilon_1$ ($\varepsilon_1 = 10^{-4}$), then generate randomly a number $r \in [0, 1]$. If $r < p_m$, then generate offspring $z(k)$ by using (4). Otherwise, generate randomly a number $r \in [0, 1]$. If $r < p_m$, then generate offspring $z(k)$ by using (5).

Step 2.2. By using Step 2.1, produce pop_m offspring individuals, which form a trial population set denoted by $P_m(k) = \{z^1(k), \dots, z^{pop_m}(k)\}$, and compute the fitness values $fit(z^i(k))$, $i = 1, \dots, pop_m$.

Step 3. If $k < N_1$, then go to Step 1. Otherwise, go to Step 3.1.

Step 3.1. Compare the fitness values of all the individuals, including those in the current population and all the offspring generated by the crossover and mutation operators, order ascendingly and relabel them. The ordered fitness value set is denoted by $FIT = \{F(1), F(2), \dots, F(pop + pop_c + pop_m)\}$, such that $F(1) \leq F(2) \leq \dots \leq F(pop + pop_c + pop_m)$. The corresponding

individual set is denoted by $\hat{X} = \{\hat{x}(1), \hat{x}(2), \dots, \hat{x}(pop + pop_c + pop_m)\}$. Let $x^b = \hat{x}(1)$, $f_b = F(1)$.

Step 3.2. Generate at random two integers $r_1 \in [2, 2 + L]$, $r_2 \in [3, 3 + L]$ such that $r_1 \neq r_2$. Here we choose $L = 10$. Let $x^a = \hat{x}(r_1)$, $f_a = F(r_1)$; $x^c = \hat{x}(r_2)$, $f_c = F(r_2)$.

Step 3.3. For some $i \in \{1, 2, \dots, n\}$, if $(x_i^b - x_i^c)f_a + (x_i^c - x_i^a)f_b + (x_i^a - x_i^b)f_c < \varepsilon_2$ ($\varepsilon_2 = 10^{-6}$), then go to Step 3.2. Otherwise, go to Step 3.4.

Step 3.4. Calculate \bar{x} by using (6), and then the fitness value $fit(\bar{x})$.

Step 3.5. If $fit(\bar{x}) < f_b$, then replace the worst solution in current population with \bar{x} .

Step 4. According to the selection strategy, select pop individuals to constitute the next population, and retain the best individual.

Step 5. If the stopping criterion is met, then stop, and record the best individual as the approximate global optimal solution of problem (1). Otherwise, set $k = k + 1$, and go to Step 1.

5 Experimental Results and Discussion

To evaluate the performance of the proposed HGA, we tested 13 benchmark problems described in [2] [3]. During the simulations, we adopted the following parameter suite:

Population size: $pop = 100$; Crossover probability: $pop_c = 0.8$; Mutation probability: $pop_m = 0.3$; Maximal number of generations: $N = 1000$; Four integers: $N_0 = N_1 = N_3 = 10$, $N_2 = 3N/5$; The degree of violation: $\delta = 10^{-6}$.

Penalty parameters for 13 benchmark problems are chosen as follows:

For **g03**, $M = 10^6$; for **g10**, $M = 10^5$; for **g02**, **g06**, **g08** and **g11**, $M = 10^4$; for **g01** and **g04**, $M = 10^3$; for **g07**, **g09** and **g12**, $M = 10^2$; for **g05**, $M = 10$; and for **g13**, $M = 0.1$.

We performed 30 independent runs for each benchmark problem. The statistical results of HGA are summarized in Table 1.

5.1 Simulated Results

We compared HGA against two approaches: SR [2] and SMES [3]. The best results are shown in Table 2. The results obtained by SR and SMES are cited from [2] [3], respectively. The mean values are provided for comparison in Table 3, and the worst results are also presented in Table 4.

As shown in Table 1, HGA is able to find the global optimum for nine benchmark problems (**g01**, **g03**, **g04**, **g05**, **g06**, **g08**, **g09**, **g11**, **g12**), and could find approximate solution very close to the global optimum for other four benchmark problems (**g02**, **g07**, **g10**, **g13**). In addition, the mean number of generations (MNG) and the mean number of fitness evaluations (MNFE) for finding the best solution are also shown in Table 1.

Table 1. Statistical results obtained by HGA for 13 benchmark problems over 30 independent runs. “MNFE” stands for mean number of fitness evaluations, and “MNG” for mean number of generations.

	Best	Mean	Median	Worst	St.Dev.	MNFE	MNG
g01	-15.000	-15.000	-15.000	-15.000	0	95063	669
g02	0.803600	0.785711	0.791450	0.744481	2.61E - 2	118317	838
g03	1.000	1.000	1.000	1.000	0	65481	470
g04	-30665.539	-30665.539	-30665.539	-30665.539	0	18267	135
g05	5126.498	5133.910	5130.149	5165.659	1.19E + 1	17038	126
g06	-6961.814	-6961.810	-6961.810	-6961.780	6.43E - 3	64448	463
g07	24.312	24.498	24.472	24.825	2.35E - 1	53614	384
g08	0.095825	0.095825	0.095825	0.095825	0	4565	74
g09	680.630	680.650	680.645	680.696	2.79E - 2	45055	325
g10	7049.499	7168.782	7130.587	7474.948	1.73E + 2	81264	579
g11	0.7500	0.7500	0.7500	0.7500	0	417	10
g12	1.000	1.000	1.000	1.000	0	15372	114
g13	0.053964	0.055683	0.055233	0.060171	2.35E - 3	26550	196

Table 2. Comparison of the best solutions found by HGA against SR, SMES, RCGA and VGA. “-” means that no feasible solutions were found.

	Optimal	SR	SMES	HGA	RCGA	VGA
g01	-15.000	-15.000	-15.000	-15.000	-15.000	-14.440
g02	0.803619	0.803515	0.803601	0.803600	0.803521	0.796231
g03	1.000	1.000	1.000	1.000	0.998	0.990
g04	-30665.539	-30665.539	-30665.539	-30665.539	-30665.539	-30626.053
g05	5126.498	5126.497	5126.599	5126.498	5126.521	-
g06	-6961.814	-6961.814	-6961.814	-6961.814	-6961.814	-6952.472
g07	24.306	24.307	24.327	24.312	24.330	31.097
g08	0.095825	0.095825	0.095825	0.095825	0.095825	0.095825
g09	680.63	680.630	680.632	680.630	680.633	685.994
g10	7049.25	7054.316	7051.903	7049.499	7057.870	9079.770
g11	0.75	0.750	0.75	0.7500	0.7500	0.75
g12	1.000	1.000000	1.000	1.000	1.000	1.000
g13	0.053950	0.053957	0.053986	0.053964	0.054008	0.134057

From Tables 2-4, it is shown that the performance of HGA is better than SR and SMES. In HGA we choose N=1000 (about 150000 fitness function evaluations (FFE)). However, SR needs 350000 FFE, and SMES requires 240000 FFE. Therefore the number of FFE required by HGA is much less than those by SR and SMES.

5.2 The Effect of SQI on HGA

It is necessary to verify how the SQI method works. In order to achieve this goal, we make the following simulations.

Table 3. Comparison of the mean solutions found by HGA against SR, SMES, RCGA and VGA. “-” means that no feasible solutions were found.

	Optimal	SR	SMES	HGA	RCGA	VGA
g01	-15.000	-15.000	-15.000	-15.000	-14.597	-14.236
g02	0.803619	0.781975	0.785238	0.785711	0.776538	0.788588
g03	1.000	1.000	1.000	1.000	0.993	0.976
g04	-30665.539	-30665.539	-30665.539	-30665.539	-30665.539	-30590.455
g05	5126.498	5128.881	5174.492	5133.910	5168.650	-
g06	-6961.814	-6875.940	-6961.284	-6961.810	-6961.560	-6872.204
g07	24.306	24.374	24.475	24.498	25.106	34.980
g08	0.095825	0.095825	0.095825	0.095825	0.095520	0.095799
g09	680.63	680.656	680.643	680.650	680.732	692.064
g10	7049.25	7559.192	7253.047	7168.782	7463.628	10003.225
g11	0.75	0.750	0.75	0.7500	0.7500	0.75
g12	1.000	1.000000	1.000	1.000	1.000	1.000
g13	0.053950	0.057006	0.166385	0.055683	0.075442	-

Table 4. Comparison of the worst solutions found by HGA against SR, SMES, RCGA and VGA. “-” means that no feasible solutions were found.

	Optimal	SR	SMES	HGA	RCGA	VGA
g01	-15.000	-15.000	-15.000	-15.000	-12.453	-14.015
g02	0.803619	0.726288	0.751322	0.744481	0.685752	0.779140
g03	1.000	1.000	1.000	1.000	0.973	0.956
g04	-30665.539	-30665.539	-30665.539	-30665.539	-30665.539	-30567.105
g05	5126.498	5142.472	5304.167	5165.659	5325.047	-
g06	-6961.814	-6350.262	-6952.482	-6961.780	-6956.065	-6784.255
g07	24.306	24.642	24.843	24.825	27.397	38.686
g08	0.095825	0.095825	0.095825	0.095825	0.094411	0.095723
g09	680.63	680.763	680.719	680.696	681.170	698.297
g10	7049.25	8835.655	7638.366	7474.948	8264.828	11003.533
g11	0.75	0.750	0.75	0.7500	0.7500	0.752
g12	1.000	1.000000	1.000	1.000	1.000	0.999
g13	0.053950	0.216915	0.468294	0.060171	0.328487	-

First we adopt the real-coded GA (RCGA), in which the SQI method is not used as a local search operator, to solve 13 benchmark problems. 30 independent runs are performed for each problem. We compare RCGA against a version of GA (VGA) presented in [3] and HGA, and present the results in the last three columns of Tables 2-4.

Analyzing the results shown in Tables 2-4, we conclude that RCGA surpasses VGA. Results in Tables 2-4 also indicate that the quality of the solutions found by RCGA are significantly worse than those by HGA for all the benchmark problems. In some cases, RCGA gets trapped into the local minima due to the premature convergence.

To recapitulate, the performance of HGA is much better than that of RCGA. The simulation results indicate that the SQI method is a powerful heuristic tool.

6 Conclusion

In this paper, we integrate the SQI method into RCGA, and propose HGA. The SQI method is a powerful heuristic way for improving the local search ability of the algorithm, avoiding its premature convergence, improving the accuracy of the minimum function value, as well as reducing the computational burden. The hybrid genetic algorithm has also been compared with two existing algorithms, SR and SMES, by solving 13 benchmark problems. Numerical results show that the performance of HGA is efficient and effective.

References

1. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading, MA (1989)
2. Runarsson, T.P., Yao, X.: Stochastic Ranking for Constrained Evolutionary Optimization. *IEEE Trans. on Evolutionary Computation* **4** (2000) 284-294
3. Mezura-Montes, E., Coello, C.A.C.: A Simple Multimembered Evolution Strategy to Solve Constrained Optimization Problems. *IEEE Trans. on Evolutionary Computation* **9** (2005) 1-17
4. Ali, M.M., Törn, A., Vitanen, S.: A Numerical Comparison of Some Modified Controlled Random Search Algorithms. *Journal of Global Optimization* **11** (1997) 377-385
5. Hrstka, O., Kučerová, A.: Improvements of Real Coded Genetic Algorithms Based on Differential Operators Preventing Premature Convergence. *Advances in Engineering Software* **35** (2004) 237-246
6. Preux, Ph., Talbi, E.-G.: Towards Hybrid Evolutionary Algorithms. *International Transactions in Operational Research* **6** (1999) 557-570

Using Ensemble Method to Improve the Performance of Genetic Algorithm

Shude Zhou and Zengqi Sun

State Key Lab of Intelligent Technology and Systems, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
zsd03@mails.tsinghua.edu.cn, szq-dcs@mail.tsinghua.edu.cn

Abstract. Ensemble method has been deeply studied and widely used in the machine learning communities. Its basic idea can be represented as: A ‘weak’ learning algorithm that performs just slightly better than random guessing can be ‘boosted’ into an arbitrarily accurate ‘strong’ learning algorithm. Inspired from the fascinating idea, the paper used ensemble method to improve the performance of genetic algorithm and proposed an efficient hybrid optimization algorithm: GA ensemble. In GA ensemble, a collection of genetic algorithms are designed to solve the same problem and population of each algorithm is sampled from a solutions pool using bagging method. Experiments on combinatorial optimization problem and GA-deceptive problems show that ensemble method improves the performance of genetic algorithm greatly.

1 Introduction

Genetic algorithm (GA) is a nature inspired stochastic optimization algorithm, in which a population of candidate solutions is evaluated and updated iteratively according to Darwinian evolutionary theory [1], [2]. In order that GA can more efficiently solve difficult problems, especially problems with high dimension, nonlinear and strong-related variables, efforts have been made to improve the performance of GA [2]. This paper deliberates on improving the performance of GA using the idea of ensemble method.

In machine learning, a collection of predictors or classifiers are combined as an ensemble for the same task. Both theoretical and empirical research has demonstrated that a good ensemble can improve accuracy by perturbing the learning set [6]. For example, neural network ensemble is a successful ensemble paradigm where a collection of a finite number of neural networks is trained for the same work, and it has already been successfully applied to diversified practical areas [4], [10]. The basic idea of ensemble method can be represented as: A ‘weak’ learning algorithm that performs just slightly better than random guessing can be ‘boosted’ into an arbitrarily accurate ‘strong’ learning algorithm [3].

Inspired from the ‘ensemble method’, there comes up with the question of whether many ‘weak’ stochastic optimization algorithms can be combined to construct ‘strong’ optimization algorithm. With this interesting question, the paper tries to develop a novel evolutionary optimization paradigm called GA ensemble. In GA ensemble, a collection of genetic algorithms are designed to solve the same problem

and population of each algorithm is sampled from a solutions pool using bagging method. Experiments show ensemble method can effectively improve the performance of GA and can convert a ‘weak’ genetic algorithm to a ‘strong’ one.

The remainder of the paper is organized as follows. In section 2, how to use ensemble method to improve the performance of GA is described in details. Section 3 presents the computational experiments. The conclusions are given in Section 4.

2 GA Ensemble

For ensemble method in machine learning, a collection of predictors or classifiers are combined as an ensemble for the same task. Both theoretical and empirical research has demonstrated that a good ensemble can improve the accuracy by perturbing the learning set [3], [4], [9]. An ensemble constructed by running the local search form many different starting points may provide a better approximation to the true unknown function than any of the individual classifiers [3]. Learning the best hypothesis for a classifier is a stochastic optimization problem. Now that ensemble method can solve such optimization problem better than one individual, it seems intuitive that ensemble method can improve the performance of evolutionary optimization than a single algorithm. Diversity is the bridge of ensemble method and evolutionary computation. Ensemble method brings more diversity into the evolutionary algorithms.

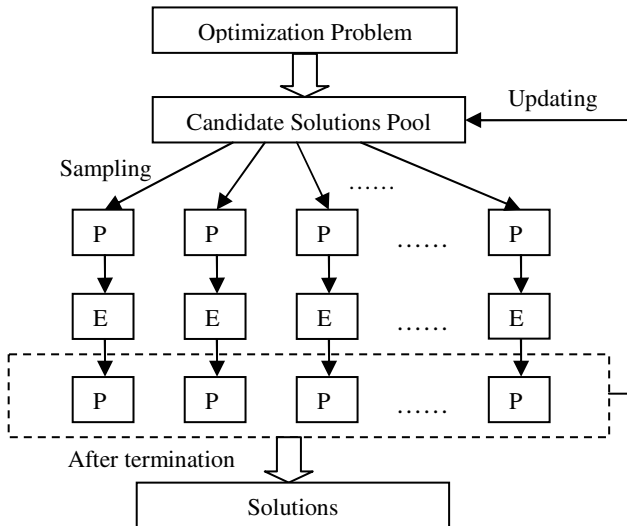


Fig. 1. The schematic procedure of GA ensemble. P=population, E=Evolution

The framework of ensemble method in GA is schematically illustrated in figure 1. There is a collection of GAs to solve the same optimization problem and information interaction happened among these algorithms during their evolution. The individual algorithms cooperate to evolve by sharing a common solution pool, from which

population of each algorithm in each generation are generated randomly. After the population is constructed, each individual algorithm can update the population using evolutionary operators such as selection, crossover and mutation operators. And then, the data in solution pool are substituted by data in all the updated populations. Till now, a loop is finished. The pseudocode of GA ensemble is as follows.

GA Ensemble Algorithm

Initialize solution pool: SP , new_SP . SP is a solution set randomly generated and new_SP is an empty set.

Initialize parameters. There are n GAs in GA ensemble. V_k is the population of GA_k , $k=1, \dots, n$.

while (generations++ < Max_generation)

for each GA_k

for each individual V_{ki} do

V_{ki} is randomly selected from solution pool

end for

Evolve one or several generations (selection, crossover, mutation)

Generate new individuals set V_k

Add V_k to new_SP

end for

$SP=new_SP$

empty new_SP

End while

Fig. 2. The pseudocode of GA ensemble

In GA ensemble, every population is sampled from a common solution pool. Though the populations are sampled from the same solution pool and there may be some same individuals in different populations, the constructed populations may have different statistical characters due to the randomness. In the field of machine learning, there are kinds of sampling method to construct new training sets for individual classifier, such as bagging method, boosting method [5], [6], [9], [10]. In this paper, bagging method is used to generate population. Suppose that, there are n GAs in GA ensemble and the n populations are sampled from solution pool $SP = \{p_1, p_2, \dots, p_N\}$ where N is the size of solution pool. Every population is constructed by draw data from SP randomly, with replacement. That is, each p_i in SP may appear repeated times or not at all in any population [5].

3 Experiments

0-1 knapsack problems and fully order-4 deceptive problems are used to verify the performance of GA ensemble. We use simple GA and GA ensemble to solve the two problems. For the purpose of fair comparisons, the number of individuals generated at each generation is made the same in both algorithms. That is, the population of GA is equal to the total number of individuals in all the populations in GA ensemble, so that the two algorithms are compared based on equal fitness evaluations.

3.1 Performance of GA Ensemble in 0-1 Knapsack Problem

Three 0-1 knapsack problems with 50, 120 and 240 items are used to test the performance of simple GA and GA ensemble. For each experiment, we randomly obtain the values for the cost and weight associated to each item, as well as the capacity of the knapsack. The parameter settings are shown in Table 1, where P_n : population size, p_m : mutation probability, p_c : crossover probability, E_n : number of GA components in GA ensemble.

Table 1. Parameter settings in simple GA and GA ensemble for solving 0-1 knapsack problem

Parameters		50-item	120-item	240-item
Simple GA	P_n	80	160	300
	p_m	0.005	0.01	0.01
	p_c	0.70	0.80	0.80
GA Ensemble	P_n	10	20	30
	E_n	8	8	10
	p_m	0.005	0.01	0.01
	p_c	0.70	0.80	0.80

The two algorithms, GA ensemble and simple GA, run 20 times respectively with max generation 1000. The best, average and worst solutions found using these algorithms are compared in Table 2. Obviously, the experiments reveal that the GA ensemble outperforms simple GA greatly for different knapsack problems. For problem with 50 items, GA ensemble obtained the optimum solution all the time over the 20 runs, while simple GA obtained the optimum solution 17 times and the average solution is 975.2. As the dimension of the problem increased, the superiority of GA ensemble became much clearer. For 120-item and 240-item problems, GA ensemble obtained the much better solutions than simple GA as shown in Table 3. For example, in solving 120-item problem, the worst solution 5.2699e+03 obtained by GA ensemble over 20 runs is better than the best solution 5.1745e+03 obtained by simple GA. It indicated that the GA ensemble is more efficient than simple GA in solving high-dimension problems.

It should be noted that the population of componential GA in GA ensemble is much less than that of simple GA. However, the solutions obtained by a collection GA with small population are much better than solutions obtained by a GA with large population. It shows that a collection of ‘week’ GAs (each with population size 10, 20 and 30) can solve problem better than a ‘strong’ GA (with population size 80, 160 and 300).

Table 2. Performance comparison of SGA and GA ensemble for knapsack problem with 50, 120 and 240 items. The best, average and worst solution are statistically drawn over 20 runs.

		50-item	120-item	240-item
Simple GA	Best	982.4	5.1745e+03	2.1718e+04
	Ave	978.9	5.0300e+03	2.1188e+04
	Worst	963.6	4.8893e+03	2.0889e+04
GA Ensemble	Best	982.4	5.3195e+03	2.3310e+04
	Ave	982.4	5.2970e+03	2.3062e+04
	Worst	982.4	5.2699e+03	2.2783e+04

3.2 Performance of GA Ensemble in GA-Hard Problem

There is a class of problems that are very difficult to optimize using simple genetic algorithm, usually called GA-hard or deceptive problems [7], [11]. To further verify the ability of GA ensemble, we use order-4 fully deceptive problem as a test case. In the experiments, we ranged the problem dimension from 40 to 200 with increasing step 40. For all trails, the crossover probability is 0.70 and the mutation probability 0.005. The population sizes in simple GA are set 80, 120, 160, 200 and 240 respectively for problems with dimension 40, 80, 120, 160 and 200; the population sizes of GA in GA ensemble are 10, 12, 16, 20, 24, and the numbers of GAs in GA ensemble are 8, 10, 10, 10 and 10. The population settings guarantee that the number of function evaluations is equal in both algorithms. Both simple GA and GA ensemble are used to solve these problems 20 independent times with max generation 1000.

The results obtained by the algorithms are compared in Table 3. For low-dimension problem such as 40 and 80 the GA ensemble performs slightly better than simple GA, while the superiority of GA ensemble becomes much clearer as the problem dimension increases. It is found that even for 200-dimension GA-hard problem GA ensemble can obtain optimum solution in some trials. We also compare the scalability of both algorithms which is measured by the average number of fitness evaluations until convergence [7], [8]. In GA ensemble, the increasing of fitness function evaluations required is much less than simple GA. GA ensemble is more effective in solving high dimension problems and exhibits better scalability.

Table 3. Performance evaluation of simple GA and GA ensemble for order-4 fully deceptive problems with different dimensions. The evaluation is calculated by counting the total number of function evaluations until converging to global optimum averaged over the successful runs. For 120, 160, 200 dimension, simple GA failed in all runs, so the evaluation is indicated '--'.

		40	80	120	160	200
Simple GA	Best	300	600	870	1110	1290
	Ave	298.5	592.5	843	1047	1152
	Worst	270	570	810	990	1110
	Evaluations	16058	119760	--	--	--
GA Ensemble	Best	300	600	900	1200	1500
	Ave	300	600	894	1165.5	1456
	Worst	300	600	870	1140	1380
	Evaluations	6350	25800	66560	128330	235900

4 Conclusions and Future Works

In the above section, ensemble method is successfully applied to enhance the performance of genetic algorithm. In this part, we summarize the characters of ensemble method in evolutionary algorithms as follows.

1. Ensemble method can efficiently improve the performance of GA in solving the two test cases.
2. GA ensemble is easy to implement. The idea of ensemble method is very simple, and users can easily implement the GA ensemble to solve their problems.

3. From weak to strong. By combining several simple or weak genetic algorithms, we can construct 'strong' genetic algorithm. For example, for 0-1 knapsack problem with dimension 120, multiple 'weak' algorithms with population size 20 cooperate, and exhibit better performance than a 'strong' algorithm with population 200. Ensemble method can convert an algorithm from weak to strong.
4. GA ensemble is more effective in solving high dimension problems and exhibits better scalability. In solving knapsack problems and order-4 deceptive problems with higher dimension, GA ensemble outperforms simple GA greatly.

More sufficient experimental verification of GA ensemble and theoretical research are our next works. And there is an open question: can we construct a 'strong' optimization algorithm by ensemble several 'weak' optimization algorithms? For many problems with high-dimension and strong-related valuables, it is very difficult to design efficient algorithm to search the optimum. However, if we can construct efficient algorithm by combining several many not-so-efficient algorithms, it will open a new door for solving hard problem. GA ensemble may be our first attempt to open the new door.

Acknowledgement

The work was funded by the National Key Project for Basic Research of China (Grant No: G2002cb312205).

References

1. Back, T.: Evolutionary algorithms in theory and practice, Oxford University Press (1996)
2. Mitsuo, G., Runwei, C.: Genetic Algorithms & Engineering Optimization. John Wiley & Sons Ltd. (2000)
3. Schapire, R.E.: The Strength of Weak Learnability. Machine Learning, Vol.5. (1990) 197-227
4. Hansen, L.K., Salamon, P.: Neural network ensembles. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.12. (1990) 993-1001
5. Breiman, L.: Bagging predictor. Machine learning, Vol.24. (1996) 123-140
6. Richard A.B.: An introduction to ensemble methods for data analysis. UCLA (2004)
7. Pelikan, M., Goldberg, D.E., Cantú-paz, E.: Linkage Problem, Distribution Estimation and Bayesian Networks. Evolutionary Computation, Vol.8. (2000) 311-340
8. Larrañaga, L., Lozano, J.A.: Estimation of distribution algorithm: A new tool for evolutionary computation. Kluwer Academic Publishers (2002)
9. Opitz, D., Maclin, R.: Popular Ensemble Methods: An Empirical Study. Journal of Artificial Intelligence Research, Vol.11. (1999) 169-198
10. Zhihua, Z., Shifu, C.: Neural network ensemble. Chinese Journal of Computers, Vol.25. (2002) 1-8
11. Goldberg, D.E., Deb, K., Horn, J.: Massive multimodality, deception, and genetic algorithms. Parallel Problem Solving from Nature, Vol.2. (1992) 37-46

Parallel Mining for Classification Rules with Ant Colony Algorithm

Ling Chen^{1,2} and Li Tu¹

¹Department of Computer Science, Yangzhou University,
Yangzhou 225009, China

²National Key Lab of Novel Software Tech, Nanjing University,
Nanjing 210093, China
lchen@yzcn.net

Abstract. A parallel ant colony algorithm for mining the classification rules is presented. By preprocessing classification information on databases and combining ACO with parallel strategies, our algorithm could extract classification rules efficiently in parallel. Experimental results on several benchmark datasets show that our algorithm can discover classification rules more quickly with better accuracy, simplicity than other methods such as improved Ant-Miner algorithm and C4.5 based on well known decision tree algorithm.

1 Introduction

Rule discovery is an important data mining task since it generates a set of symbolic rules that describe each class or category in a natural way. The prediction rules for knowledge discovery are often represented as IF-THEN form, and the classification rules can be described as: IF<conditions> THEN <class>. In each rule, the <condition> part, which is called the antecedent, usually contains a logic combination of the predictor attributes in the form: term1 AND term2 AND...AND term_j. Each term is a triple <attribute, operation, value>, such as <Color =Red>. The consequent part <class> shows the predicted class whose cases satisfy the <condition>. In recent years, algorithms of classification are presented, such as the algorithms based on classical decision tree[1] and C4.5[2]. Wesley Romão et al. have also presented an algorithm for classification problem based on GA[3].

M.Dorigo et al. first advanced the ant colony optimization algorithm (ACO) to solve discrete optimization problem[4]. In 2002, Parpinelli, Lopes, & Freitas first proposed an algorithm for classification rule discovery [5] based on ACO, and got results better than that of the traditional decision tree algorithm. Based on the framework of Parpinelli's algorithm, some improved algorithms have presented, such as the ant classification system ant-classifier[6] and the Ant-Miner2[7]. Since all these algorithms neglected the difference of the influences by the attributes on the classification, they have high time complexity, and the quality of the rules they get should be improved further.

In this paper, we present a parallel ant colony algorithm for classification rules discovery (CRPA). By combining ACO with the parallel strategy, CRPA divides the total ants into several groups which are assigned to the processors so as to exca-

vate all kinds of rules in parallel. Since each attribute has different influences on different classes, we define probability functions for the ants to select the attribute and its values of the antecedent in the rule to strengthen the positive feedback for rules mining.

2 The Framework of Algorithm CRPA

In CRPA, each processor is assigned a value $B_p = \{C = D_{V,p}^C\}$ which is the consequent part of the rules it should discover, where $D_{V,p}^C$ is the p -th class ID of V partitions defined in the class attribute C , where $V = total-class$, $1 \leq p \leq V$.

The framework of the proposed algorithm CRPA is described as follows.

Algorithm CRPA:

Input: training set.

Output: sequential rule lists of each class, $rule_p = \{rule_p^k\}, rule_p^k = [I_p^k \Rightarrow B_p]$;

1. Initialize the parameters;
2. Transform continuous attributes into discrete ones;
3. Divide the total ants into *total-class* groups and assign each group to a processor, $m_0 = m / total-class$;
4. Broadcast the training set to all the processors;
5. For $p = 1$ to *total-class* par-do
6. While (Number of the uncovered cases in the training set of each class > *max-uncovered-cases*) do
7. $k = 1$;
8. Repeat
9. While (*allowed_k* is not empty) do
10. ant_k selects attribute i ;
11. $allowed_k = allowed_k - \{i\}$;
12. Ant_k selects j -th partition of $A_i: D_{V,j}^{A_i}$;
13. If (number of covered cases of $I_p^k = I_p^k + \{term_{i,j}\} > min-cases$) Then
14. Compute Q of $rule_p^k$;
15. If (the value of Q becomes smaller) Then
16. Remove $term_{i,j}$ from I_p^k ;
17. Ant_k updates the pheromone on A_i ;
18. End If
19. End If
20. End While
21. Update the pheromone on $term_{i,j}$ globally;
22. $k = k + 1$;
23. Until ($k \geq m_0$) or (ant_k constructed the same rule with $ant_1 \sim ant_{k-1}$)
24. Select the best rule $rule_p^k$ among all constructed rules into $rule_p$;

25. According to $rule_p^k$, select the cases correctly covered by the selected rule from the training set of class p and broadcast to each processor;
26. Receive the updated training set of processor p ;
27. Update the parameters of α, β ;
28. End While
29. End For

Our parallel ant colony algorithm is based on the computational model of massively parallel processors (MPP) which adopts the message passing method[8].

3 Select the Attribute

In line 10 of CRPA, the k th ant in processor p , where the rules for the p th class are mined, selects the j th attribute according to the following probability function:

$$Pa_p^k(t) = \begin{cases} \frac{[\tau_j^p(t)]^\alpha [\eta_j^p]^\beta}{\sum_{r \in allowed_k} [\tau_r^p(t)]^\alpha [\eta_r^p]^\beta} & j \in allowed_k \\ 0 & otherwise \end{cases} \quad (1)$$

Here, $allowed_k$ is the set of the attributes not selected by the k th ant. The value of η_j^p is set according to the correlation degree of attribute j to class p by correlation coefficient. Suppose the j th attribute value of case $_i$ is x_i , and y_i is defined as follows:

$$y_i = \begin{cases} 0, & case_i \notin class\ p \\ 1, & case_i \in class\ p \end{cases}, i \in [1, n]. \quad (2)$$

Then η_j^p is defined as:

$$\eta_j^p = \frac{\frac{1}{n} \sum_{t=1}^n (x_t - \bar{x})(y_t - \bar{y})}{\sqrt{\frac{1}{n} \sum_{t=1}^n (x_t - \bar{x})^2} \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \bar{y})^2}}, p \in [1, total - class]. \quad (3)$$

We set $\tau_j^p(0) = \eta_j^p$. Line 17 of CRPA updates the pheromone $\tau_j^p(t)$ according to formula (4), where constant ρ is the coefficient of evaporation.

$$\tau_j^p(t+1) = (1 - \rho) \cdot \tau_j^p(t) + \Delta \tau_j^p, p \in [1, total - class]. \quad (4)$$

4 Select the Attribute Partition

After selecting the attribute i , ants should choose relevant $term_{ij}$ by the probability function of attribute partition as follows:

$$Pv_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}^p(t)]^\alpha [\eta_{ij}^p]^\beta}{\sum_{r \in [1, V]} [\tau_{ir}^p(t)]^\alpha [\eta_{ir}^p]^\beta} & j \in [1, V] \\ 0 & otherwise \end{cases} \tag{5}$$

Here, V is the number of the value partitions of attribute i . The heuristic function η_{ij}^p for $term_{ij}$ is defined as follows:

$$\eta_{ij}^p = \frac{|T_{ij}^p|}{|T_{ij}|}, p \in [1, total - class]. \tag{6}$$

Here $|T_{ij}^p|$ is the number of cases where $A_i = D_{V,j}^{A_i}$ in class p , $|T_{ij}|$ is the number of cases where $A_i = D_{V,j}^{A_i}$ in training set. Line 21 of CRPA updates the pheromone $\tau_{ij}^p(t)$ according to formula (7), where constant ρ is the coefficient of evaporation.

$$\tau_{ij}^p(t+1) = \begin{cases} \tau_{ij}^p(t) + (\frac{Q}{d} - \rho) \cdot \tau_{ij}^p(t), & term_{ij} \in rule \\ (1 - \rho) \cdot \tau_{ij}^p(t), & term_{ij} \notin rule \end{cases}, p \in [1, total - class]. \tag{7}$$

Here, d is the length of constructed rule. Q is the quality of constructed rules in the algorithm and is defined as follows:

$$Q = \begin{cases} \frac{TP}{TP + FN + FP}, & if TN = 0 \\ \frac{TP}{TP + FN} \times \frac{TN}{TN + FP}, & otherwise \end{cases} \tag{8}$$

Here TP is the number of cases covered by the rule that have the class predicted by the rule; FP is the number of cases covered by the rule that have a class different from the class predicted by the rule; FN is the number of cases that are not covered by the rule but that have the class predicted by the rule and TN is the number of cases that are not covered by the rule and that do not have the class predicted by the rule.

Line 27 in CRPA adaptively adjusts the parameters α, β to balance τ_{ij}^p and η_{ij}^p as follows:

$$\alpha = e^{-\psi}, \beta = \frac{1}{\alpha}. \tag{9}$$

Where, $\psi = \frac{\sum_{i=1}^a \sum_{j=1}^V |\tau_{ave}^p - \tau_{i,j}^p|}{aV}$ is the pheromone distributing weight to measure the

distribution of pheromone on terms and $\tau_{ave}^p = \frac{\sum_{i=1}^a \sum_{j=1}^V \tau_{i,j}^p}{aV}$ is defined as the average amount of pheromone on all the terms on processor p .

5 Experimental Results and Discussion

In this section, we show the test results on the several data benchmarks from data warehouse UCI to compare our method with that of improved Ant-Miner and C4.5 algorithm. The algorithms are tested on the passive parallel processors Dawn 2000 using MPI (C bounding).

Table 1. The accuracy comparisons of three algorithms on four databases

Databases	Three methods		
	Improved Ant-Miner Accuracy (%)	C4.5 Accuracy (%)	CRPA Accuracy (%)
Breast cancer	93.11	94.71	97.31
Tic-tac-toe	75.44	80.18	86.35
Wine	97.90	92.20	98.99
Heart disease	60.50	58.33	64.01

From the comparisons above, we could see that the accuracy of C4.5 is better than the improved Ant-Miner on Breast cancer database, but the accuracy of our algorithm CRPA could reach to 97.31%, which is much better than the other two algorithms by 3-4 percentages and is the best. The comparisons on the other three databases show that ant-based algorithms are better than the decision tree based algorithm C4.5.

Table 2. Comparisons of average numbers of rules and terms of three algorithms

Databases	Average number of rules			Average number of terms		
	Improved Ant-Miner	C4.5	CRPA	Improved Ant-Miner	C4.5	CRPA
Breast cancer	6.3	11.1	5.6	13.2	44.1	10.5
Tic-tac-toe	8.7	83.0	6.8	11.0	384.2	9.1
Wine	4.1	5.7	3.5	10.8	15.7	7.9
Heart disease	10.5	49.0	8.1	16.2	183.4	14.6

As shown in Table 2, we find that the average number of discovered rules by C4.5 is a little larger on four databases. Furthermore, it can be seen from the number of terms per-rule that, the result of C4.5 is even more redundant, which reaches to 384.2 on Tic-tac-toe. We also find that the discovered rules by improved Ant-Miner have been greatly simplified, and CRPA has got the simplest rules.

6 Conclusion

In this paper, we present a parallel algorithm for classification rule discovery by ACO(CRPA). Compared with the traditional serial C4.5 algorithm and the advanced Ant-Miner, the experimental results on several benchmark databases indicate that our parallel algorithm CRPA could excavate better classification rules in terms of simplicity and accuracy. Associated rules discovery will be our future work.

Acknowledgement

This work was supported in part by the Chinese National Natural Science Foundation under grant No. 60473012, Chinese National Foundation for Science and Technology Development under contract 2003BA614A-14, and Natural Science Foundation of Jiangsu Province under contract BK2005047.

References

- Chang, S.-Y., Lin, C.-R., & Chang, C.-T.: A fuzzy diagnosis approach using dynamic fault trees. *Chemical Engineering Science*, 57(15) (2002) 2971-2985.
- Quinlan, J.R.: *C4.5: Programs for Machine Learning*. San Francisco: Morgan Kaufmann, (1993).
- Wesley Romão, Alex A. FREITAS, Itana M.de S.Gimenes.: Discovering interesting knowledge from a science and technology database with a genetic algorithm. *Applied Soft Computing*, 4 (2004) 21-137.
- Dorigo, M., Gambardella, L.M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Trans. On Evolutionary Computation*, 1(1) (1997) 53-66.
- Parepinelli, R. S., Lopes, H.S., & Freitas, A.: An Ant Colony Algorithm for Classification Rule Discovery. In H. A. a. R. S. a. C. Newton(Ed.), *Data Mining: Heuristic Approach: Idea Group Publishing*, (2002).
- Shelokar, P. S., Jayaraman, V. K., Kulkarni, B. D.: An ant colony classifier system: application to some process engineering problems. *Computers and Chemical Engineering* 28 (2004) 1577-1584
- Bo Liu, Hussein A.Abbass, Bob McKay.: Density-based Heuristic for Rule Discovery with Ant-Miner, *The 6th Australia-Japan Joint Workshop on Intelligent and Evolutionary System*, (2002) 180-184.
- Harry F. Jordan, Gita Alaghband: *Foundations of Parallel Processing*, Prentice Hall (2003).

A Genetic Algorithm Approach on Reverse Logistics Optimization for Product Return Distribution Network

Gengui Zhou, Zhenyu Cao, Jian Cao, and Zhiqing Meng

College of Business and Administration,
Zhejiang University of Technology, Zhejiang 310032, China
ggzhou@zjut.edu.cn

Abstract. Traditionally, product returns have been viewed as an unavoidable cost of distribution systems. Up to now there are few studies to address the problem of determining the number and location of centralized product return centers where returned products from retailers or end-customers are collected for manufacturers' or distributors' repair facilities while considering the distribution system. To fill the void in such a line of research, this paper proposes a nonlinear mixed-integer programming model and a genetic algorithm that can solve the distribution problem with forward and reverse logistics simultaneously. Compared with a partly enumeration method, the numerical analysis shows the effectiveness of the proposed model and its genetic algorithm approach.

1 Introduction

Reverse logistics is actually very involved in modern industry in the distribution activities such as product returns, source reduction/conservation, recycling, substitution, reuse, disposal, refurbishment, repair and remanufacturing [1][2]. Over the last decade, reverse logistics has had a significant economic impact on industry as well as society. This impact can be seen either as detrimental to a company, and thus avoided, or as a competitive advantage with potential for capturing market share. Companies that receive items back from the customer who try to hide from the significance of reverse logistics miss profit-making opportunities. On the other hand, companies that use reverse logistics as an opportunity for enhanced business will prosper by maintaining customer support, the ultimate issue for profitability [3][4].

Recently, Min et. al. proposed a mathematical model and its solution procedure that can optimally create the reverse logistics network linking initial collection points, centralized product return centers, and manufacturing facilities, which deal with both forward and reverse product flows [5]. But the location/allocation decisions regarding the initial collection points and centralized product return centers only cope with the success and failure of reverse logistics operations on the assumption of deterministic product demand and returned products. Obviously, it is not always the case in practice for the reverse logistics network.

Based on the research work of Min et. al., this paper develops a nonlinear mixed-integer programming model to solve the reverse logistics problem involving product returns with stochastic demand and return products. As it is an *NP*-hard problem, a genetic algorithm approach is proposed to determine the number and location of centralized product return centers (i.e., reverse consolidation points) where returned products from retailers or end-customers could be collected, sorted, and consolidated into a large shipment destined for manufacturers' or distributors' repair facilities. Numerical analysis shows the effectiveness of the proposed method to deal with such a kind of complicated problem compared with a designed partly enumeration method.

2 Modeling Formulation

In developing the mathematical model for reverse logistics network design, we formulate the following nonlinear mixed-integer programming:

$$\begin{aligned}
 \min \quad TC = & \sum_{i \in W} (fw + uw \cdot S_i)Y_i + \sum_{i \in W} (fr + ur \cdot R_i)Z_i - \sum_{i \in W} (wr \cdot Y_i \cdot Z_i) \\
 & + \sum_{i \in W} \sum_{j \in C} p_{ij}^f x_{ij}^f + \sum_{j \in C} (us \cdot E[\max\{(\sum_{i \in W} x_{ij}^f - D_j^f), 0\}] \\
 & + uc \cdot E[\max\{(D_j^f - \sum_{i \in W} x_{ij}^f), 0\}]) + E[\sum_{j \in C} \sum_{i \in W} (p_{ij}^r x_{ij}^r D_j^r)] \\
 & + ua \sum_{i \in W} E[\max\{0, \sum_{j \in C} (x_{ij}^r D_j^r) - R_i Z_i\}] \tag{1}
 \end{aligned}$$

$$\text{s.t.} \quad \sum_{j \in C} x_{ij}^f \leq S_i Y_i \quad (i \in W), \tag{2}$$

$$S_i Y_i \leq MAX_SUPPLY \quad (i \in W), \tag{3}$$

$$R_i Z_i \leq MAX_REPAIR \quad (i \in W), \tag{4}$$

$$\sum_{i \in W} x_{ij}^r = 1 \quad (j \in C), \tag{5}$$

$$x_{ij}^f, x_{ij}^r, S_i, R_i \geq 0 \quad (i \in W, j \in C), \tag{6}$$

$$Y_i, Z_i = 0 \text{ or } 1 \quad (i \in W). \tag{7}$$

2.1 Model Parameters

$W = \{ 1, 2, \dots, NW \}$: set of candidate centers,

$C = \{ 1, 2, \dots, NC \}$: set of customers,

fw : fixed cost to maintain a distribution center,

uw : variable cost,

MAX_SUPPLY : maximum capacity for distribution centers,

fr : fixed cost to maintain a product return center,

ur : variable cost associated a product return center,

MAX_REPAIR : maximum capacity for product return centers,

- wr : saved cost if a distribution center plays a role as a product return center,
- p_{ij}^f : unit shipment cost from candidate center i to customer j ,
- p_{ji}^r : unit shipment cost from customer j to candidate center i ,
- $p_{ij}^f = p_{ji}^r = a + bL_{ij}$: where L_{ij} is the distance from candidate center i to customer j ,
- D_j^f : demand of customer j in the uniform probability distribution of $U[D_{j,min}^f, D_{j,max}^f]$,
- D_j^r : returned products of customer j in the uniform probability distribution of $U[D_{j,min}^r, D_{j,max}^r]$
- us : unit processing cost of those exceeded products when the number of products from distribution center to customer is larger than those required
- uc : unit chance loss cost of those shortage products when the number of products from distribution center to customer is smaller than those required
- ua : unit processing cost of those products when the number of returned products to return center is larger than its capacity

2.2 Decision Variables

- x_{ij}^f : quantity of products shipped from distribution center i to customer j ,
- x_{ij}^r : ratio of returned products to product return center i over its total returned products of customer j ,
- S_i : capacity of distribution center i to be designed,
- R_i : capacity of returned center i to be designed,
- $Y_i = \begin{cases} 1 & \text{if candidate center } i \text{ is located as distribution center} \\ 0 & \text{Otherwise} \end{cases}$
- $Z_i = \begin{cases} 1 & \text{if candidate center } i \text{ is located as returned center} \\ 0 & \text{Otherwise} \end{cases}$

2.3 Constraints

In Equation (1), the objective function minimizes total costs which are comprised of maintaining cost of distribution centers and returned centers, shipping cost of forward logistics and reverse logistics, processing cost of those exceeded products to customers, chance loss cost of those shortage products, and processing cost of those exceeded return products. $E[\bullet]$ represents the expected value because of the stochastic property on demand and returned product from customers. Constraints (2) guarantee that the number of products distributed from distribution center is no larger than its designed capacity. Constraints (3) guarantee that the designed capacity of a distribution center is no larger than its maximum capacity. Constraints (4) guarantee that the designed capacity of a product return center is no larger than its maximum capacity. Constraints (5) guarantee that all returned products are back to the product return centers. Constraints (6) means that all those shipped products are not zero.

Formulation (1)-(7) is a nonlinear mixed-integer programming. A genetic algorithm approach is developed together with the simulation of Monte Carlo method to give out an optimal or near optimal solution of the problem.

3 Genetic Algorithm Approach

If we order all decision variables in a sequence, the genetic representation of the problem’s solution can be divided into NW segments with $4+2NC$ genes in each segment. The first gene in each segment represents the processing capacity of the corresponding product return center and the second represents the capacity of the corresponding distribution center. The third gene indicates if it should be established as a product return center or not and the fourth indicates if it should be established as a distribution center or not. From genes 5 to $4+NC$ they encode the quantities of shipped products from the distribution center to customers in sequence. From genes $5+NC$ to $4+2NC$ they encode the ratios of returned products from customers to the product return center. In this way, it is easy to give out the genetic representation of the above problem. Figure 1 shows a simple example with 2 candidate centers and 2 customers.

In order to raise the feasibility of the encoded solutions, the alleles in the first and second genes are no larger than its maximum capacity and the sum of figures from 5 to $4+NC$ must be less than that in the second gene in the initial population generated randomly.

Candidate 1								Candidate 2							
46	458	1	1	233	225	0.9	0.9	928	847	0	0	0	0	0.1	0.1

Fig.1. Illustration of the genetic representation

Simply one-point crossover is adopted to improve all individuals in each generation. A mixed strategy with exchange mutation and perturbation mutation is designed to improve all individuals in each generation. If two randomly selected points are in the same gene segment of an individual, exchange mutation is operated on them. If one randomly selected point is fixed, only perturbation mutation is operated on it.

It is possible to generate infeasible individuals both in the initial population and in the process of genetic operations. Two modification strategies to cope with them are designed as follows:

Strategy I: increasing the designed capacity of the distribution center within its maximum capacity;

Strategy II: decreasing the shipping products to a customer according to the largest saving of shipment cost among customers associated with the distribution.

Simply, we take the objective value of Equation (1) as each individual’s fitness value after decoding from its genetic representation. In order to keep feasible individuals in the evolutionary process, the fitness value for the k^{th} individual vk is calculated as $eval(vk)=M \times TC(vk)$, where $TC(vk)$ is the total cost and

M is a penalty factor for those infeasible individuals ($M > 1$). In the calculation of the objective or Equation (1), the first four items can be calculated directly. However, because there are three single random variables in the 5th to 7th item of Equation (1), the expected value of these are taken as the costs. For the last item, because there are multiple random variables, we have to use Monte Carlo method to obtain its expected value on cost.

Simply we adopt the $(\mu+\lambda)$ -selection strategy. Finally, the reproduction-evaluation-selection circle consists of the whole procedure of the proposed GA approach.

4 Enumeration Method

To test the effectiveness of the proposed GA approach, a partly enumeration method could be designed as follows:

- Step 1:** select an alternative for the determination of distribution centers and product return centers;
- Step 2:** use the FMINCON function in Matlab to obtain the current optimal solution;
- Step 3:** if the enumeration is over, end; otherwise, go to **Step 1**.

FMINCON is a function in Optimization Toolbox of Matlab. It finds a constrained minimum of a scalar function of several variables starting at an initial estimate. For medium-scale optimization as this problem, FMINCON uses Sequential Quadratic Programming (SQP) method. For large-scale optimization, FMINCON use trust-region and preconditioned conjugate gradient method.

5 Numerical Analysis Method

The numerical tests are operated on a randomly generated examples by the proposed genetic algorithm approach written in VC++ and the partly enumeration method written in Matlab. The experiments are undertaken on a Pentium 4 PC with 128M of memory and the genetic parameters are set as follows: population size: 50, generation: 5000, crossover rate: 0.75, mutation rate: 0.1, Probability for modification Strategy I: 0.7, Probability for modification Strategy II: 0.2.

There are 3 candidate centers and 3 customers. The coordinate points of all three candidates are (60,137), (108, 26), and (127,143) respectively. The coordinate points of all three customers are (72, 95), (17, 138), and (18, 4) respectively. The demands of three customer are regarded in the uniform probability distribution $U[200, 250]$ and the returned products are in the uniform probability distribution $U[20, 25]$. All other parameters in the model are: $a= 0.5$, $b= 0.1$, $fw= 6000$, $uw= 100$, $fr= 2000$, $uw= 50$, $wr= 2000$, $MAX_SUPPLY= 500$,

Table 1. Best result of numerical experiment

Candidates	R	S	Z	Y	x_{i1}^f	x_{i2}^f	x_{i3}^f	x_{i1}^r	x_{i2}^r	x_{i3}^r
1	43	498	1	1	207	215	76	0.9	1.0	0.0
2	25	177	1	1	17	9	151	0.1	0.0	1.0

Table 2. Comparison between GA and enumeration method

Scale	GAs		Partly Enumeration		Efficiency	Error
	The Best	CPUs	The Optima	CPUs		
3×3	103641.44	14	98387	50.5930	3.61	5.34%
5×10	309697.34	97.8	302670	12891	131.81	2.32%

$MAX_REPAIR= 100$, $us= 300$, $uc= 500$, $ua= 500$. In the total 20 trials by the proposed GA approach, we obtain 20 alternatives in 280 seconds. Among them the largest total cost is 110516.17, the smallest is 103641.44, and the average is 10795.62. The best alternative in detail is listed in Table 1.

In order to further illustrate the effectiveness of the proposed GA approach, we test the GA approach and the partly enumeration method on the same problems. The results on comparison are listed in Table 2, from which it could be concluded that the partly enumeration method is able to obtain better alternative than the GA approach in results but the GA results are very close to the optima within the error of 6%. However, in running time the GA approach is much effective than the partly enumeration method on the same problem with 3 times faster for smaller scale problems and 100 times faster for larger scale problems. Actually, it is not suitable for the partly enumeration method to solve the problem with more than 5×10 scale.

6 Conclusion

This paper developed a nonlinear mixed-integer programming model to design a distribution system with reverse logistics. A GA approach was proposed in turn at aiming to provide a minimum total cost involving products return. Numerical analysis showed that the proposed GA approach is able to relatively efficiently obtain the near optimal solution for such kind of NP-hard problems.

Acknowledgements

This research work was partially supported by grant No. Y104171 from Zhejiang Provincial Nature Science Foundation and No. NX05GL07 from Zhejiang Provincial Philosophical & Social Science Foundation.

References

1. Dowlatshahi S.: Developing a theory of reverse logistics. *Interfaces* **30** (2000) 43-55
2. Stock J.R.: Reverse logistics, White Paper, Oak Brook. IL: Council of Logistics Management (1992)
3. Krumwiede D.W. and Sheu C.: A model for logistics entry by third-party providers, *Omega*, **30**(2002)325-333
4. Mirchandani P.B. and Francis R.L.: *Discrete Location Theory*. Wiley, New York (1989)
5. Min H., Ko H.J., and Ko C.S.: A genetic algorithm approach to developing the multi-echelon reverse logistics network for product returns. *Omega*, forthcoming issue (2005)

Multi-objective Evolutionary Design and Knowledge Discovery of Logic Circuits with an Improved Genetic Algorithm

Shuguang Zhao, Licheng Jiao, and Jun Zhao

School of Electronic Engineering, Xidian University,
Xi'an 710071, China
Sgzhaoy@xidian.edu.cn

Abstract. To improve evolutionary design of circuits in efficiency, scalability and optimizing capability, a genetic algorithm based approach was proposed. It employs a gate-level encoding scheme supporting flexible changes of functions and interconnections of comprised logic cells, a multi-objective evaluation mechanism of fitness with weight-vector adaptation and circuit simulation, and an adaptation strategy for crossover probability and mutation probability to vary with individuals' diversity and genetic-search process. It was validated by experiments on arithmetic circuits, obtaining circuits with expected functions, novel structures, and higher performances in gate usage and operating speed as compared with the results of both conventional and evolutionary approaches. Moreover, by studying the circuits evolved for problems of increasing scales, some novel, efficient and generalized principles have been discerned, which are easy to verify but difficult to dig out by human experts with existing knowledge.

1 Introduction

With ever-growing scales and complexities of electronic circuits, automated design of circuits becomes even more important and challenging. Evolutionary design of circuits (EDC), a main branch of evolvable hardware [1,2] promises to realize automated design of circuits by applying bio-inspired techniques especially GA [3] to circuit design tasks. As to logic or digital circuits, EDC is mainly implemented at a function-level [1,2,4] or a gate-level [5-8]. A gate-level evolution usually takes logic gates as basic units and evaluates individual's fitness through software simulation, which endows it with some advantages such as applicability and analyzability.

Till now, most relevant works reported were concentrated on combinational especially arithmetic circuits, mainly for the sake of finding out novel or efficient building-blocks. Among them, the techniques and results reported by Koza *et al* [5], Miller *et al* [6,8] and Coello Coello *et al* [7] are outstanding, but they were weak in meeting the demand of multi-objective optimization involving circuits' operating speed.

We investigated a novel approach to multi-objective gate-level evolution of larger scale circuits. It is built on the bases of an EGA framework and embedded with a few measures to improve its performances, as introduced below.

2 Circuit Model and Encoding Scheme

As depicted in Fig. 1, the circuit model adopted is an array of C rows by L columns of interconnected logic units, which are uniform and multifunctional. For most logic circuits, a two-input with 4 selectable functions, AND, OR, NOT and XOR (eXclusive-OR), which are commonly used and logically self-contained, is feasible. For some sequential ones, an additional memory function is also required. This model is universal and convenient to represent either sequential or combinational circuits, depending on internal feedbacks are allowed or not [2]. All units including the virtual units for external inputs are orderly numbered, and each unit is encoded as a character string $[IS1_{CN}, IS2_{CN}, TS_{CN}]$. Where CN is its sequence number, $CN=i+j \cdot C$, $IS1_{CN}$ and $IS2_{CN}$ are sequence numbers of the two units output to it respectively, and TS_{CN} indexes its function selected. By linking all units' encoding strings with that of array's outputs, $[OS_1, \dots, OS_q]$, the array's encoding, i.e. a chromosome is formed as a net-list,

$$[IS1_1, IS2_1, TS_1] \cdots [IS1_C, IS2_C, TS_C] [OS_1, \dots, OS_q] \tag{1}$$

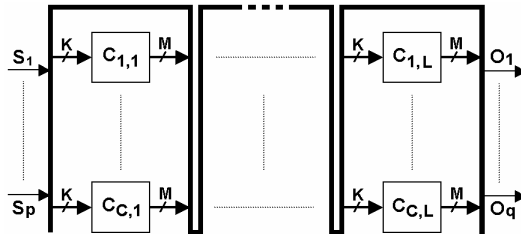


Fig. 1. The model for gate-level evolution

3 Dynamic Multi-objective Evaluation

To design circuits is by nature a difficult multi-objective optimization problem as

$$\text{Maximize } f(X) = (f_1(X), f_2(X), \dots, f_n(X)), \quad X \in \Omega \tag{2}$$

To solve the problems efficiently and simply with a GA, a well-known fitness function in the form of *sum of weights* is introduced and modified, which converts the problem into its following single-objective equivalent

$$\text{Maximize } \text{Fit}(X) = \sum_{i=1}^n w_i \cdot \text{Fit}_i(X) \tag{3}$$

Where $\text{Fit}_i(X)$ denotes the normalized objective function related with $f_i(X)$; w_i denotes the corresponding weight factor and satisfies $\sum w_i(0)=1$. To let the GA give attention to all objectives, the weight factors $\{w_i\}$ is enabled to change dynamically in a way similar to that of the Back-Propagation Learning Algorithm in ANN

$$w_i(t+1) = \alpha \cdot w_i(t) + (1-\alpha) \cdot [2/N - \overline{\text{Fit}_i(t)} / \sum_{j=1}^N \overline{\text{Fit}_j(t)}] \tag{4}$$

Where $\alpha \in [0,1]$ is a constant suggested 0.8 by experiments; $\overline{Fit}_i(t)$ denotes the average fitness of all individuals in the current population. Thus, if $\sum w_i(0)=1$, then $\sum w_i(t)=1$ will hold forever as expected. And, the more an objective is optimized, the less its weight factor and the consequent optimizing pressure on it would be, and vice versa. But the initial values of weight factors are still meaningful to express user-preferences, although it is mostly assigned $w_i(0)=1/n$ for convenience, $1 \leq i \leq n$.

For problems of gate-level evolution, the first design target is to make circuits fully comply with the expected behaviors, which is usually described with a truth table or a state table. It is natural to measure a circuit's operation quality by a ratio of its Number of Matched Operations (*NMO*) to the Total Number of defined Operations (*TNO*),

$$Fit_1 = NMO/TNO \quad (5)$$

Where an operation corresponds to a row of the specified truth table or state table, and it will be scored 'correct' or 'error' by a simulation sub-program designed for figuring out every unit's logic level and *active time* for every operation. The second design target is to maximize the efficiency of resource usage, estimated with the Number of *Unused Gates* (*NUG*) have no effect on the candidate's behavior, e.g. an *AND* gate or an *OR* gate with its two inputs connected together, even numbers of *NOT* gates linked in serial, a gate with its output not referred, etc. By recognizing all *Unused Gates* with the simulation results and their features, the corresponding objective function was computed as

$$Fit_2 = \exp(-k_1 \bullet NUG) \quad (6)$$

The third design target is to maximize the operating speed of a circuit, which can be estimated with its Maximal Propagation-Delay (*MPD*) derived from the simulation results involving every unit's *active time*. The objective function was defined as

$$Fit_3 = \exp(-k_2 \bullet MPD) \quad (7)$$

Where, K_1 and K_2 are user-defined constants. As a result, the fitness function to synthetically evaluate the candidate was formed as

$$Fit(t) = \sum_{i=1}^3 w_i(t) \bullet Fit_i(t) \quad (8)$$

4 Adaptation of Genetic Parameters

To improve the GA's performances, its crossover probability P_c and mutation probability P_m were allowed to self-adapt to the individuals' diversity and the genetic-procedure, which was estimated by dividing the current generation number t by the specified maximal one t_{max} . The individuals' concentration degree was estimated as

$$f_d(t) = \overline{f}(t) / [\varepsilon + f_{max}(t) - f_{min}(t)] \quad (9)$$

Where $f_{max}(t)$, $f_{min}(t)$ and $\overline{f}(t)$ are maximal, minimum and average fitness of all individuals of the current generation respectively. As $f_d(t)$ satisfies $0 < f_d(t) < +\infty$, and it

will simultaneously vary with the distribution or diversity of individuals but in a reverse direction, P_c and P_m were enabled to adapt in the following manners

$$P_c = \begin{cases} P_{c0} / f_d(t) & t < t_0 \\ P_{c0} \bullet e^{-k_3 \bullet (t-t_0) / t_{max}} / f_d(t) & t_0 \leq t \leq t_1 \\ P_{c0} \bullet e^{-k_3 \bullet (t_1-t_0) / t_{max}} / f_d(t) & t_1 \leq t \leq t_{max} \end{cases} \quad (10)$$

$$P_m = \begin{cases} P_{m0} \bullet f_d(t) & t < t_0 \\ P_{m0} \bullet e^{-k_4 \bullet (t-t_0) / t_{max}} \bullet f_d(t) & t_0 \leq t \leq t_1 \\ P_{m0} \bullet e^{-k_4 \bullet (t_1-t_0) / t_{max}} \bullet f_d(t) & t_1 \leq t \leq t_{max} \end{cases} \quad (11)$$

Where, P_{c0} , P_{m0} , t_0 , t_1 , k_3 and k_4 are user-defined constants. According to equation (10) and equation (11), P_c and P_m will slowly decrease during the evolution process while responding to the changes of individuals' diversity. With such a self-adaptation mechanism simulating some principles of bionomics and developmental biology [9-10], P_c and P_m can be expected to be suitable for most cases of an evolution process.

5 Experimental Results and Discussions

With the approach proposed above, evolutionary design experiments on some benchmark problems of increasing scales [5-8], were implemented successfully.

An n -bit digital multiplier is a combinational circuit that *output the product of two groups of n -bit binary numbers*. A 3-bit multiplier evolved is depicted in fig. 2, where GN stands for gate number and DN for delay number. It is 10.7% more efficient and 20% faster than the best one designed by a human expert, and it is in fact as good as the most efficient one evolved by Miller *et al* [6,8], which consists of 21 standard gates and 2 nonstandard one-input-inverted AND gates, which is actually a combination of 2 gates. As to a 4-bit multiplier that is rather difficult even for human experts to design, our result depicted in fig. 3 is much better than that designed by human experts (with GN=64, DN=24) and that evolved from a conventional design by Miller *et al* [6,8], which consists of 57 gates including 10 nonstandard AND gates and features DN=20. All these circuits feature wondrous reuses of inner outcomes, arguing our approach is effective and it surpassed its congeners and human experts.

Besides, it was exciting that we have inferred some novel principles from the circuits evolved for problems of increasing scales by using Case-Based Reasoning (CBR) techniques. For a binary addition that is the core of a binary multiplication, a set of expressions that efficiently derive the carry from addends have been identified as

$$CF_{n+1} = CF_n \oplus [I_{n+1} \bullet (I_1 \oplus \dots \oplus I_n)] \quad (12)$$

Where CF_n denotes the least-bit of the carry derived from n bits of addend, I_1, \dots, I_n ; CF_{n+1} denotes the least-bit of the carry derived from $n+1$ bits including I_{n+1} , $n \geq 1$. As $CF_1=0$, it is convenient to derive and efficiently implement CF_n for binary additions of arbitrary bits by repeatedly applying Equation (12).

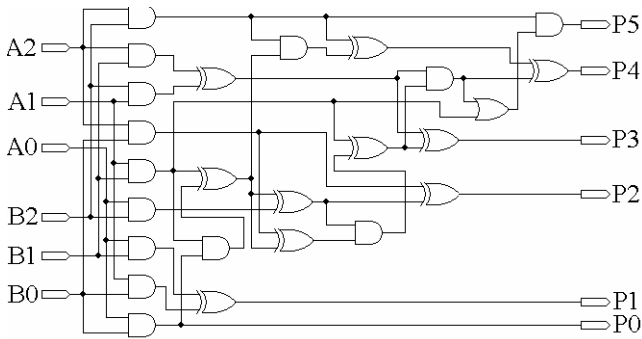


Fig. 2. A 3-bit multiplier evolved (GN=25, DN=9)

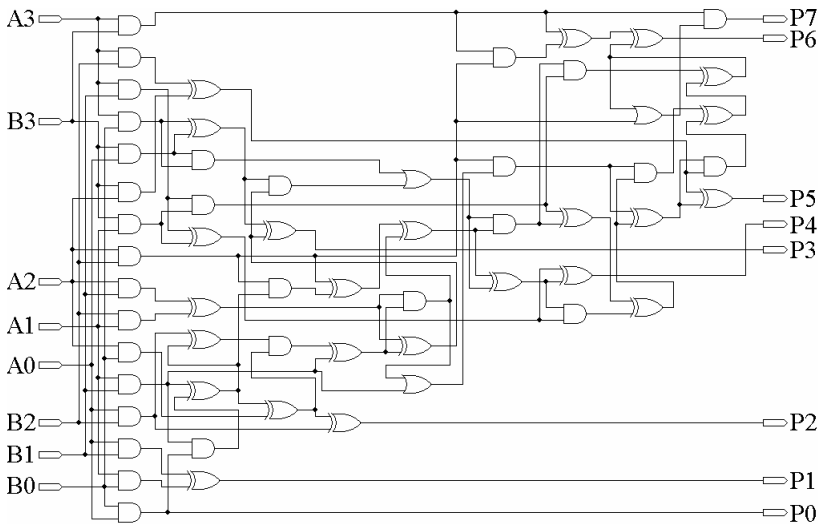


Fig. 3. A 4-bit multiplier evolved (GN=58, DN=18. GN=55 if input-inverted AND gates used)

A similar result has also been obtained for even-parity checkers that *output '1' iff inputs contain nonzero even numbers of '1'*

$$F_{n+1} = I_{n+1} \cdot (I_1 + \dots + I_n) \oplus F_n \quad (13)$$

Where F_n denotes the output of a n -bit even-parity checker with n inputs, I_1, \dots, I_n ; F_{n+1} denotes the output of a checker with $n+1$ bits of input including I_{n+1} ; $F_1=0$.

Although the principles extracted such as equation (12) and equation (13) are easy to verify with knowledge of Boolean algebra, they are rather difficult for human experts to dig out with conventional approaches. These results argue that the approach proposed is helpful to acquisition and discovery of relevant knowledge.

6 Conclusions

A novel approach for evolutionary design and knowledge discovery of logic circuits was proposed and validated in this paper, showing preliminary but promising results. In future we will study how to interactively enhance evolutionary design and knowledge discovery in order to solve large-scale problems and acquire new knowledge.

Acknowledgements

This work was partially supported by National Natural Science Foundation of China under grant No. 60133010 and grant No. 60374063. It was also granted financial support from China Postdoctoral Science Foundation.

References

1. Yao X., Higuichi T.: Promises and Challenges of Evolvable Hardware. *IEEE Transactions on Systems Man and Cybernetics-Part C*, 1999, Vol. 29, No. 1, pp. 87-97
2. Zhao, S.G.: Study of the Evolutionary Design Methods of Electronic Circuits. PhD. dissertation, Xidian University, China, 2003
3. Goldberg D. E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, 1989
4. Zhao, S.G., Yang W. H.: Intrinsic Hardware Evolution Based on a Prototype of Function Level FPGA. *Chinese Journal of Computers*, 2002, Vol. 25, No. 6, pp. 666-669
5. Koza J. R.: *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MIT Press, 1992
6. Vassilev V. K., Job D., Miller J. F.: Towards the Automatic Design of More Efficient Digital Circuits. *Proceedings of the Second NASA/DOD Workshop on Evolvable Hardware (EH'00)*, PaloAlto, pp. 151-160
7. Coello Coello A. C., Christiansen A. D., Aguirre A. H.: Use of Evolutionary Techniques to Automate the Design of Combinational Circuits. *International Journal of Smart Engineering System Design*, 2000, Vol. 2, No. 4, pp. 299-314
8. Miller J. F., Job D., Vassilev V. K.: Principles in the Evolutionary Design of Digital Circuits: Part I. *J. of Genetic Programming and Evolvable Machines*, 2000, Vol.1, No.1, pp. 8-35
9. Shang Y. C., Cai X. M.: *General Bionomics*. Peiking University Press, Peiking, 1992
10. Gilbert S. F.: *Developmental Biology*. 6th edition, Sinauer Assoc. Inc., Sunderland, 2000

Robust Mobile Robot Localization Using an Evolutionary Particle Filter

Bo Yin, Zhiqiang Wei, and Xiaodong Zhuang

Computer Science Department of Ocean University of China, Postfach 26 60 71,
Qingdao, China
{ybfirst, weizhiqiang}@ouc.edu.cn

Abstract. The application of the auxiliary particle filter to the robot localization problem is considered. The auxiliary particle filter (APF) is an enhancement of the generic particle filter. However, APF suffers from the impoverishment problem and needs a large number of particles to represent the system posterior probability density function. An evolutionary computing method, the genetic algorithm is introduced into APF to remove early convergence yet improves the quality of potential solutions. Experiment results show that the evolutionary APF algorithm needs fewer particles and is more precise and robust for mobile robot localization in dynamic environment.

1 Introduction

Mobile robot localization is the problem of estimating a robot's pose (location, orientation) relative to its environment. It plays a pivotal role in various successful mobile robot systems. Occasionally, it has been referred as "the most fundamental problem to providing a mobile robot with autonomous capabilities" [1].

Many existing approaches rely on Kalman filter for robot state estimation [2]. Although the Kalman filter constitutes a powerful framework, its applicability is restricted by the assumption that the state vector is Gaussian distributed. In order to represent non-linearity and non-Gaussian characteristics better, particle filter was proposed in [3]. Particle filter out-performs the EKF for non-linear systems and has been successfully used in robotics. However, particle filter is limited by its need for a large number of samples to represent the PDFs making it computational expensive. Particle filter also suffers from the so-called particle impoverishment problem.

In this paper, the auxiliary particle filter (APF) [4] will be applied in mobile robot localization. An adaptive search method in evolutionary computing, the genetic algorithm (GA), is adopted to reduce the number of particles while collapse of particles to a single point will be avoided. This algorithm will not only improve the sampling efficiency, but also provide a good representation of the PDF through a specific selection scheme in GA and avoid pre-mature convergence of the particles by applying genetic crossover and mutation operations [5]. The genetic operations will ensure that the particles will be sufficiently shuffled beyond the high concentration area, yet maintaining their quality as potential solutions to the estimation.

2 The Robot Localization Problem

A convenient way to analyze the robot localization problem is through a state-space approach [6]. The robot is regarded as a partially observable Markov decision process with a three-dimensional state vector $x = [x, y, \theta]^T \in X$, i.e. the position and orientation of the robot. We assume an initial distribution $p(x_0)$ at time $t = 0$, and a given stochastic *transition* model $p(x_{t+1}/x_t, u_t)$ for an action (control signal) u_t that is issued at time t and brings the robot stochastically from state x_t to x_{t+1} .

We assume that in each time step t the robot observe a sensor vector $y_t \in Y$, which is related to the robot state through a stochastic *observation* model $p(y_t/x_t)$. The observations $\{y_t\}$ is assumed to be conditionally independent given the states $\{x_t\}$. Robot localization amounts to estimating in each time step t a posterior density $p(x_{t+1}/y_t)$ over the state space X , that characterizes the *belief* of the robot about its current state at time t given its initial belief $p(x_0)$ and the sequence of observations $y_1 \cdots y_t$. Using the Bayes rule, this posterior density for time $t+1$ reads to proportionality

$$p(x_{t+1}/y_{t+1}) \propto p(y_{t+1}/x_{t+1})p(x_{t+1}) \quad (1)$$

where the prior density $p(x_{t+1})$ corresponds to the propagated posterior from the previous time step

$$p(x_{t+1}) = \int p(x_{t+1}/x_t)p(x_t/y_t)dx_t \quad (2)$$

where we use the Markov assumption that the past has no effect beyond the previous time step. The above two formulas constitute an efficient iterative scheme for optimal (Bayesian) filtering. However, when the transition and observation models are nonlinear and/or non-Gaussian, it is not possible to calculate the posterior density analytically. If this is the case, the problem has to be tackled by using approximations or simulations. This is exactly what particle filters are designed for.

3 The Auxiliary Particle Filter

As shown in the previous section, mobile robot localization is a Bayesian filtering problem, which amounts to estimating the state of the robot at current time-step t , given knowledge about the initial state and all measurements $y_1 \cdots y_t$ up to the current time. Particle filter provides an efficient means to tackle this problem. However, one problem concerned with all particle filters is how to optimally sample from the posterior. An elegant solution to this problem has been given by Pitt and Shephard[4] under the name ‘auxiliary particle filter’. The algorithm is as follows: in order to sample from the posterior $p(x_{t+1}/y_{t+1})$, just insert the likelihood inside the mixture

$$p(x_{t+1}/y_{t+1}) \propto \sum_{i=1}^I \omega_i^j p(y_{t+1}/x_{t+1}) p(x_{t+1}/x_t^i) \quad (3)$$

and treat the products $\omega_i^j p(y_{t+1}/x_{t+1})$ as the component probabilities in order to sample from the mixture. Because the likelihood $p(y_{t+1}/x_{t+1})$ in the above product involves the unobserved state vector x_{t+1} , an approximation of the mixture is

$$\hat{p}(x_{t+1}/y_{t+1}) \propto \sum_{i=1}^I \omega_i^j p(y_{t+1}/\mu_{t+1}^i) p(x_{t+1}/x_t^i) \quad (4)$$

where μ_{t+1}^i is any likely value associated with the i -th component transition density $p(x_{t+1}/x_t^i)$, for example its mean. After a set of particles x_{t+1}^j have been sampled from the transition density $p(x_{t+1}/x_t^i)$ with probability $\omega_i^j p(y_{t+1}/\mu_{t+1}^i)$, $j=1, \dots, I$, their weights are set proportional to

$$\omega_{t+1}^j \propto \frac{p(y_{t+1}/x_{t+1}^j)}{p(y_{t+1}/\mu_{t+1}^j)} \quad (5)$$

where μ_{t+1}^j is the associated likely value of the component $p(x_{t+1}/x_t^j)$ in (4) from which the particle j was sampled.

The auxiliary particle filter is particularly efficient, since it requires only the ability to sample from the transition model and evaluate the likelihood function $p(y_t/x_t)$. However, in the robot localization problem, the auxiliary particle filter will also need a large number of particles to represent the PDF and suffer from the sample impoverishment problem to some degree.

4 Evolutionary Auxiliary Particle Filter

In order to reduce the limitations of particle filter, the Genetic Algorithm is a promising alternative. We proceed by retaining the auxiliary particle filter (APF) algorithm, but operate on the particles from the GA perspective.

4.1 The Genetic Selection

Here, we use the selection process aiming to derive desirable characteristics in the particles. The selection process should be such that diversity on particles should be maintained as much as possible. After the selection process, we aim at resupplying new particles at the vicinities of potential solutions. In many GA literatures, it has been pointed out that the proportional selection scheme widely used in particle filters is not efficient. This is due to the fact that, the scheme is biased by the scale of weights of particles. The selectivity could be much reduced if the weights are almost the same. In this work, we use the selection schemes proposed in [7], i.e. Tournament Selection or Truncation selection.

4.2 Genetic Crossover and Mutation

Now, we use the genetic Crossover and Mutation operators to fill the gaps between the particles and resupply new particles as potential solutions. Firstly, we use two particles $(x_t^i, \omega_t^i), (x_t^j, \omega_t^j)$ selected randomly to perform the Crossover operation. The new particles that replace the original ones are given by

$$\begin{cases} x_t^{in} = \xi x_t^i + (1-\xi)x_t^j \\ x_t^{jn} = \xi x_t^j + (1-\xi)x_t^i \end{cases} \quad (6)$$

where ξ is a uniform random number in range $[0,1]$.

It is also noted that the range of the new particles is the same as the original particles, but fills in the gap between particles. To overcome the limitation in initially fixed particles, we use the mutation operator to operate on the resulting particles. A particle is then altered as

$$x_t^n = x_t + r \times \sigma^2 \quad (7)$$

where r is a Gaussian random number of zero mean and variance σ^2 .

Moreover, in order to exercise control on the operators, we employ constraints through the crossover probability p_c and mutation probability p_m , which are set the same value as in [7]. These constraints are given as follows:

$$p_c = 0.9, \quad p_m = 0.1, \quad \sigma^2 = 0.02(1+1/n) \quad (8)$$

where n is the iteration count, hence σ^2 decreases against iterations.

4.3 The Evolutionary Auxiliary Particle Filter Algorithm

The evolutionary auxiliary particle filter algorithm (EAPF) is described as follows:

- (1) Initialization (the same as APF)
- (2) Sampling phase (the same as APF)
- (3) The evolution phase
 - a) For $j=1, \dots, I$, perform *Tournament or Truncation* selection;
 - b) For $j=1, \dots, I$, perform *Crossover and Mutation* operation;
- (4) Normalize the weights
- (5) Output phase
- (6) For $t=t+I$, go to (2)and begin a new iteration

5 Experiment Results and Algorithm Performance Analysis

In this section, we will illustrate the performance of the EAPF algorithm applied on our Guiding II mobile robot. The robot is equipped with 12 sonar sensors and 1 color CCD camera. Using the sensor measurements, the EAPF algorithm and the generic particle filter (PF) are employed to perform robot localization in our office environment, which is shown in Fig.1.

At the start, the initial belief $p(x_0)$ was set to be a uniform distribution over the whole environment. Then, we used EAPF and PF with different sample sizes to perform robot localization respectively. The robot was programmed to follow the path shown in Fig.1 starting from point A, going through point B to point C, then coming back to A for a number of times. The distance between the locations was about 30m. Each algorithm with different size of samples was used to perform 10 times of path traversals respectively. In Fig.2 we plot the average localization errors of the two algorithms with different sample sizes.

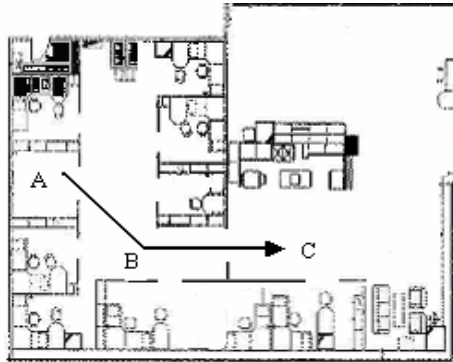


Fig. 1. The environment of the office

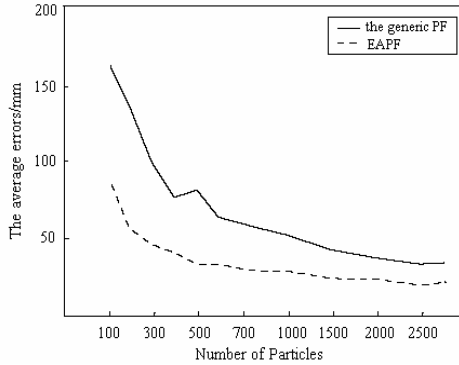


Fig. 2. Average errors with different sample sizes

From Fig.2, we can see that EAPF performs out of PF. EAPF only needs a smaller number of particles to obtain a higher localization precision. For example, EAPF only needs 300 samples to achieve the precision that PF needs about 1700 samples to obtain. The average time to localize the robot once by the two algorithms was 18ms and 24ms respectively.

Comparing the experiment results, we can see that the introduction of the genetic algorithm improves the performance of the auxiliary particle filter. The genetic operations make the auxiliary particle filter maintain the particles' diversity as much as possible, and perform optimal selection on the particles at the same time. These make

the particles sufficiently shuffled beyond the high concentration area while the particle impoverishment problem is avoided. Therefore, the evolutionary auxiliary particle filter can generate good quality particles and represent the desired PDF better. As a result, the precision and robustness of robot localization are improved greatly.

6 Conclusion

In this paper, we have proposed a variation to the auxiliary particle filter using evolutionary computing techniques. Three genetic operations are introduced into auxiliary particle filter to improve its performance. These genetic operations can overcome the particle impoverishment problem and make particles move towards regions with large value of PDF, so the EAPF can represent the desired PDF better. Experiment results show that EAPF needs fewer particles and is more precise and robust than PF algorithm in mobile robot localization. It is important to apply this algorithm in more challenge situations, such as the SLAM problem, so as to further test its performance.

References

1. Sebastian Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert: Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 128(2001) 99-141
2. S. I. Roumeliotis and G.A. Bekey: Bayesian estimation and Kalman filtering: A Unified Framework for Mobile Robot Localization. In *proc. IEEE conference on Robotics and Automation*, San Fransisco(2000) 2985-2992
3. N.Gordon, D. Salmond, and A. Smith: Novel approach to nonlinear/non-gaussian Bayesian state estimation. *IEEE proceedings-F*, 140(1993) 107-113
4. M. K. Pitt and N. Shephard: Filtering via simulation: Auxiliary particle filters. *Journal of American Statistical Association*, 94(1999) 590-599
5. T. Higuchi. Monte Carlo filter using the genetic algorithm operators. *J. Statist. Simul.* , 59(1997) 1-23
6. Nikos Vlasssis, Bas Terwijn, and Ben Kröse: Auxiliary Particle Filter Robot Localization from High-Dimensional Sensor Observations. In *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Washington, DC(2002) 7-12
7. N.M. Kwok and S. Kwong: Mobile Robot Simultaneous Localization and Mapping Using a Evolutionary Particle filter. In *Proceedings of Genetic and Evolutionary Computation Conference (GECCO)*, Chicargo(2003) 178-185

Hybrid Genetic Algorithm for Solving the Degree-Constrained Minimal Bandwidth Multicast Routing Problem*

Yun Pan^{1,2}, Zhenwei Yu¹, and Licheng Wang³

¹ Dept. Computer Science, China University of Mining and Technology-Beijing

² Institute of Computing and Technology, Chinese Academy of Science

³ Dept. Computer Science, Shanghai Jiao Tong University

{pany, zwyu}@cumtb.edu.cn, wanglc@sjtu.edu.cn

Abstract. In this paper, we propose a programming model for the degree-constrained minimal bandwidth multicast routing problem in overlay networks, and design a hybrid genetic algorithm to solve it. The algorithm combines heuristic searching and genetic searching together for global optimal seeking. The complexity analysis and numerical experiments suggest our proposed model and algorithm are practical and efficient.

1 Introduction

IP multicast [1] is an efficient way to economize the bandwidth resources. But IP multicast has not been put into practice in large scale due to some drawbacks which are embedded into IP multicast mechanism itself [2]. Recently, enlightened by the idea of end-to-end [3] and impelled by requirements of various kinds of services, many researchers focus on implementing multicast function on application layer [4, 5]. At present, there are two solutions with the intention to answer the question of how to construct multicast application systems: Application layered multicast [4] and overlay multicast [5]. Application layered multicast focuses on building multicast tree over the virtual network which consists of end nodes and the multicast function is eventually afforded by end nodes. Overlay multicast inclined to construct multicast tree over the virtual network which consists of both end nodes and median proxy nodes. Consequently the multicast function is afforded by end nodes and median proxy nodes in overlay networks. Both of the two above solutions have the following advantages: needless to conduct revolution over the Internet architecture, easiness of service deployment, strong adaptive ability, robustness, and so forth.

However, comparing with application layered multicast, overlay multicast has been paid more attention because of the following traits [6]: (1) The performance of overlay multicast surpasses that of application layered multicast, and even

* Partially supported by the National Research Fund for the Doctoral Program of Higher Education of China under Grant No. 20030290003.

matches the performance of IP multicast; (2) Overlay multicast is more suitable for large scale group communication; (3) Overlay multicast is a long-term solution, while application layered multicast is just a temporary one which is suitable for solving problems which need emergent setting-modification.

How to find proper multicast tree in the overlay network is a key issue about overlay multicast problem. On application layer, different nodes may have different functions. Some nodes possess the multicast function while others only possess the forward function. Under this situation, the problem of link reusing arises. What is more, each interior node may possess different ability to access user nodes. In order to distinguish these two cases, we employ two concepts: degree-constraint and multicast vector. The former denotes the number of permitted access links, while the latter represents the distribution of multicast nodes in the overlay network. Then, we propose a programming model for the degree-constrained minimal bandwidth (DCMB) multicast routing problem in overlay network, and design a hybrid genetic algorithm for solving it.

2 Degree-Constrained Minimal Bandwidth Multicast Routing

To calculate the consuming of bandwidth in the overlay multicast routing, we should consider a special case as shown in Figure.1(a). The bandwidth between the nodes 1 and 3 should be computed twice because the node 3 has no multicast function while the bandwidth between the nodes 1 and 2 should be computed only once (See Figure.1(b)).

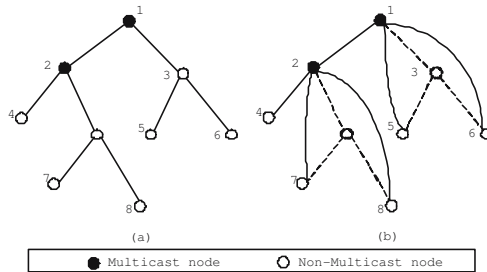


Fig. 1. Illustration of calculation of bandwidth in overlay network

Suppose the network topology be an undirected graph $G = (V, E)$, where V denotes the set of underlying nodes, and E the set of physical links (also termed as edges). For $\forall v \in V$, $d_{max}(v) \in \mathbb{Z}^+$ denotes its degree-constraint. For $\forall (u, v) \in E$, $e(u, v) \in \mathbb{R}^+$ denotes the bandwidth of the link. For a multicasting session pair (s, D) , where s and D denote the source node and the set of sink nodes respectively, the degree-constrained minimal bandwidth multicast routing problem(DCMB) can be modeled as follows:

$$\begin{cases} \min_{T \subset G} & B(T) \\ s.t. & d_T(v) \leq d_{max}(v), \forall v \in V_T \end{cases} \quad (1)$$

where V_T and $d_T(v)$ denote the set of nodes in T and the degree of the node v in T , respectively. The function $B(T)$ is defined by

$$B(T) = \sum_{i \in V_T} c(i). \tag{2}$$

Before give the definition of the function $c(i)$ in above formula, we need to introduce some useful definitions at first.

Definition 1. *The multicast tag function, which represents whether a node possesses the multicast function, is defined by*

$$x(i) = \begin{cases} 0, & \text{If node } i \text{ does not possess the multicast function,} \\ 1, & \text{If node } i \text{ possesses the multicast function.} \end{cases} \tag{3}$$

Without loss of generality, suppose that each multicasting source node possesses the multicast function, i.e., set $x(s) = 1$. In the following description of algorithm, we use a multicast tag vector \mathbf{X} representing the multicast tag function, i.e., $X_i := x(i)$.

Definition 2. *For $i \in V_T$, the antecedent multicast node $a(i)$ denotes the nearest antecedent node which possesses the multicast function in T .*

For example, from Figure.1(a), we have $a(2) = a(4) = a(7) = 2$. Clearly, $a(i) = i$ iff $x(i) = 1$.

Definition 3. *The antecedent link tag function, which represents whether an edge belongs to a path from some node to its antecedent multicast node, is defined by*

$$y(i, u, v) = \begin{cases} 1, & \text{If edge } (u, v) \text{ belongs to some path from } a(i) \text{ to } i, \\ 0, & \text{Otherwise.} \end{cases} \tag{4}$$

Now, we give the definition of the aforementioned function $c(i)$, which represents the total bandwidth from the node i to its antecedent multicast node $a(i)$:

$$c(i) = \sum_{u \in V} \sum_{v \in V} e(u, v) y(i, u, v) \tag{5}$$

The degree-constrained minimal bandwidth multicast routing problem, defined by model (1) is an instance of constrained minimal Steiner problem which is a well-known NP-hard problem and has not been solved effectively. We will design a hybrid genetic algorithm to solve the DCMB problem in the following sections.

3 Hybrid Genetic Algorithm for the DCMB Model (1)

The problem of representing and coding of multicast trees is the first issue we must deal with. Multicast trees can be derived from spanning trees by pruning

methods, while the representation of spanning trees is a well-developed topic. For a given spanning tree T' , the corresponding multicast tree T can be pruned out by removing the non-destination leaf nodes in T' . Traditional heuristic algorithms for spanning-tree concerned optimization problems always take edge weights as heuristic information. This kind of algorithms inclined to be trapped into the local optimal solutions. Therefore, we consider a hybrid strategy: let different heuristic information to induce different local optimal solutions, and then introduce genetic operations into the population of heuristic information with the expectation of finding a global optimal solution within an acceptable time limitation.

We adopt a continuous genetic node-weight strategy to represent the heuristic information for spanning trees. Suppose $\mathbf{w} \in [0, 1]^n$, ($n = |V|$), be the genetic weight matrix assigned to nodes of the network. Set $e'(i, j) = h(e(i, j), w_i, w_j)$ as the genetic cost matrix assigned to edges of the network. Then, the minimal spanning tree T' can be determined by Prim algorithm or Kruskal algorithm [7]. The corresponding multicast tree T can also be derived from T' by pruning procedure. Finally, the fitness-value of the chromosome \mathbf{w} , denoted by $f(\mathbf{w})$, is defined by the sum of original weights of nodes in T . In short, given a chromosome \mathbf{w} , we can obtain a multicast tree, denoted by $T(\mathbf{w})$. Then, if $T(\mathbf{w})$ satisfies the degree-constraint condition, i.e., $d_{T(\mathbf{w})}(v) \leq d_{max}(v)$ holds for every $v \in V_{T(\mathbf{w})}$, the fitness value $f(\mathbf{w})$ can be defined by $f(\mathbf{w}) = \frac{1}{1+B(T(\mathbf{w}))}$; Otherwise, set $f(\mathbf{w}) = 0$. Now, the pair $(\mathbf{w}, f(\mathbf{w}))$ represents the evolutionary evaluation environment. In general, we can choose \mathbf{w} at random in the interval $[0,1]$, and set the weight transform function h as the production form or sum form of $e(i, j), w_i$ and w_j .

Based on aforementioned dissection, we design a hybrid genetic algorithm, DCMB-GA, to solve the DCMB problem model (1): generate each chromosome $\mathbf{w} \in [0, 1]^n$ at random; employ the improved roulette-wheel selecting operator, the uniform multi-points crossing operator, and the gene mutating operator, as well as the gene piece mutating operators (including the shifting operator and the reversing operator); adopt elitist policy such that the most excellent individual of current generation will be allowed to pass to the next generation without being destroyed.

The complexity of the decoding algorithm is bounded by $O(n^2)$, while the encoding method directly generates a vector at random as a chromosome. The most time-consuming step in fitness-evaluating procedure is that computes the function $y(j, u, v)$ (See Definition 3), thus the complexity of is bounded by $O(n^3)$. Therefore, given the population scale pop and the maximal evolution generation $maxG$, the total complexity of DCMB-GA is $O(maxG \cdot pop \cdot n^3)$.

4 Simulations and Analysis

By using simulation, we analyze the performance of DCMB-GA algorithm from two aspects: the convergency and the quality of the optimal solutions. The population scale pop and the maximal evolution generation $maxG$ are set to 30 and 100

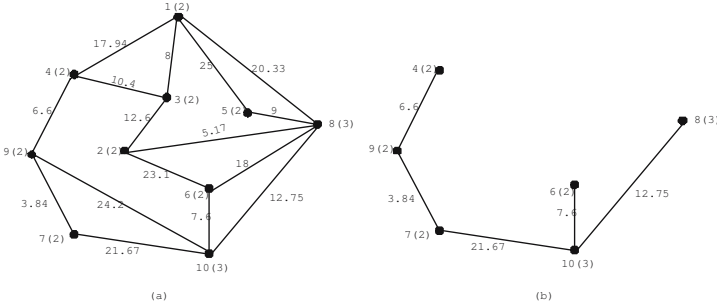


Fig. 2. (a) The topology of experiment network; (b) The optimal multicast tree

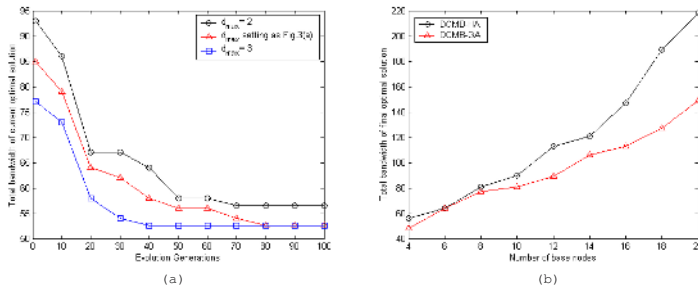


Fig. 3. (a) The convergency of DCMB-GA; (b) The least cost vs. number of base nodes

respectively. In order to comparison the performance of the proposed algorithm, we also design a heuristic algorithm¹, DCMB-HA, for the model (1).

For convergency analysis, we adopt the same topology structure with [8] as listed in Figure.2(a). Suppose $s = 4$ and $D = \{6, 7, 8\}$. The degree-constraint is given in a pair of parentheses beside each node. The weight marked on each arc represents *bandwidth* (while in [8] it is *delay*). The output of DCMB-GA algorithm is just the same optimal solution in [8] (See Figure.2(b)). But the total bandwidth is 74.13 instead of 52.46, since node 10 doest not possesses multicast function, resulting in counting the bandwidth of the reused link between node 7 and node 10 for twice.

From Figure.3 one can see that the total bandwidth decreases quickly alone with the process of evolution. These descending curves indicate the convergence of DCMB-GA algorithm. Figure.3 also shows that the proposed algorithm converges very fast. Meanwhile, the looser the degree-constraints, the faster it converges and the better the solution’s quality is.

In order to analyze the quality of the solutions, Waxman’s stochastic network model [9], which has the same properties of real networks, is used in our simulations. In our experiment, the cost of each link is random and distributed uniformly in [2,10]. The degree-constraint of each node is a random number between and the actual degree of the node in the graph. The number of base nodes

¹ Since space limitation, the descriptions of the algorithms are abridged.

(including the source node and the destination nodes of the multicast sessions) varies from 4 to 20 with the fixed increment 2. For each experiment network, we run DCMB-GA 100 times and record the average of total bandwidth of final optimal solutions. Comparisons are made against the heuristic algorithm DCMB-HA.

Finally, we depict these results in Figure.3(b). It shows that the quality of the solutions found by DCMB-GA is better than that of found by DCMB-HA.

5 Conclusions

By employing a continuous genetic node-weight strategy to represent the heuristic information for spanning trees, we propose a hybrid genetic algorithm to solve the degree-constrained minimal bandwidth multicast routing problem. Given the population scale pop and the maximal evolution generation $maxG$, the complexity of chromosome encoding and decoding is no more than $O(n^2)$, while the complexity of the main algorithm DCMB-GA is $O(maxG \cdot pop \cdot n^3)$. Our experiment results show that the proposed algorithm performs well.

References

1. Deering, S.E., Cheriton, D.R.: Multicast routing in datagram internetworks and extended LANs. *ACM Transactions on Computer Systems* 2 (1990) 85–110
2. Diot, C., Levine, B.N., Lyles, B., Kassem, H., Balensiefen, D.: Deployment issues for the IP multicast service and architecture. *IEEE Network* 1 (2000) 78–88
3. Saltzer, J.H., Reed, D.P., Clark, D.D.: End-to-end arguments in system design. *ACM Transactions on Computer Systems* 2 (1984) 277–288
4. Chu, Y., Sanjay, G.R., Zhang, H.: A case for end system multicast. *ACM SIGMETRICS Performance Evaluation Review* 1 (2000) 1–12
5. Yousefi'zadeh, H., Jafarkhani, H., Habibi, A.: Layered media multicast control (LMMC): Rate allocation and partitioning. *IEEE/ACM Transactions on Networking* 3 (2005) 540–553
6. Kwon, M., Fahmy, S.: Path-aware overlay multicast. *Computer Networks* 1 (2005) 23–45
7. Raidl, G.R., Julstrom, B.A.: Edge sets: An effective evolutionary coding of spanning trees. *IEEE Transactions on Evolutionary Computation* 3 (2003) 225–239
8. Feng, Y., Yu, Z., Pan, Y.: Heuristic genetic algorithm for degree-constrained multicast routing problem. *Proceedings of 2004 International Conference on Machine Learning and Cybernetics*, (2004) 2448–2452
9. Medina, A., Lakhina, A., Matta, I., Byers, J.: BRITE: an approach to universal topology generation. *Proceedings of the Ninth International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, (2001) 346–353

Using Fuzzy Possibilistic Mean and Variance in Portfolio Selection Model

Weiguo Zhang^{1,2} and Yingluo Wang¹

¹ The School of Management, Xi'an Jiaotong University, Xi'an 710049, China,
wgzhang@scut.edu.cn, zhwg61@263.net

² School of Business Administration, South China University of Technology,
Guangzhou 510641, China

Abstract. There are many non-probabilistic factors that affect the financial markets such that the returns of risky assets may be regarded as fuzzy numbers. This paper discusses the portfolio selection problem based on the possibilistic mean and variance of fuzzy numbers, which can better described an uncertain environment with vagueness and ambiguity to compare with conventional probabilistic mean-variance methodology. Markowitz's mean-variance model is simplified a linear programming when returns of assets are symmetric triangular fuzzy numbers, so the possibilistic efficient portfolios can be easily obtained by some related algorithms.

1 Introduction

The mean-variance methodology for the portfolio selection problem, posed originally by Markowitz [1], has played an important role in the development of modern portfolio selection theory. It combines probability and optimization techniques to model the behavior investment under uncertainty. The return is measured by mean, and the risky is measured by variance, of a portfolio of assets. In Markowitz's mean-variance model for portfolio selection, it is necessary to estimate the probability distribution, strictly speaking, a mean vector and a covariance matrix. It means that all mean returns, variances, covariances of risky assets can be accurately estimated by an investor. There exist a number of studies for finding efficient portfolios from solving mean-variance model, but it would be very difficult to obtain the explicit solution of efficient portfolio under general constraints such as non-negativity constraints (precluding short sales) or the lower and upper bounds constraints on investment rates (see Markowitz [1], Perold [2], Pang [3], VÖRÖS [4] and Best [5], etc..). The basic assumption for using Markowitz's mean-variance model is that the situation of asset markets in future can be correctly reflected by asset data in the past, that is, the mean and covariance of assets in future is similar to the past one. It is hard to ensure this kind of assumption for real ever-changing asset markets.

Recently, a few of authors studied the fuzzy portfolio selection problem. Watada [6] presented portfolio selection models using fuzzy decision theory. Tanaka and Guo [8] proposed the portfolio selection models based on fuzzy

probabilities and possibilistic distributions. Zhang and Nie [7] introduced the admissible efficient portfolio model under the assumption that the expected returns and risks of assets have admissible errors. Zadeh [9] proposed possibility theory based on possibilistic distributions. Carlsson and Fullér [10] defined the notions of possibilistic mean value and variance of fuzzy numbers.

In this paper, we consider the portfolio selection problem based on the possibilistic mean and variance under the assumption that the returns of assets are fuzzy numbers. The possibilistic mean value corresponds to the return, while the possibilistic variance corresponds to the risk. Especially, we obtain a linear programming model when returns of assets are symmetric triangular fuzzy numbers.

2 Possibilistic Mean and Variance

A fuzzy number A is a fuzzy set of the real line \mathcal{R} with a normal, fuzzy convex and continuous membership function of bounded support. The family of fuzzy numbers will be denoted by \mathcal{F} . Let A be a fuzzy number with γ -level set $[A]^\gamma = [a_1(\gamma), a_2(\gamma)] (\gamma > 0)$.

Carlsson and Fullér ([10]) introduced the possibilistic mean value of A as

$$\overline{M}(A) = \int_0^1 \gamma[a(\gamma) + b(\gamma)]d\gamma, \tag{1}$$

where Pos denotes possibility, i.e.,

$$Pos[A \leq a_1(\gamma)] = \Pi((-\infty, a_1(\gamma))) = \gamma, Pos[A \geq a_2(\gamma)] = \Pi([a_2(\gamma), \infty)) = \gamma.$$

Let A with $[A]^\gamma = [a_1(\gamma), a_2(\gamma)]$ and B with $[B]^\gamma = [b_1(\gamma), b_2(\gamma)] (\gamma \in [0, 1])$ be fuzzy numbers.

Carlsson and Fullér ([10]) also introduced the possibilistic variance and covariance of fuzzy numbers as

$$Var(A) = \int_0^1 \gamma([\overline{M}(A) - a_1(\gamma)]^2 + [\overline{M}(A) - a_2(\gamma)]^2)d\gamma \tag{2}$$

and

$$Cov(A, B) = \int_0^1 \gamma([\overline{M}(A) - a_1(\gamma)](\overline{M}(B) - b_1(\gamma)) + [\overline{M}(A) - a_2(\gamma)](\overline{M}(B) - b_2(\gamma)))d\gamma \tag{3}$$

The following theorem shows that the possibilistic variance of linear combinations of fuzzy numbers can easily be computed in a similar manner as in probability theory ([10]).

Theorem 2.1. *Let A and B be two fuzzy numbers and let $\lambda, \mu \in \mathcal{R}$ such that $\lambda\mu > 0$. Then*

$$Var(\lambda A + \mu B) = \lambda^2 Var(A) + \mu^2 Var(B) + 2\lambda\mu Cov(A, B),$$

where the addition of fuzzy numbers and the multiplication by a scalar of fuzzy number are defined by the sup-min extension principle [9].

3 Possibilistic Efficient Portfolio Model

In Markowitz’s mean-variance model, the return rate of risky asset is denoted as a random variable. It is well-known that the returns of risky assets are in a fuzzy uncertain economic environment and vary from time to time, the future states of returns and risks of risky assets cannot be predicted accurately. Fuzzy number is a powerful tool used to describe an uncertain environment with vagueness and ambiguity. Based on these facts, we consider the portfolio selection problem under the assumption that the returns of assets are fuzzy numbers.

In order to describe conveniently, we use the following notations:

- r_i denotes the return rate of risky asset i ,
- x_i denotes the proportion of total investment funds devoted to the asset i ,
- l_i and $u_i (u_i \geq l_i \geq 0)$ are the lower and upper bounds of x_i , respectively.

Let $r_i = (a_i, \alpha_i), i = 1, \dots, n$ be symmetric triangular fuzzy numbers with center a_i and width $\alpha_i > 0$. A γ - level set of r_i can be denoted as $[r_i]^\gamma = [a_i - (1 - \gamma)\alpha_i, a_i + (1 - \gamma)\alpha_i]$, for all $\gamma \in [0, 1], i = 1, \dots, n$.

According to (1), (2) and (3), we easily obtain

$$\overline{M}(r_i) = a_i, i = 1, \dots, n,$$

$$Var(r_i) = 2 \int_0^1 \gamma(\alpha_i - \gamma\alpha_i)^2 d\gamma = \frac{\alpha_i^2}{6}, i = 1, \dots, n,$$

$$Cov(r_i, r_j) = 2 \int_0^1 \gamma(\alpha_i - \gamma\alpha_i)(\alpha_j - \gamma\alpha_j) d\gamma = \frac{\alpha_i\alpha_j}{6}, i, j = 1, \dots, n.$$

Then the possibilistic mean value of the return associated with the portfolio (x_1, x_2, \dots, x_n) is given by

$$\overline{M}(r) = \overline{M}\left(\sum_{i=1}^n r_i x_i\right) = \sum_{i=1}^n \overline{M}(r_i) x_i = \sum_{i=1}^n a_i x_i,$$

and therefore, the possibilistic variance of the return associated with the portfolio (x_1, x_2, \dots, x_n) is given by

$$Var(r) = \left[\sum_{i=1}^n \alpha_i^2 x_i^2 + \sum_{i \neq j=1}^n \alpha_i \alpha_j x_i x_j\right] / 6 = \left(\sum_{i=1}^n \alpha_i x_i\right)^2 / 6.$$

Analogous to Markowitz’s mean-variance methodology for the portfolio selection problem, the possibilistic mean value of the return is termed measure of return and the possibilistic variance of the return is termed measure of risk.

Thus, the possibilistic mean-variance model of portfolio selection problem may be described by

$$\begin{aligned}
 \min \text{Var}(r) &= \left(\sum_{i=1}^n \alpha_i x_i\right)^2 / 6 \\
 \text{s.t.} \quad &\sum_{i=1}^n a_i x_i \geq \mu, \\
 &u_i \geq x_i \geq l_i, i = 1, 2, \dots, n, \\
 &\sum_{i=1}^n x_i \leq 1.
 \end{aligned} \tag{4}$$

Furthermore, the possibilistic mean-variance model (4) is equal to the following linear programming:

$$\begin{aligned}
 \min \sum_{i=1}^n \alpha_i x_i \\
 \text{s.t.} \quad &\sum_{i=1}^n a_i x_i \geq \mu, \\
 &u_i \geq x_i \geq l_i, i = 1, 2, \dots, n, \\
 &\sum_{i=1}^n x_i \leq 1.
 \end{aligned} \tag{5}$$

The possibilistic mean-variance model (5) is a simple linear programming, we easily obtain the possibilistic efficient portfolios by some related algorithms.

4 Numerical Example

In order to illustrate our proposed effective means and approaches of the efficient portfolios in this paper, we consider a real example of portfolio selection. In this example, we select five stocks from Shanghai Stock Exchange, their returns $r_i, i = 1, \dots, 5$ are regarded as symmetric triangular fuzzy numbers. Based on both the historical data and the future information, A γ - level set of $r_i, i = 1, \dots, 5$ are given by

$$\begin{aligned}
 [r_1]^\gamma &= (0.04 + 0.021\gamma, 0.10 - 0.021\gamma), \\
 [r_2]^\gamma &= (0.07 + 0.03\gamma, 0.16 - 0.03\gamma), \\
 [r_3]^\gamma &= (0.08 + 0.042\gamma, 0.21 - 0.042\gamma), \\
 [r_4]^\gamma &= (0.10 + 0.06\gamma, 0.28 - 0.06\gamma), \\
 [r_5]^\gamma &= (0.11 + 0.081\gamma, 0.36 - 0.081\gamma).
 \end{aligned}$$

The lower and upper bounds of the proportions of total investment funds devoted to the asset $i, i = 1, \dots, 5$ are, respectively, given by

$$l_i = 0.1, i = 1, 2, 3, 4, 5, u_1 = 0.4, u_2 = 0.4, u_3 = 0.5, u_4 = 0.6, u_5 = 0.7.$$

Thus, we obtain the possibilistic mean-variance model:

$$\begin{aligned}
 &\min 0.021x_1 + 0.03x_2 + 0.042x_3 + 0.06x_4 + 0.081x_5 \\
 &s.t. \quad 0.07x_1 + 0.115x_2 + 0.145x_3 + 0.19x_4 + 0.235x_5 \geq \mu, \\
 &\quad 0.1 \leq x_1 \leq 0.4, \\
 &\quad 0.1 \leq x_2 \leq 0.4, \\
 &\quad 0.1 \leq x_3 \leq 0.5, \\
 &\quad 0.1 \leq x_4 \leq 0.6, \\
 &\quad 0.1 \leq x_5 \leq 0.7, \\
 &\quad x_1 + x_2 + x_3 + x_4 + x_5 \leq 1.
 \end{aligned}$$

We give the possibilistic efficient portfolios as $\mu = 0.10, 0.15, 0.17, 0.19$ in Table 1.

Table 1. The possibilistic efficient portfolios

μ	0.10	0.15	0.17	0.19
x_1	0.1	0.1	0.1	0.1
x_2	0.313043	0.373333	0.106667	0.1
x_3	0.1	0.1	0.1	0.1
x_4	0.1	0.326667	0.593333	0.166667
x_5	0.1	0.1	0.1	0.533333

5 Conclusions

Fuzzy number is a powerful tool used to describe an uncertain environment with vagueness and ambiguity. In this paper, we have discussed using possibilistic mean and variance for portfolio selection decision problem which returns of assets are fuzzy numbers. We have obtained the linear programming model for portfolio selection problem replaced Markowitz’s mean-variance model when returns of assets are symmetric triangular fuzzy numbers. Using some related algorithms for solving linear programming problem easily obtain the possibilistic efficient portfolios.

Acknowledgements

This research is supported by the National Natural Science Foundation of China and China Postdoctoral Science Foundation(No.2005037241).

References

1. Markowitz, H.: Portfolio selection: efficient diversification of Investments. Wiley, New York (1959)
2. Perold, A.F.: Large-scale portfolio optimization. Management Science 30(1984) 1143–1160

3. Pang, J.S.: A new efficient algorithm for a class of portfolio selection problems. *Operational Research* 28(1980) 754–767
4. VÖRÖS, J.: Portfolio analysis-An analytic derivation of the efficient portfolio frontier. *European journal of operational research* 203(1986) 294–300
5. Best, M.J., Hlouskova, J.: The efficient frontier for bounded assets. *Math.Meth.Oper.Res.* 52(2000) 195–212
6. Watada, J.: Fuzzy portfolio selection and its applications to decision making. *Tatra Mountains Mathematical Publication* 13(1997) 219–248
7. Zhang, W.G., Nie., Z.K.: On admissible efficient portfolio selection problem. *Applied Mathematics and Computation* 159(2004) 357-371
8. Tanaka, H., Guo, P., I.Burhan Türksen.: Portfolio selection based on fuzzy probabilities and possibility distributions. *Fuzzy sets and systems* 111(2000) 387–397
9. Zadeh, L.A.: Fuzzy Sets. *Inform. and Control* 8 (1965) 338-353
10. Carlsson, C., Fullér, R.: On possibilistic mean value and variance of fuzzy numbers. *Fuzzy Sets and Systems* 122(2001) 315–326

A Novel Genetic Algorithm for Multi-criteria Minimum Spanning Tree Problem*

Lixia Han¹ and Yuping Wang²

¹ School of Science, Xidian University, Xi'an 710071, China
han_li_xia@126.com

² School of Computer Science and Technology, Xidian University, Xi'an 710071, China
ywang@xidian.edu.cn

Abstract. The multi-criteria Minimum Spanning Tree problem is an NP-hard problem, and is difficult for the traditional network optimization techniques to deal with. In this paper, a novel genetic algorithm (NGA) is developed to deal with this problem. First, based on the topology of the problem, the proposed algorithm adopts a heuristic crossover operator and a new mutation operator. Then, in order to enhance the ability of exploration of crossover, a new local search operator is designed to improve the offspring of crossover. Furthermore, the convergence of the proposed algorithm to globally optimal solution with probability one is proved. The simulation results indicate that the proposed algorithm is effective.

1 Introduction

In many real-life network applications, network design problems generally require simultaneous optimization of multiple and often conflicting objectives. The multi-criteria Minimum Spanning Tree problem (brief for mc-MST) is an important example of such problems. For single-criteria MST problem, some effective polynomial-time algorithms were developed, e.g., algorithms proposed by Prim [1], Dijkstra [2]. However, the mc-MST problem is much more difficult to deal with. Currently there have been some algorithms for this problem. For example, Corley [3] proposed an algorithm which extends the ideas of Prim's algorithm [1]. Unfortunately, the algorithm can lead us to a solution set with non-efficient trees. Hammacker and Ruhe [4] developed a non-exact procedure for searching for efficient point for the bi-criteria space. However, the algorithm can only applicable to two objective problems. Recently, Gengui Zhou [5] proposed a genetic algorithm to solve bi-criteria spanning tree problem, but its speed of convergence is slow.

As one of the Evolutionary Computation (EC) techniques, the genetic algorithms (Gas) have been receiving great attention and successfully applied in many research fields in the last decade [5~10]. In this paper, we present a framework using GAs that simultaneously optimize multiple objectives and produce a set of non-dominated solutions. Based on the topology of the problem, the proposed algorithm adopts a heuristic crossover operator and a new mutation operator. In order to enhance the

* This work is supported by National Natural Science Foundation of China (60374063).

exploration ability of crossover, a new local search operator is designed to improve the crossover offspring. Furthermore, the convergence of the proposed algorithm to globally optimal solution with probability one is proved. Finally, the simulation results demonstrate the effectiveness of the proposed algorithm.

2 The Multi-criteria MST

Given a connected undirected graph $G = (V, E, c)$ with V being the set of nodes, E being the set of edges connecting the nodes, and $c = (c_{ij}^1 \dots c_{ij}^k)$ being a vector function that assigns k weights to each edge $(i, j) \in E$, where each weight c_{ij}^r corresponds to a certain measure like distance, cost, etc. For each spanning tree can define an objective function $\sum_{(i,j) \in \text{spanning tree}} c_{ij}^r$ for each $r \in \{1, 2, \dots, k\}$. Thus we can define a k -obj-

ective minimal spanning tree problem. The mc-MST problem is a problem of optimizing simultaneously these k objectives.

The mc-MST problem is very different from single objective MST. For single objective MST, there is usually only one optimal solution, however, for mc-MST, there are usually a set of optimal solutions called Pareto Optimal Solutions.

3 The Proposed Algorithm

In this paper we proposed a genetic algorithm to solve the mc-MST problem. Keeping its network characteristic unchanged, we adopt edge set representation [9] to represent the potential solutions, i.e., a solution is encoded by a set of edges forming the spanning tree of the graph, and this encoding scheme is capable of uniquely representing all possible spanning trees.

3.1 Initialization of Population

A genetic algorithm needs an initial population of diverse genotypes. For the edge-set representation, we therefore need an algorithm that generated spanning trees. A technique based on Dijkstra's algorithm to generate random spanning tree is used.

Dijkstra's algorithm [2] greedily builds a minimum spanning tree from a start node by repeatedly appending the lowest cost edge that joins a new node to the growing tree. By using this algorithm we can only obtain one spanning tree. However, if each new edge is chosen at random, then multiple spanning trees can be obtained. This produces a scheme we call Random Dijkstra (RD).

3.2 Crossover Operator

An ideal multi-criteria optimization approach should simultaneously optimize multiple objectives. In this paper, we propose a novel crossover operator called multi-criteria crossover (MCC). In order to understand MCC easily, we define the "good

gene” as follows. If weight $c_{ij}^r (r = 1, 2, \dots, k)$ for edge (i, j) is the lowest one among all edges incident to node j (or to node i), edge (i, j) is called a “good gene” incident to node j (or to node i) with respect to the r -th weight. The main idea of MCC is that it constructs k spanning trees $s_1 \dots s_k$ from two parents. S_i can be constructed as follows: First, find the common edges of both parents. Then for each node, append the “good gene” of the common edge with respect to the i -th weight incident to that node. Third, randomly choose one edge from two parents such that resulting graph contains no cycle. If all edges from both parents are tested but S_i still can’t become a spanning tree. Then randomly take edge from edges left in E . The process is repeated until each S_j forms a spanning tree.

The details of MCC can be described in the following steps.

For $i=1$ to k do

Step1. Find the good genes (edges) of the common edge of two parents with respect to the i th weight.

Step2. Add these good genes to graph S_i .

Step3. Set $E_0 = T_1 \cup T_2$. Choose an edge $e = (u, v) \in E_0$ at random. If $s_i \cup e$ contains no circle, let $s_i = s_i \cup e$. If s_i is a spanning tree, stop; else, if $E_0 \neq \Phi$, go to step 3. else $E_0 = E \setminus \{T_1 \cup T_2\}$, go to step 3.

This operator offers the heritability of good genes and generates k offspring for each pair of parents. It avoids repair and penalty mechanisms. As a result, it is helpful to enhance the speed of exploration.

3.3 A New Local Search Scheme

Local search algorithm is an important approach for the most successful meta-heuristics to solve a wide variety of single objective combinatorial problems. It can be easily revised to be applicable to multi-criteria problems. The main difference between the single-criterion and multi-criteria cases lies in the acceptance criterion of new solution. In this paper, we adopt an acceptance criterion based on Pareto concept: A solution is accepted if it is better in at least one criterion than the current one. A new local search scheme is designed to improve the offspring generated by crossover operator. It is not an exact local optimization algorithm; instead it uses relatively small number of individuals in each search process. Thus it usually cannot generate the local optimal solution. However, it can generate the solution good enough using much less computation than general exactly local optimization search algorithm, and simultaneously improve multiple objectives. The detail is as follows.

Let S be any offspring generated by crossover operator. The offspring generated by local search scheme can be easily generated by the following pseudo code.

Choose an edge $e \in E \setminus S$ at random and insert it into S

For $r = 1$ to k do

Delete the edge with the largest weight c_{ij}^r in the circle of $S \cup e$

Obtain a new individual θ_r ,

Enddo

Delete the dominated individuals among $\{\theta_1 \cdots \theta_k\}$, and randomly select one non-dominated individuals as the result of local search from S .

3.4 Mutation Operator

Mutation is a very crucial factor in the biological evolutionary process. It usually produces spontaneously random changes in various chromosomes. Here the mutation is performed by random deletion of an edge and random addition of a feasible one.

Suppose that T is an individual chosen to take part in mutation operator. The mutation operator deletes an edge from T randomly, and inserts an edge in $E \setminus T$ randomly such that the insertion of the edge will make the resulting individual feasible.

It is clear that the mutation operator always produces feasible individual. The edges deleted and inserted are random selected. This is helpful to keep to diversity of the population and the mutation operator can search the solution space effectively.

Novel Genetic Algorithm (NGA):

Step1. (Initialization) Choose population size N , proper crossover probability p_c and mutation probability p_m , respectively. Use RD to generate initial population $P(0) \subset X$ (X is the feasible solution space). Let the generation number $t = 0$.

Step2. (Crossover) Choose the parents for crossover from $P(t)$ with probability p_c . If the number of parents chosen is odd, then randomly choose additional one from $P(t)$. Afterwards, randomly match every two parents as a pair and use the proposed crossover operator to each pair to generate offspring. All these offspring constitute a set denoted by O .

Step3. (Local Search) For each offspring generated by crossover, the proposed local search scheme is used to it to generate an improved offspring. All these improved offspring constitute a set denote by O_1 .

Step4. (Mutation) Selection the parents for mutation from set O_1 with probability p_m . For each chosen parent, the proposed mutation operator is used to it to generate a new offspring. These new offspring constitute a set denoted by O_2 .

Step5. (Selection) Select N individuals from the set $P(t) \cup O_1 \cup O_2$ as the next generation population $P(t+1)$ as follows. First, put non-dominated solutions into $P(t+1)$, then remove those solutions from set $P(t) \cup O_1 \cup O_2$, and add non-dominated solutions to $P(t+1)$ from $\{P(t) \cup O_1 \cup O_2 \setminus P(t+1)\}$. This process continues until $P(t+1)$ is fully filled. Let $t = t + 1$.

Step6. (Termination) If termination conditions hold, then stop, and keep the best solution obtained as the approximate global optimal solution of the problem; otherwise, go to step 2.

4 Global Convergence

Let X be the feasible search space of the network optimization problem.

Definition 1. For two individuals a and b if $Prob\{MC(a)=b\}>0$, then individuals b is called to be reachable from individuals a by crossover and mutation, where $MC(a)$ represents the offspring that was generated from a by crossover operator and mutation operator.

For the proposed algorithm NGA, we have the following conclusion:

Theorem 1. For any two individual $a, b \in X$, b is reachable from a by crossover and mutation operator in finite times.

Theorem 2. The population sequence $P(0), P(1), \dots, P(t), \dots$ is Pareto monotone, i.e., $P(t+1)$ is better or at least no worse than $P(t)$ for any t .

Theorem 3. The proposed genetic algorithm (NGA) converges to the Pareto optimal solutions with probability one.

5 Simulation Results

In this paper, we apply NGA to the two-criteria case, i.e., $k = 2$. Five instances are generated randomly. We execute NGA and GA1 for ten independent runs, and compare the NGA with GA1 [5] for these test cases. We use mean values of C-measure [11] to evaluate the performance of the related algorithms. The results are given in Table 1. C-measure compares pairs of non-dominated set by calculating the fraction of each set that is obtained by one algorithm and covered by the set obtained by another algorithm in a typical run, where $C(NGA, GA1)$ represents the percentage of solutions obtained by NGA which are covered by solutions obtained by GA1.

Table 1. Mean values of the C measure between NGA and GA1

Algorithm	R10	R20	R30	R40	R50
C(GA1, NGA)	59%	65%	87%	88%	94%
C(NGA, GA1)	3%	1%	1%	1%	1%

It can be seen from table 1 that at least 59% solutions obtained by GA1 are covered by solutions obtained by NGA, while only at most 3% solutions obtained by NGA are covered by solutions obtained by GA1 for all test problems. This indicates the proposed algorithm is effective.

6 Conclusions

A novel genetic algorithm for mc-MST problem is proposed in this paper. Without neglecting its topology, the proposed algorithm adopts a heuristic crossover operator and a new mutation operator. In order to enhance the ability of exploration, a new

local search scheme is designed to improve the crossover offspring. Furthermore, the global convergence of the proposed algorithm to Pareto optimal solution with probability one is proved. At last, the simulated results show the effectiveness of the proposed algorithm.

References

1. R. C. Prim: Shortest Connection Networks and Some Generalizations. *Bell system Technical Journal* 36(1957) 1389-1401
2. E. W. Dijkstra: A Note on Two Problems in Connection with Graphs. *Numerische math.* 1 (1959) 269-271
3. H. W. Corley: Efficient Spanning Trees. *Journal of Optimization Theory and Application* 45 (1985) 481-485
4. H. W. Hammacker, G. Ruhe: On Spanning Tree Problems with Multiple Objectives. *Annals of Operations Research* 52 (1994) 209-230
5. Gengui Zhou, Mitsuo Gen: Genetic Algorithm Approach on Multi-criteria Minimum Spanning Tree Problem. *European Journal of Operational Research* 114 (1999) 141-152
6. G. Zhou and M. Gen: Genetic Algorithm on Leaf-constrained Minimum Spanning Tree Problem. *Beijing Math.* 7 (1998) 50-62
7. H. Chou, G. Premkumar and C. H. Chu: Genetic Algorithms for Communications Network Design-An empirical Study of the Factors that Influence Performance, *IEEE Transaction on Evolutionary Computation.* 5 (2001) 236-249
8. J. Knowles and D. Corne: A New Evolutionary Approach to the Degree Constrained Minimum Spanning Tree Problem. *IEEE Transaction on Evolutionary Computation.* 4 (2000) 125-134
9. Günther R. Raidl and Bryant A. Julstrom: Edge Sets: An Effective Evolutionary Coding of Spanning Trees. *IEEE Trans. Evolutionary Computation.* 7 (2003) 225-239
10. Allen Arnold O: *Probability, Statistic, and Queuing Theory with Computer Science Application.* Second Edition. Boston, MA: Academic Press, Inc, 1990
11. E. Zitzler and L. Thiele: Multiobjective Evolutionary Algorithm: A Comparative Case Study and the Strength Pareto Approach. *IEEE Trans. On Evolutionary Computation.* 4 (1999) 257-271

A Software Architecture for Multi-agent Systems*

Vasu S. Alagar¹ and Mao Zheng²

¹ Department of Computer Science, Concordia University,
Montreal, Quebec H3G 1M8, Canada
`alagar@cse.concordia.ca`

² Department of Computer Science, University of Wisconsin-LaCrosse,
La Crosse, WI 54601, USA
`zheng.mao@uwlax.edu`

Abstract. Agent technology is widely used in the construction of large software systems, in particular E-Commerce and secure-critical systems. To fully utilize the potential of agents in the software system, it is essential to embed the BDI (*Beliefs, Desires, Intentions*) properties of agents in the software agents that model them. This paper introduces a formal software architectural design of a Multi-agent system (MAS) in which the BDI architecture is embedded. We embed the BDI properties of agents in an extended state machine (ESM) model and suggest that an implementation of the BDI architecture in a high-level programming language can be tested for conformance by generating test cases from the ESMs.

1 Introduction

A Multi-agent system (MAS) is an open distributed system of agents that interact among themselves according to a variety of protocols. The traditional basis for agent communication in an MAS is the theory of *speech acts*[3]. Adapting this theory modern Agent Communication Languages (ACL) have been developed. The two most commonly used ACLs are the Knowledge Query and Manipulation Language (KQML)[9] and the Foundation for Intelligent Physical Agents ACL (FIPA-ACL)[6]. The ACLs focus mainly on the structure of messages, not on the message itself. The messages are represented as performatives in KQML and FIPA-ACL. They have a formal semantics based on Belief-Desire-Intention (BDI) logics[10]. The ACLs have enabled inter-operability, an important criteria in making heterogenous agents collaborate. Flexibility, autonomy, reactivity, time-constrained behavior, and inter-operability of agents in a MAS can be handled with tools that have been developed in distributed systems and software engineering disciplines. Conversely, the full potential of well developed agent theories [11, 4] must be embedded as much as computationally feasible in the software artifacts that are developed using software engineering principles. This

* This work is supported by grants from Natural Sciences and Engineering Research Council, Canada.

paper addresses the later issue. We give a formal software architecture which embeds the BDI properties of a MAS.

Agents in a MAS are distinguished from software agents in that they are governed by BDI semantics. In order that inter-operability is consistent with BDI semantics, it becomes necessary to verify (prove) that the performatives engaged by agents conform to their mental states. To this end the logics of BDI developed by Rao [11] was adopted as the semantic basis of ACLs. However, it is shown [7] that the computational complexity of such logics is so hard that formal verification of MAS using current theorem provers seem infeasible. Recently [8, 5] several researchers are investigating special methods for model checking such systems. Model checkers work on a model of the program, not on program itself. Moreover, in a non-trivial MAS application state-explosion is common, making model checking not practical for large systems. Based on the architecture in this paper it is possible to develop [12] a conformance testing methodology of an implemented system. The paper is organized as follows: Section 2 reviews the basics of BDI logic concepts from Rao's work. In Section 3 we embed the BDI concepts in a formal theory of software agents. Section 4 gives the details of agent representation as ESMs and provides the behavioral semantics for a multi-agent system as a collection of ESMs interacting through shared events. We conclude the paper in Section 5.

2 Basic Concepts of MAS

The seminal work of Rao [10] for BDI architecture and logics has become central to research in MAS. A recent paper by Bordini, etal [5] formally proves the BDI properties in AgentSpeak(L), the agent programming language proposed by Rao. In this section we distil the essential concepts from these publications and present it in a form suitable for the discourse in this paper. We outline the properties of software components that embed these concepts.

A BDI agent is created with a set of beliefs and plans. At any instant of its life, the belief set represents the information that an agent has about its environment (its world). The set of beliefs may be structured into subsets, where each subset is the information the agent presently has about one world. An agent performs actions either because it wants to achieve its own goal or it is interacting with another agent in its environment. Desires are actions of the first kind. They are triggered by events initiated by the agent, not affecting any other agent in its environment. In our formalism we call such events as *internal events*. In an interaction, an agent performs a set of basic actions that are allowed by its set of goals and justified by its beliefs. In an interaction the set of beliefs may change. We call the events that trigger interactions as *shared* (or *external*) events. An intention is a particular course of action to which the agent has committed. An action intended at an instance is to be committed at a future instance. It is however possible that an intended action is never committed, because the expected criterion for discharging that action may never arise. This implies that the agent not believe that it will be able to do what it intends to do. However,

at no instance an agent can intend an action with the full belief that it will not do it. We conceptualize these concepts from agent theory within the framework of software engineering.

Towards embedding a BDI architecture into a software architecture, we encapsulate a BDI agent into a software component. Below we introduce software components and their properties required in such an embedding. The theory behind the embedding is sketched in the next section.

A component type $T = \langle Fr, Ar \rangle$ defines a black-box view, called frame Fr , and a grey-box view, called architecture Ar . The type T may include more than one grey-box view. The black-box defines the interface types, and the particular grey-box view Ar as a structured implemented version of Fr . It is usual to define interface of a component as an instance of either *notifies-interface* type or *receives-interface* type, and introduce four kinds of *connectors* between the interfaces of two component instances: (1). *binding* of a receives-interface to a notifies-interface between two components at the same level of nesting, (2). *delegating* from a receives-interface of a component to a receives-interface of one of its subcomponent on the adjacent level, (3). *subsuming* from a notifies-interface of a component to a notifies-interface of a component at its higher adjacent level, and (4). *exempting* an interface of a component from participating in the architectural connection. In our formalism, we call the interfaces as *ports*. The essential properties of components for MAS design are listed below:

1. The software component is designed to achieve the *goals* of a BDI agent. The goals are identified and enumerated as a list of requirements for the software component
2. The component may be autonomous, indulging in fulfilling its own *desires*. It has complete control over the set of actions it will perform. As a consequence, the component also has a *proactive* behavior. It may indulge in analytic reasoning and in performing other computational activities that are essential for advancing its goals.
3. The component responds to every stimulus from its environment. The environment of a component is a collection of components, encapsulating other agents, with whom it interacts. That is, the component is *reactive*, consistent with its intentions.
4. The component has sufficient computational power, *resources*, and *knowledge* to complete the *tasks* assigned to it from its environmental partners and deliver results within specified time bounds. Resources include its set of beliefs, and databases for transacting with other agents.
5. The component communicates with other components in its environment through messages (events) at its *ports*, where a port is an abstraction of an access point for a bidirectional communication.
6. The component can dynamically change its intended course of action whenever the context changes. Context is an aggregation of its internal state, the *role* it plays, the outstanding events (committed), and the outcome of its analytic reasoning.

Below we use the term agent to refer to the component encapsulating a BDI agent, as well as the agent according to the BDI semantics. No confusion should arise, for the intended reference will be clear in the context of its usage.

3 A Theory of Agent-Based Systems

In this section we introduce an abstract model of the agents in the software architecture and give an abstract interpretation of it. In doing so, we will be demonstrating that a software agent (component) is faithful to the BDI concepts of the agent that it encapsulates. We are not providing a formal proof here, however the proof steps are straightforward given the details of abstract interpretation.

The *global context* for a MAS is a tuple $GC = \langle \mathcal{A}, \mathcal{R}, \mathcal{P} \rangle$, where \mathcal{A} is a finite set of agent-types, \mathcal{R} is the disjoint union of the set M of messages for agent interactions, and the set $m = \{create, dispatch, engage, disengage, dispose, (silent)\}$ of *control messages*, and \mathcal{P} is a finite set of applications. An agent type is defined by a role. It has a formal definition, and has several implementations. All agents of an agent type play the same role, however they may have different implementations so that they may be used in different contexts. A port type defines a set of messages that can occur at a port of that type. The symbol @ is used to introduce port types. An agent of the agent type $A[L] \in \mathcal{A}$, where L is the list of port types, is created by instantiating each port type in L by a finite number of ports and assigning the ports to the agent. Any message defined for a port type can be received or sent through any port of that type. For instance, $A_1[p_1, p_2 : @P; q_1, q_2, q_3 : @Q]$, and $A_2[r_1 : @P; s_1, s_2 : @Q]$ are two agents of the agent type $A[@P, @Q]$. An *incarnation* of an agent A_i is a copy of A_i with a name different from the name of any other agent in the system, and with its port types renamed, if necessary. Several incarnations of the same agent can be created and distinguished by their *ids*. Letting *ids* to be positive integers, $A_1[1]$, $A_1[2]$ are two distinct incarnations of the agent A_1 . Every incarnation of an agent retains the same port interfaces. For instance $A_1[1][a_1, a_2 : @P; b_1, b_2, b_3 : @Q]$ and $A_1[2][p_1, p_2 : @P; q_1, q_2, q_3 : @Q]$ are two distinct incarnations of the agent $A_1[p_1, p_2 : @P; q_1, q_2, q_3 : @Q]$.

Based on component definitions of BDI agents we can define a software architecture of a MAS as the triple $\langle A, C, G \rangle$, where $A = \{A_1, \dots, A_n\}$ is a finite set of agents, C is the set of *connectors*, connecting the interfaces of agents in A , and G is the set of *configurations* formed by the agents in A . A connector, denoted by \leftrightarrow , is a communication link (channel) that connects the port (receive) of one agent with a port (notify) of another agent and is formally characterized by the protocols of message exchange along that connector. A configuration G defines a finite collection of agents $B \subset A$, with the set of connectors $\{\leftrightarrow | \leftrightarrow \in C\}$, established between the agents in B for communication. Thus we may write $G = \{A_i \leftrightarrow A_j \mid A_i, A_j \in B, \leftrightarrow \in C\}$. Typically a configuration is created to achieve a specific result from the collaboration of agents in that configuration. Once achieved, the configuration may either be changed to suit the next task or

may be completely abandoned in favor of creating a new one. That is, agents in an architectural configuration belong to different configurations, communicate along connectors and collaboratively achieve a task. Agents may join and leave configurations, thus dynamically changing the configuration topology.

3.1 Modes of Agents

All agents share the *modes* in the set $Modes = \{\text{initial, wait, dispatched, engaged, pause, disposed}\}$. The partial function $next_mode : Modes \times m \rightarrow Modes$ provides the new mode for a given mode and a control message. When an agent is created its ports, id, and attributes are initialized and is in mode *initial*. This mode is the result of the *create* control message. An internal (silent) transition takes the agent to the mode *wait*, where its ports, id, and attributes are initialized. The attributes include data attributes and other pre-defined functions. The databases for inclusion are beliefs, specific strategies for the role it is assigned, and its knowledge-base. The control message *dispatch* takes an agent from *wait* to the next mode *dispatched*. In this mode, it receives the information on the configuration in which it will engage. This information includes the port link connections and the protocols for interaction. When the message *engage* is received the mode of the agent changes from *dispatched* to *engaged*, where it is interacting with other agents. After completing the task it automatically returns to *pause* mode. If the system determines that its services are no longer required for that configuration it will be sent the control message *disengage* and the mode is changed to *wait*. However, if the system determines that its services are required by another configuration, it is sent the *dispatch* control message. When an agent in mode *wait* receives the message *dispose* it changes its mode to *disposed*. In all modes other than *initial* and *engaged* an agent will be executing some internal actions. In *engaged* mode, an agent may indulge in internal activity when not prompted by a shared event.

The function $which_mode? : A \rightarrow Modes$ provides the mode of an agent at any time in the system. The pair $\langle A_i, which_mode?(A_i) \rangle$ is called the *situation* of agent A_i . The *setting* of a system is a collection of situations:

$$setting \triangleq \{ \langle A_i, which_mode?(A_i) \rangle \mid A_i \in A \}$$

Two agents $A_i, A_j \in setting$ are connected if and only if $which_mode?(A_i) = which_mode?(A_j) = \text{engaged}$, and there exists a G such that $A_i \leftrightarrow A_j \in G$. A set $INI \subset \{s \mid s \in \mathbb{P}setting\}$ defines several possible initializations.

The resources in possession of an agent are formalized by *abstract data types*. The abstract data type specifications can be imported to more than one agent type. The computations that an agent performs are abstracted through the data type operations. For example, to insert a new belief b into its set of beliefs B , we write $B' = insert(b, B)$. Here, the term *insert* is defined in the *SET* abstract type definition included in the agent definition.

Once the role of the agent is defined, the events necessary to perform the role are defined. This set of events include its internal events (desires), and external

events (for communication). The fulfillment of the desires and communication tasks are governed by certain conditions, formalized as logical expressions involving the terms on the local attributes, and terms defined in the included abstract data types. In the process of executing its plans and fulfilling the desires, an agent intends certain actions. We call them constrained actions. In timed systems these actions are *time-constrained*. Time constrained actions are to be committed by the agent within a window of time. Some of these constraints are made explicit in the statement of requirements derived from the plan. Some others are imposed by the designer or the implementer of the system in order that the system exhibit *safety* and *liveness* properties.

3.2 Component Behavior and BDI

An abstract interpretation of agent behavior is shown in Figure 1. The filled arrows in the figure indicate flow of events. The dynamic behavior of the agent depends on the context, defined by the set of messages that can be received from or sent to other agents in a specific application. At any instant the agent may be busy in performing its own internal actions or it may be idle until it receives a message of interest to it. The states (Θ), transitions (Λ), and time constraints (Υ) in the figure describe the behavior of an agent in fulfilling a task in a specific context. A message from the context is received by the agent only when the internal state of the agent, and the port-condition where the message is received satisfy the specification for state change. When the specification is not satisfied in that context the message is ignored by the agent. When the incoming message is accepted it may cause its state to change, and may also require an internal computation, which updates the agent’s beliefs, knowledge, and attributes, shown by the arrow labeled with ‘Att.Func.’ Some internal action or an output message may have to be performed in future according to certain

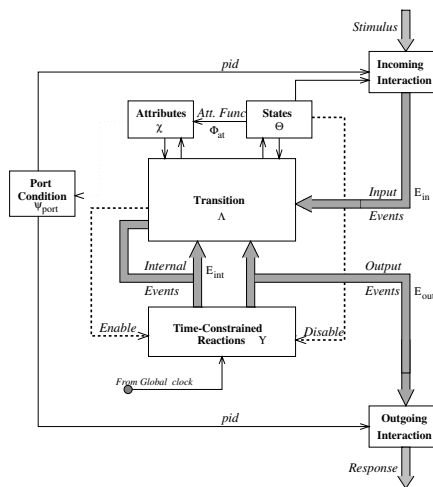


Fig. 1. Agent Modes and Agent Behavior

constraints, including time constraints. The dotted arrow connecting the block of computation to that of time-constrained reaction signifies the enabling of one or more future actions due to a computation. This constitutes the intention part of the agent. Based on an internal clock, an outstanding event is performed by the agent, thereby generating either an internal event or an output message (or control message).

Bordini et al [4] have given an interpretation of AgentSpeak(L) programs. By comparing it with the above abstract interpretation of a software component we can be convinced of the the faithfulness of the embedding of a BDI agent in the software component that encapsulates it.

4 Formal Notation for Agents

The behavior of the component (agent) $A_i \in A$ is represented as an ESM, which is an 8-tuple $(\mathcal{E}, \Theta, \mathcal{X}, \mathcal{L}, \Phi_{at}, \Lambda, \Upsilon, \circ)$ where

1. \mathcal{E} is a finite set of events and includes the silent-event tick. The set $\mathcal{E} - \{\text{tick}\}$ is partitioned into three disjoint subsets: the set \mathcal{E}_{int} is the set of events internal to the agent; the set $\mathcal{R} - \mathcal{E}_{int}$ is the disjoint union of \mathcal{E}_{in} , the set of input messages, and \mathcal{E}_{out} , the set of output messages. Each $e \in (\mathcal{E}_{in} \cup \mathcal{E}_{out})$, is associated with a unique port-type. Every input event is decorated with the symbol ‘?’ and every output message is decorated with ‘!’. Internal events are assumed to occur at the *NULL* port \circ .
2. Θ is a finite set of states. $\theta_0 \in \Theta$, is the *initial* state.
3. \mathcal{X} is a finite set of typed attributes. The attributes can be of one of the following two types: i) an abstract data type; ii) a port reference type.
4. \mathcal{L} is a finite set of abstract data types used in \mathcal{X} .
5. $\Phi_{at} : \Theta \rightarrow 2^{\mathcal{X}}$ associates with each state θ a set of attributes, possibly empty, called the *active* attribute set.
6. Λ is a finite set of *transition specifications*. A transition $\lambda \in \Lambda$, labeled by an event e is specified in (*guard, action*) paradigm as a three-tuple $\langle (\theta, \theta') ; g_\lambda ; act_\lambda \rangle$, where
 - (a) $\theta, \theta' \in \Theta$ are the source and destination states of the transition λ ;
 - (b) The guard g_λ on a transition is of the form $var_g \wedge tc$, where var_g , a conjunction of predicates on the terms of the attributes in state θ , serves as a precondition for enabling the transition and tc is a conjunction of time constraint predicates of the form $lower \leq tc \leq upper$.
 - (c) The action act_λ is a conjunction of two predicates, one on the attribute terms in the post state θ' , and the other on the clock variables that may need to be initialized. The clock initialization predicate is optional.
7. Υ is a finite set of *time-constraints*. A timing constraint $v_i \in \Upsilon$ is a tuple $(\lambda_i, e'_i, [l, u], \Theta_i)$ where,
 - (a) $\lambda_i \neq \lambda$ is a transition specification.
 - (b) $e'_i \in (\mathcal{E}_{out} \cup \mathcal{E}_{int})$ is the *constrained event*.
 - (c) $[l, u]$ defines the minimum and maximum response times.

- (d) $\Theta_i \subseteq \Theta$ is the set of states wherein the timing constraint v_i will be ignored.

The time constraints may be removed for specifying un-timed agent systems. In the definition given above we have embedded the BDI concepts as follows:

- The *triggering events* of AgentSpeak(L) are included in the set \mathcal{E} .
- The function Φ_{at} localizes the beliefs, knowledge, resources of an agent to the respective states where they will be used. If Φ_{at} does not map an attribute to a state, then no effective computation may take place in that state, except in timed systems where time is allowed to progress in such a state.
- The event e labeling the transition λ is a triggering event (in the sense of AgentSpeak(L)) if it is an input event and g_λ is true at the instant e occurs. In this case, the event e enables the action act_λ .
- If g_λ includes a time constraint (any temporal indicator) then it must have been enabled by some other event e' . That is, the action act_λ was intended at the firing of e' .

4.1 Semantics of Multi-agent System

Based on the formal ESM models for agents and parallel composition of ESMs we can give a semantics for the behavior of a MAS.

An agent responds to every message it receives. Every message is of the form $\langle e, t \rangle$, denoting that the message e occurs at time t . The *status* of an agent at any instant t is a tuple $\mathcal{OS} = (\theta; \mathbf{a}; \mathcal{R})$, where θ is the current state of the agent, \mathbf{a} is the vector showing the values for the attributes in that state, and \mathcal{R} is the vector of triggered requests not yet fulfilled. A *computational step* occurs when the agent with status $(\theta; \mathbf{a}; \mathcal{R})$, receives a message $\langle e, p_i, t \rangle$ and there exists a transition specification that can change the status of the agent. Formally, the semantics is

$$\frac{[\theta] \wedge e \in \mathcal{E}(\theta) \wedge var_\lambda(\theta) \wedge tc_\lambda}{\theta \xrightarrow{e} \theta' \wedge act_\lambda(\mathbf{a}) \wedge lower \leq t \leq upper},$$

where $[\theta]$ is the state predicate for being in state θ . An *execution* is a sequence of transitions starting from an initial state. The behaviour of the ESM is the set of executions. A computation c of an agent A_i is a sequence of alternating statuses and messages

$$\mathcal{OS}_0 \xrightarrow{\langle e_0, p_0, t_0 \rangle} \mathcal{OS}_1 \xrightarrow{\langle e_1, p_1, t_1 \rangle} \dots$$

To understand the behavior of a MAS $\langle A, C, G \rangle$ consisting of a set of agents $A = \{A_1, \dots, A_n\}$, where A_i is an instance of some agent type $\mathcal{A} \in GC$ we consider the parallel composition M of the ESMs M_1, \dots, M_n , where M_i is the ESM for agent A_i . The composition is defined by the connectors. The set of (global) states in M is a set $G \subset \Theta_1 \times \dots \times \Theta_n$, where Θ_i are the set of states of the ESM M_i . In considering the agents in GC we have included the environment as an agent, and included it in the set of agents A . Thus, the interpretation of the MAS through the

behavior of the machine M is meaningful. The states in the set Θ of the machine M constructed as a parallel composition of the ESMs M_i are all the reachable global states of the system of agents A . Each state $\theta_i \in \Theta$ is a tuple $(\theta_{1,i_1}, \dots, \theta_{n,i_n})$, where $\theta_{j,i_j} \in \Theta_j$, $1 \leq j \leq n$. The time constraints on state transitions in M are evaluated against a global clock time. The semantics for the behavior of the composed machine M is obtained by composing the semantics of each machine. Informally, an event occurring at a global state $\theta = (\theta_{1,i_1}, \dots, \theta_{n,i_n})$ is either an internal event occurring at a state θ_{k,i_k} of the machine M_k (agent state) or it is shared by two machines (two agents), say M_j and M_k . In the former case, the next state θ' for machine M is the tuple $(\theta_{1,i_1}, \dots, \theta'_{k,i_k}, \dots, \theta_{n,i_n})$, where θ'_{k,i_k} is the next state in machine M_k . In the second case, the machines M_j and M_k change their statuses simultaneously. That is,

$$\theta' = (\theta_{1,i_1}, \dots, \theta'_{j,i_j}, \dots, \theta'_{k,i_k}, \dots, \theta_{n,i_n})$$

5 Conclusion

We have proposed a software architecture for MAS. The architecture embeds the virtues of BDI concepts. In a software system, necessarily finiteness and determinism are essential. As such the set of beliefs, knowledge, and plans that can be represented and manipulated are finite, their sizes bounded by the system resources. In BDI architecture agents may be allowed an unbounded number of mental states, and mental modifications. In addition, non-determinism is permitted. In an ESM representation of an agent there is no non-determinism. The number of events that can occur in a state and the number of times an event is allowed to occur at a port within a finite interval of time are bounded. The guards are chosen such that the state transitions are deterministic. Yet, there is neither loss of generality nor rigor, because in the product machine, concurrent activity is certainly feasible. States of an ESM are abstractions of real-world situations, and they are allowed to include any abstract data type. Consequently, states include unbounded amount of information.

Agent specifications can be refined in two tiers: in one tier modes and messages can be added; in another tier, states and events may be added, timing constraints may be strengthened, data type abstractions can be refined, and new port types can be added. The two-tier refinement provides flexibility and controlled design of MAS from its initial architectural design.

Ensuring specific properties at the architecture level is of little value unless they are also ensured in the resulting implementation. As opposed to model checking which operates on the global states of the ESMs, testing is an activity conducted on the implementation. In [1, 2, 12] algorithms are discussed to generate test cases that will test an implementation for specific properties. These algorithms may be adapted to test an implementation of MAS for BDI properties in a black-box manner. The formalism for MAS discussed in this paper is the design specification in which BDI properties are embedded. Consequently test cases generated from the formal ESM model can be directed to test the fulfillment of BDI properties in the MAS implementation, usually done directly in an agent programming language. These issues need deeper study and experimentation.

References

1. Alagar, V.S., Ormandjieva, O., Zheng, M.: Incremental Testing for Self-Evolving Systems. Proceedings of Third International Conference on Quality of Software. Dallas, U.S.A., November 6-7 (2003) 12-19
2. Alagar, V.S., Ormandjieva, O., Chen, M., Zheng, M.: Automated Test Generation from Object-Oriented Specifications of Real-Time Reactive Systems. Proceedings of 10th Asia Pacific Software Engineering Conference. Chiang Mai, Thailand, December 10-12 (2003) 406-414
3. Austin, J.: How to Do Things With Words. Oxford University Press, Oxford, UK (1962)
4. Bordini, R.H., Moreira, A.F.: Proving BDI properties of agent-oriented programming languages. *Annals of Mathematics and Artificial Intelligence*, **0**: 1-30 (2004)
5. Bordini, R.H., Fisher, M., Pardavilla, C., Woolridge, M.: Model Checking AgentSpeak. Proceedings of AAMAS'03. Melbourne, Australia, July 14-18 (2003)
6. Foundations of Intelligent Physical Agents.: FIPA Specification Part 2 - Agent Communication Language. (www.fipa.org) (1999)
7. Halpern, J.Y., Vardi, M.Y.: The complexity of reasoning about knowledge and time. ACM Symposium on Theory of Computing. Baltimore, USA, ACM Press, May (1986) 304-315
8. Lomuscio, A., Lasica, T., Penczek, W. : Bounded Model Checking for Interpreted Systems: Preliminary Experimental Results. Proceedings of FAABS 2002. Lecture Notes in Artificial Intelligence, Vol. 2699. Springer-Verlag, Berlin Heidelberg New York (2003) 115-125
9. Patil, R., Fikes, R.F., Patil-Schneider, P.L., Mckay, D., Finin, T., Gruber, T., Neches, R.: The DARPA Knowledge Sharing Effort: Progress Report. In Nebel, B., Rich, C., Swartout, W (eds.): KR'92. Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference. Morgan Kaufmann, San Mateo, California (1992) 777-788
10. Rao, A.S., Georgeff, M.P.: BDI Agents: From theory to practice. In Lesser, V., Gasser, L. (eds.): Proceedings of the First International Conference on Multi-Agent Systems (ICMAS'95). AAAI Press, Menlo Park, CA, 12-14 June (1995) 312-319
11. Rao, A.S.: AgentSpeak(L): BDI agents speak out in a logical computable language. In: Van de Velde, W., Perram, J. (eds.): Seventh Workshop on Modeling Autonomous Agents in a Multi-Agent World (MAAMAW'96). Lecture Notes in Artificial Intelligence, Vol. 1145. Springer-Verlag, Berlin Heidelberg New York (1996) 42-55
12. Zheng, M.: Automated Test Generation From Formal Specification of Real-Time Reactive Systems. Ph.D. Thesis, Department of Computer Science, Concordia University, Montreal, Canada. (2002)

User-Oriented Multimedia Service Using Smart Sensor Agent Module in the Intelligent Home*

Jong-Hyuk Park¹, Jun Choi¹, Sang-Jin Lee¹, Hye-Ung Park², and Deok-Gyu Lee³

¹ Center for Information Security Technologies, Korea University,
5-Ka, Anam-Dong, Sungbuk-Gu, Seoul, Korea
{hyuks00, choijun, sangjin}@korea.ac.kr

² LG Investment & Securities Co.,Ltd.,LG I&S Bldg. 184-1, Bangei-dong,
Songpa-gu, Seoul 138-050, Korea
hypark@iflg.com

³ Division of Information Technology Engineering, SoonChunHyang University,
Eupnae-ri, Shinchang-myun, Asan-si, ChoongNam, Korea
hbrhcdbr@sch.ac.kr

Abstract. As the interest about Ubiquitous Computing has been increasing, it is actively processing research which advanced countries try to realize it such as Smart Space, Cool Town, Easy living, TRON project, and so on. The aim of these projects provides user oriented intelligent service considering relationship among main components (user, object, and environment) of a ubiquitous era. In this paper, we propose User-oriented intelligent Multimedia Service system in the Intelligent Home (IHUMS). The proposed system conducts intelligently the context information (user preference, user location, device status, etc.) using smart sensor agent module. It also provides the interoperability of multimedia among incompatible devices, authentication method which is suitable for the Intelligent Home, and transparent and secure service.

1 Introduction

Rapid growth of computing power has finally caused the advent of the era of ubiquitous, where thousands of devices are connected to network from tools to environments, and telecommunication paradigms are changed from people-to-people to people-to-machine and objects to objects. Advocated by Mark Weiser of Xerox PARC in 1998, ubiquitous computing has provided the users with valuable services by interaction among numerous computers without user's recognition, through unlimited connections regardless whenever and wherever, where all the objects and spaces in daily life come to be intelligent. Furthermore, following the establishment of the intelligent Home environment at home, a variety of affluent and convenient services of in-house systems such as security, health-care, U-commerce will be available soon [1-3].

As the interest about Ubiquitous Computing has been increasing, it is actively processing research which an advanced countries tries to realize it such as Smart

* This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

Space [10], Cool Town [11], Easy living [12], TRON [13] project, and so on. The aim of these projects provides user oriented intelligent service considering relationship among main components (user, object, and environment) of a ubiquitous era.

The rest of this paper is organized as follows: In Section 2, we discuss Related Works about the IH. In Section 3, we describe architecture, design, operation process which included individualized protocol, and analyze the proposed system. We conclude in Section 4.

2 Related Works

WSN and CAC are the core technology which realizes the IH. Wireless Sensor Network (WSN) consists of sensor nodes which are portable computers with built-in micro-controllers. Moreover, it consists of sensing, data processing, and communication modules. WSN is formed by an inside of a particular incident/situation or a number of sensor-nodes in its surroundings and the sensor network algorithm and self protocol have self-organizing capabilities [16].

Context Awareness Computing (CAC) is the technology allowing communication between user and computer. Its aim is to bring the standard of understanding of communication up to such level of human world. CA recognizes the situation (a location, a place, sound levels, duties, private situations and time) of the user being faced and acquires correctly the information in accordance with the situation as a desired form. The system which provides above resources is considered as a context awareness system. The situations mean situation information for at least one object. It is a person, a place, time and subjects that could be the object of situations and they are suitable factor between users or applications. The computing resources equipped the ability of context awareness are required the function that obtains and extracts a situation data, converts the data into the form suited for present situation [6 - 8].

To offer multimedia service in the IH, the following requirements should be considered: user-friendliness and transparency of use, multimedia interoperability among incompatible devices, authentication, confidentiality, integrity, availability, and mobility [4, 5, 9].

3 IHUMS System

3.1 System Architecture

The proposed IHUMS system consists of two parts. One part involves the process of transmitting multimedia contents to the IH after it has been created. The other part is the course which the user consumes the contents in the IH.

The first part follows multimedia contents distribution model as seen in [17]. A front part of IMSS consists of CA, DOI-RA, CP, MD, CS, PG etc. Certificate Authority (CA) issues a certificate to authenticate that user are authorized to creators distributors, etc. The Digital Object Identifier Registration Authority (DOI-RA) provides a function that generates identifier in order to manage the multimedia in efficient and systematic way. Contents Provider (CP) is a copyright holder who provides multimedia, creating safe contents by downloading packager. Multimedia Distributor (MD)

manages safe multimedia files which are distributed by copyright holder, and provides interface that users can consume. It consists of CMS (Contents Management System) and LMS (License Management System) [18].

In this paper, we will discuss in detail the multimedia service in the IH into details which is the second part (refer to Fig.1).

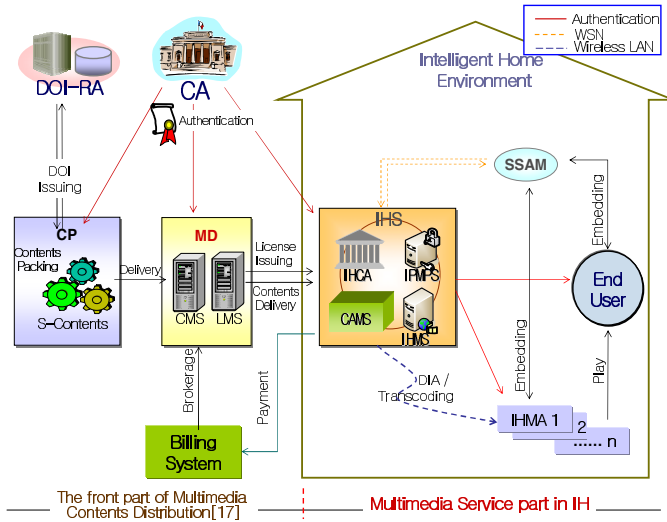


Fig. 1. IHUMS System Model

3.2 System Design

Our new system consists of four main bodies which are IHS, IHMA, SSAM, and End User.

- **IHS (Intelligent Home Server):** It is a core element in system construction of the IH and consists of IHCA, IPMPS, CAMS, and IHMS. The IHCA (IH Certificate Authority) is responsible for user and IHMA (IH Multimedia Appliance) authentication. It is authenticated by IHCA acting as a private CA authenticated by a public CA. The IPMPS (Intellectual Property Management & Protection System) is responsible for the security function of multimedia contents downloaded to IHMS (IH Multimedia Server) through external line. It provides the minimum security details applicable to the IH such as user right setting, authorization, management, etc. in order to ensure the transparency of users. The CAMS (Context Aware Management System) stores and controls context information collected by SSAM (Smart Sensor Agent Module) through wireless sensor network into database. Also, it determines a multimedia service by receiving current user status.
- **SSAM:** It has a function that collects context information on each individual by connecting to End User and IHMA, and controls the network communication among SSAMs. Its components are as follows : a Low Power Manager, sensors (motion, voice, brightness, and device status check sensor), operation system,

module for wireless packet transmission and context information processing, middleware, and application, etc.

- **IHMA:** It is a device that intelligently displays multimedia contents in the IH. It has a complemented multimedia service such as PDA, PC, DTV, etc., transmits the context information of multimedia devices to IHS by SSAM.
- **End User:** He/She is an authorized user authenticated by biological recognition and uses the multimedia contents in the IH. They always carry a device (mobile phone, necklace, watch, etc.) in which SSAM is embedded, and send user context information in real-time to IHS through the device.

3.3 System Operation Process

The operation process of the proposed system consisted of 3 sectionalized protocols, and the notations in table 1 are used throughout this paper.

Table 1. Notations appeared in this paper

Notations	Meanings
E, E'	Distinctive symmetric key cipher algorithm
$r_1, r_2, r_3, r_4 / PK$	Random Number / Public key of IHS
k	Symmetric key is used by User and IHS in each session
KEY	Symmetrical key predefined between IHS and IHMA
$X_ID / H()$	Unique Identifier of X / Hash function
X_Auth / X_Cert	Authentication / Certificate of X
BIO_Info	Recognition Information of the user's biometric information such as a fingerprint, iris, face etc.
$X_Time.Date$	Time or Date information of the X
Req_X / Res_X	Request / Response of the X
Usr_Stat_Info	User context information (user position, contents starting / Termination point, additional information on contents, etc.)
$Updating_Cont_Adv$	Advertising of newly added contents information
$IHMA_Des_Info$	Information on description of IHMA
$Adp_Multi_Service$	Adapted multimedia service
$Down, Stream, or Play(Cont)$	Download, Streaming, or Play Contents
X_Autho	Authorization of using right about the contents X

3.3.1 End User and Sectional Authentication Protocol of IHMA and IHS

In this sub-section, we describe an authentication process between IHCA of IHS and End User / IHMA. Each step is organized as follows.

STEP 1. User authenticates IHS by Certificate of the IHS.

$End\ User \rightarrow IHS : Req_IHS_Auth,$

$End\ User \leftarrow IHS : Res_IHS_Cert[H(UserID||DeviceID)],$

$End\ User \rightarrow IHS : IHS_Auth_Success_Message$

STEP 2. A process of joint ownership after the generation of a session key k between User and IHS in order to use the symmetric key encryption in the stages of 3 of user authentication process.

$End\ User \rightarrow IHS : E_{PK}(k, r_1, User, IHS),$

$End\ User \leftarrow IHS : E_K(r_{1-1}, r_2), End\ User \rightarrow IHS : E_K(r_2-1)$

STEP 3. User authentication process.

End User \rightarrow *IHS* : *Req_User_Auth*,

End User \leftarrow *IHS* : $E_k(\text{BIO Info} \parallel \text{CurrentTime.Date})$,

End User \rightarrow *IHS* : $E_k(\text{User_Auth.Success} \parallel \text{User_Auth.Time.Data})$

STEP 4. A process of mutual authentication using predefined symmetric key between IHS and IHMA where two encoded algorithms are being used to ensure mutual security and confidentiality.

IHS \rightarrow *IHMA* : *IHMA-IHS_Auth*, r_3 , *IHS* \leftarrow *IHMA* : $E_{KEY}(\text{IHS}, r_3, r_4)$, r_4 ,

IHS \rightarrow *IHMA* : $E'_{KEY}(r_3, r_4)$

3.3.2 The Protocol Between IHS and End User

In this sub-section, we describe the protocol between IHS and End User. This protocol is a process until the End User downloads the preferred contents through IHS and uses them after the user is authenticated. Each step is organized as follows.

STEP 1. To use the User and IHMA authentication method described in 3.3.1, Authentication is accomplished.

End User \leftrightarrow *IHS* : *Req_User-IHS_Auth*

STEP 2. Authenticated user requests new multimedia contents in need through IHS. The contents are downloaded to IHMS through external line.

End User \rightarrow *IHS* : $E_k(\text{Req_New_Cont})$, *IHS*: *Down(Cont)*

STEP 3. The IPMPS of IHS controls the copyright of the contents by the management of user rights such as setting, authorization, management etc. and the contents are allowed to be played when conditions are satisfied.

End User \leftarrow *IHS* : *Cont_Autho*, *End User*: *Play(Cont)*

STEP 4. In addition to playing contents, the context information such as user location, the status of IHMA, information of multimedia contents, contents start and termination point, etc. is being transmitted to CAMS of IHS for user-oriented service in future.

End User \rightarrow *IHS* : $E_k(\text{User_Stat_Info})$

3.3.3 The Protocol Between IHS and SSAM || IHMA

In this sub-section, we describe the protocol between IHS, SSAM and IHMA. Each step is organized as follows.

STEP 1. IHS advertises updating information on new contents downloaded upon user's request through SSAM.

IHS \rightarrow *IHMA* : $E'_k(\text{Updaing_Cont_Adv})$

STEP 2. CAMS determines the service by taking users into account after receiving information (user context, IHMA description, etc.) on whereabouts of user and description of the nearest IHMA.

IHS \leftarrow *IHMA* : $E'_k(\text{IHMA_Des_Info})$, *IHS* : *Descision_Service*

STEP 3. CAMS commands an adapted multimedia service by applying DIA and video trans-coding method with IHMS, and IHMS plays the contents by producing a high-resolution media service with IHMA.

IHS : *Command execution Adp_Multi_Service*,

IHS \rightarrow *IHMA* : *Streaming(Adp_Multi_Service)*, *IHMA*: *Play(Cont)*

3.4 Analysis of the Proposed System

In this section, we compare the existing system- multimedia service at S / K-Com.[19, 20] demonstration enterprise consortium in Korea - with the proposed IHUMS system. We also discuss security issues of the proposed system.

- **User-oriented intelligent multimedia service:** In the existing system, user selects preferred contents and commands passively by selecting the device to play. However, the proposed system provides a user-oriented multimedia service suitable for the IH. The proposed system can execute automatically information on user preference, status, etc. without the passive intervention of the user by the SSAM.
- **User and device authentication method suitable for the IH:** The existing system uses an officially approved certificate method using PKI in user authentication, and the device authentication uses a separated service provider as CA. However, this method contains problem with a decrease in speed and additionally needed authentication when new device is being added. Whereas, in the proposed system, user and device authentication are assigned to IHS acting as a subordinated private authentication body authenticated by precedent authentication body upon primary setting, and derive efficiency in speed by using predefined symmetrical key method instead of public key upon device authentication.
- **Multimedia interoperability among multimedia devices:** The existing system offers a multimedia service separately to two different devices, and in order to offer that service, formatted contents applicable to each device are needed to be made beforehand. Whereas, proposed system offers a multimedia interoperability among incompatible devices with a uni-formatted contents by adapted digital item and real-time video transcoding.
- **Efficiency of user rights management of multimedia contents:** The existing system is operated by XrML based on license system for the safety of general online contents in user rights management. This system has an inconvenience of user to reconnect to the main license server through external line in case of switching of user rights between user A and user B. Whereas, in proposed system, the convenience of user and the efficiency of user rights management can be increased by using a license system capable of switching user rights through IPMPS of IHS.
- **Mobility:** For its own device, user can receive authentication and service through a certificate. Accordingly mobility can be guaranteed with $H=(UserID||DeviceID)$ issued to user. Even if user's $H=(UserID||DeviceID)$ is intercepted in the middle, safe mobility can be guaranteed after using the shared symmetric key.
- **Confidentiality:** User's information can be safely transmitted as communication between user and IHS is performed using session key k created by user. After the user's authentication to be completed, user transmits IHS message such as $E_{PK}(k,r_i,User,IHS)$, including information about session key k . *Connection/Non-connection Confidentiality* - Communication can be realized safely in connection status using the symmetric key exchanged between user and IHS. In case of non-connection status, although any other device attacks it disguised as a current device, the symmetric key between User and IHS might not be revealed. Accordingly confidentiality can be maintained in non-connection status.

We can consider the following attacks when offering multimedia service in the IH. We discuss security safety about the attack in our new system.

- **Reflection Attack:** In the process of a mutual personal-identification in *STEP 4* of sub-section 3.3.1, an attacker disguised as IHS starts second session after *the last stage* in *STEP 4* transmitting r_4 to IHMA, then IHMA transmits $E_{KEY}(r_4)$ with additional random number in a platform where two and more of sessions can be operated simultaneously. In these stages, IHMA recognizes IHS as an authorized user by transmitting acquired $E_{KEY}(r_4)$ without knowing the *KEY* in order to proceed *the last stage* after forced termination of the second session by the attacker. This proposed system ensures the security by using two symmetric key algorithm as mentioned in 3.3.1 to resolve the reflection attack[15] from the using of identical encryption system.
- **Forged and counterfeited License file Attack:** It is important for the license file to be properly authorized to use contents. The security of the system would be vulnerable if attacker forges and counterfeits the license file by acquiring it. This system prevents the forge and counterfeit of license file by using XML signature.
- **IP Spoofing Attack:** W-LAN (Wireless Local Area Network) is used in home environment to command mutually among devices. The high-frequency waves allow access to the attacker from outside to bug the system as it can be travel through wall always causing a great danger in security in home environment. As seen in 3.3 protocol, this system prevents the spoofing attack by means of applying packet encoding, using a symmetrical key algorithm on important information packet.
- **Denial of Service (DoS) Attack:** Attacker can send a large amount of useless traffic or transmit radio wave to interfere WLAN connection on IHS and IHMA. This system can limit the damage caused by the attacks by authentication of user and device on IHS, and assigning resources on sole IHMA network.

4 Conclusion

In this paper, Our new system offers a user-oriented intelligent multimedia service that is an advanced form of the general multimedia service in the existing home network by context awareness using SSAM in conjunction with the WSN technology. Also, by resolving the interoperability issue among incompatible multimedia devices in home network, it adaptively controls and identifies the resolution among different devices, and resolves the issue by using optimized video trans-coding on each devices. Finally, we propose and influence the way of improvement on security and efficiency of authentication of user and home device, and increases the efficiency of user rights management of multimedia in the IH.

References

1. Mark Weiser: The Computer for the 21st Century, Scientific American (1991), 94-104
2. Mark Weiser: Hot topic: Ubiquitous Computing IEEE Computer (1993), 71-72

3. Max Muhlhauser: Ubiquitous Computing and Its Influence on MSE, Proceedings of International Symposium on Multimedia Software Engineering (2000), 48-55
4. Enyi Chen, Degan Zhang, Yuanchun Shi, Guangyou Xu: Seamless Mobile Service for Pervasive Multimedia. PCM (2004), 754-761
5. Frank Stajano: Security for Ubiquitous Computing, Wiley (2002)
6. Cliff Randell, Henk Muller: Context Awareness by Analyzing Accelerometer Data, <http://www.cs.bris.ac.uk/Tools/Reports> (2000)
7. Philip Robinson: Context-Awareness in trust management for Business Applications, I-trust workshop (2003)
8. A.K. Dey and G.D. Abowd: Towards an understanding of context and context-awareness, HUC99.
9. Zhexuan Song, Ryusuke Masuoka, Jonathan Agre, and Yannis Labrou: Task Computing for Ubiquitous Multimedia Services, MUM (2004), 27-29
10. Smart Space Project, NIST, <http://www.nist.gov/smartspace/>
11. Cool Town Project, Hewlett-Packard, <http://www.cooltown.hp.com>.
12. Easy living Project, MicroSoft, <http://research.microsoft.com/easyliving/>
13. TRON Project, The University of Tokyo <http://www.ertl.jp/ITRON/home-e.html>
14. Henning Schulzrinne, Xiaotao Wu, Stylianos Sidiroglou: Ubiquitous Computing in Home Networks, IEEE Communications Magazine (2003), 128-135
15. A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone: Handbook of Applied Cryptography, CRC Press (1996), 503 / 530
16. M. Satya: IEEE Pervasive Computing Magazine, <http://www.computer.org/pervasive>.
17. Jong-Hyuk Park, Sung-Soo Kim, Jae-Won Han, Sang-Jin Lee.: Implementation of the H-IPMP System Based on MPEG-21 for secure Multimedia Distribution Environment, WSEAS TRANSACTIONS on INFORMATION SCIENCE and APPLICATIONS, Issue 5, Volume 1 (2004), 1301-1308
18. Jong Hyuk Park, Heung-Soo Park, Sangjin Lee, Jun Choi, Deok-Gyu Lee: Intelligent Multimedia Service System Based on Context Awareness in Smart Home. KES 2005, 1146-1152
19. SKT Website, <http://www.sktelecom.com>
20. KT Website, <http://www.kt.co.kr>

Meta-learning Experiences with the MINDFUL System

Ciro Castiello and Anna Maria Fanelli

CILab - Computational Intelligence Laboratory,
Dipartimento di Informatica, Università degli Studi di Bari,
Via E. Orabona, 4 - 70126 Bari, Italy
{castiello, fanelli}@di.uniba.it

Abstract. In this paper, we present an original meta-learning framework, namely the MINDFUL (Meta INductive neuro-FUzzy Learning) system. MINDFUL is based on a neuro-fuzzy learning strategy providing for the inductive processes applicable both to ordinary base-level tasks and to more general cross-task applications. The results of an ensemble of experimental sessions are detailed, proving the appropriateness of the system in managing meta-level contexts of learning.

1 Introduction

The main research efforts in the field of machine learning have been often paid in the development of the so-called *base-learning* strategies, where data-based models exhibit generalisation capabilities when tackling a particular task. However, base-learning approaches are subjected to a number of limitations, which can be theoretically stated by the no free lunch theorems [1]. They assess that, when averaged over all possible tasks, the generalisation performance of any two learning algorithms is practically identical. In other words, there is no way to find a learning method which can outperform all others when we consider all the possible universe of tasks. This kind of conclusions are related to the fundamental role played by the learning bias, that embeds the set of assumptions that altogether instruct a learner and influence the hypothesis selection. The common base-learning strategies make use of a fixed form of bias and therefore a learner is restricted to a limited area of expertise.

Meta-learning consists in extending the traditional base-learning mechanisms to perform a dynamical search of a suitable bias for a particular task, possibly exploiting past accumulated learning experience [2, 3]. The design of a meta-learning system relies on the definition of proper mechanisms of dynamic bias discovering. The main question is related to the possibility of acquiring and exploiting a kind of high-level knowledge (meta-knowledge), useful for improving the capabilities of a given learning system [4]. A number of meta-learning approaches can be distinguished in the literature panorama; most of them are characterised in terms of the involved learning models employed to manage cross-task applications. Actually, several strategies of model combination and model

selection have been proposed, involving different learners which constitute a pool of bias-specific candidates [5, 6]. On the other hand, the working scheme of a single learning model could be organised to continuously adapt to new situations. In this case, meta-knowledge is used to modify the learner behaviour or architecture, which ultimately determines the selection of suitable biases [7].

The research activity described in this paper takes part in the investigation of the latter category of meta-learning strategies, presenting a particular framework based on neuro-fuzzy integration, namely the MINDFUL (Meta-INDuctive neuro-FUZZY Learning) system. MINDFUL is centred on the employment of a single inductive learning model, to bring forward a kind of parameter selection approach. In practice, a neuro-fuzzy scheme plays the twofold role of base-learner (to tackle ordinary predictive learning tasks) and meta-learner (to produce a form of meta-knowledge to be used in future base-learning situations). The meta-learner provides an explicit meta-knowledge, in terms of fuzzy rules, representing a significant form of bias to direct the learning process of the base-learner. In this way, the idea of meta-learning is translated to a more qualified level: instead of simply arbitrating or selecting a learning strategy among a pool of candidates, the focus is shifted to the analysis of a particular model, in order to improve it. A preliminary experimental session, described in the paper, proves the appropriateness of the MINDFUL system in retaining the knowledge accumulated during learning, showing improved performances when tackling new tasks.

The paper is organised as follows. In the next section we introduce the MINDFUL system, presenting the underlying hybrid learning framework. The results of an ensemble of experimental sessions are presented in section 3 to show the effectiveness of the proposed meta-learning methodology. Finally, some conclusive remarks are drawn in section 4, together with guidelines for future work.

2 Description of the MINDFUL System

In this section, we are going to trace an outline of the MINDFUL system working scheme, highlighting the neuro-fuzzy model underlying the overall learning strategy and the particular nature of the involved meta-knowledge.

2.1 The KERNEL of the MINDFUL System

The core of the MINDFUL system is represented by a neuro-fuzzy model. By its own name, the KERNEL of the system suggests the working engine of the adopted strategy: Knowledge Extraction and Refinement by NEURO-fuzzy Learning. The KERNEL provides for the inductive process, applicable both to the ordinary base-learning level and to a more general cross-task level. Starting from the analysis of observational data, the neuro-fuzzy model produces a fuzzy rule base which codifies the processed information in a linguistically comprehensible fashion. Each fuzzy rule is expressed in the form:

$$\text{IF } x_1 \text{ is } A_1^r \text{ AND } \dots \text{ AND } x_m \text{ is } A_m^r \text{ THEN } y_1 \text{ is } b_1^r \text{ AND } \dots \text{ AND } y_n \text{ is } b_n^r, \quad (1)$$

where the index $r = 1, \dots, R$ indicates the r -th rule among the R comprised into the rule base; A_i^r are fuzzy sets (defined over the input components x_i , $i = 1, \dots, m$); b_j^r are fuzzy singletons (defined over the output components y_j , $j = 1, \dots, n$). Gaussian membership functions are employed to describe the fuzzy sets A_i^r , and the fuzzy inference system evaluates the fulfilment degree of each rule using the product operator as the particular T-norm interpreting the AND connective. The inferred output is evaluated as the average of the R activation strengths weighted by the fuzzy singletons. The connectionist component of the KERNEL is represented by a three-layers feed-forward neural network, (the *neuro-fuzzy network*), reflecting in its topology the structure of the fuzzy inference system. The free parameters of the neuro-fuzzy network are therefore associated with the parameters of the Gaussian functions and fuzzy singletons. The learning scheme of the network is articulated in two successive steps, intended to firstly initialise a knowledge structure and then to refine the obtained fuzzy rule base:

Extraction of initial knowledge in form of fuzzy rules. An unsupervised learning of the neuro-fuzzy network is performed in this step, to cluster the input data and initialise the fuzzy rule base. A modified winner-takes-all scheme applies a rival penalising mechanism that rewards the winner unit and penalises the rival one (second winner). Such algorithm is able to automatically adapt the network structure, since it identifies a proper number of clusters (starting from a guessed number) for the problem at hand, coinciding with the number of rule nodes. Two distinct learning rates are involved in this learning process, for the winner and the rival units, respectively.

Refinement of fuzzy rules in terms of accuracy. The base of knowledge is refined employing a supervised learning process of the neuro-fuzzy network (based on a gradient descent technique), in order to attune the fuzzy rule base to the numerical data. In this step, the learning rate value is one of the main parameters to be set for a proper learning session.

We do not provide the mathematical details concerning the formalisation of the learning algorithms (whose discussion does not concern the scope of this article), addressing the reader to some other previous works of ours [8, 9]. In the following, we underline how the meta-learning practices performed by the MINDFUL system differ from the common base-learning activities in terms of the involved meta-knowledge.

2.2 The Involved Meta-knowledge

The KERNEL of the MINDFUL system is identically employed both at base- and meta-level of activity. Actually, the fuzzy rule bases, derived by examining different base-level tasks, express the knowledge connected with the solution of singular problems, but a kind of cross-task knowledge could be identified during the repeated application of the system to different categories of tasks. This *meta-knowledge* correlates “higher-level” pieces of information, concerning the

characterisation of the tasks under analysis and the learner behaviour with distinct bias configurations. On the one hand, an analysis of the learning tasks is engaged to derive a set of numerical meta-features that are relevant for qualifying each base-level dataset. On the other hand, a process of performance assessment is carried on to evaluate the different hypotheses that can be originated during the base-learning activity employing distinct bias configurations.

The characterisation of each task \mathbb{T} has to be determined by examining its corresponding dataset T : a collection of morphological features $\mathbf{z} = (z_1, \dots, z_p) \in \mathbb{R}^p$ can be extracted from T for this purpose. Obviously, we assume that the vector \mathbf{z} , besides banally providing numerical information concerning the data, might furnish a somewhat precise knowledge of the particular task underlying the dataset at hand. (Actually, this kind of assumption is widespread in meta-learning contexts, where the employment of numerical meta-features represents a common practice [5, 10].) The components of the vector \mathbf{z} include a number of meta-features which can be classified as general, statistical and information-based, according to their informational contents. These kinds of meta-features have been considered in literature to generically characterise a learning task, however our choice for the measures composing the vector \mathbf{z} derives from a deeper analysis which allowed a suitable selection among the commonly employed meta-features and introduced novel mechanisms of evaluation [9, 11].

The process of performance assessment aims at identifying the best bias configuration, to be applied to the neuro-fuzzy learning model, in correspondence of each base-level task. To this aim, an analysis of the working scheme underlying the KERNEL allows to identify the most discriminating factors in determining the inductive leap: they constitute the components of a vector \mathbf{b} , representing the bias configuration. Particularly, $\mathbf{b} = (\alpha_\omega, \alpha_\rho, \eta, \tilde{R})$, where α_ω and α_ρ represent the learning rates values (related to the winner and rival units) involved in the unsupervised learning algorithm; η is the learning rate value adopted in the supervised learning process and \tilde{R} represents the guessed number of clusters to be chosen to initiate the clustering procedure. These parameters affect the training and the topology of the neuro-fuzzy network and therefore stand as the bias of the overall learning strategy. By employing distinct bias configurations, different hypotheses (namely, fuzzy rule bases) can be derived for each learning task. During the base-learning activity, the performance assessment process records the bias configuration yielding the best hypothesis, evaluated both in terms of accuracy (performance results) and comprehensibility (number of rules in the obtained base of knowledge).

While a generic training set $T = \{(\mathbf{x}_k, \mathbf{y}_k)\}_{k=1}^K$ collects observational data related to particular base-level tasks, the information deriving from the meta-feature extraction and performance assessment processes is used to build up a meta-training set $T_{meta} = \{(\mathbf{z}_k, \mathbf{b}_k)\}_{k=1}^{K'}$. The meta-training set is obtained at the end of a base-learning session of experiments (involving different learning tasks) and represents the starting point for the meta-learning activity. The MINDFUL system replies over T_{meta} the neuro-fuzzy learning strategy performed by its KERNEL, and the obtained fuzzy rule base codifies the meta-knowledge

correlating meta-features with configurations of bias. This meta-knowledge could prove its usefulness in future base-level applications, suggesting the proper bias values to be employed without the need of further experimentations.

3 Experimental Session

In this section, we present a preliminary experimental session, performed over a number of synthetic domains of tasks, to evaluate the competence of the MINDFUL system in retaining and profiting of the accumulated past experience. The overall experimental activity is organised in two phases: the system is firstly applied to solve base-level problems, in order to compile a meta-training set; successively the hybrid learning procedure is employed to derive an appropriate meta-knowledge, useful to improve the system efficiency for successive base-learning applications.

Each synthetic domain identifies a number of classification tasks, sharing common notations. Some of them have been tackled by the MINDFUL system, with different bias configurations for each task, in order to seek the best learning performance attainable. The task environments selected for our experimental purposes refer to a number of particular synthetic domains, specially designed to gather a set of distinct problems, sharing similar characteristics. Particularly, we identified domains referring to different concept learning problems (classification tasks with two possible outputs). In each domain six tasks are involved, labelled as T_0, \dots, T_5 . For each task inside the synthetic domains, 1000 sample points have been randomly generated (by means of a uniform distribution of probability) and properly labelled with their target values to compose the associated datasets.

The information concerning the accumulated base-learning experience can be compiled in form of numerical data into a meta-training set, to be employed for accomplishing the successive meta-learning phase. Once the meta-knowledge has been properly derived, it can be used in novel learning problems to infer suitable parameter settings, whose overall appropriateness can be evaluated.

In the following, all the stages of the simulation (conducted into the `Matlab`[©] 6.5 environment) will be briefly detailed.

3.1 Base-Learning Phase

In the preliminary base-learning experiments the system analysed different base-level tasks, drawn from the synthetic domains previously described. A number of five bias configurations ($Bias_1, \dots, Bias_5$) have been adopted during the separate applications of the KERNEL to the specific tasks. The bias component values have been selected by committing to the past experience of usage of the neuro-fuzzy strategy, in different contexts of (base-level) application [8, 9]. The five bias configurations have been in turn applied to investigate each base-level task, and a 10-fold cross-validation procedure has been executed to identify the configuration providing the best performance results for each problem (a total number of 400 experiments have been carried out).

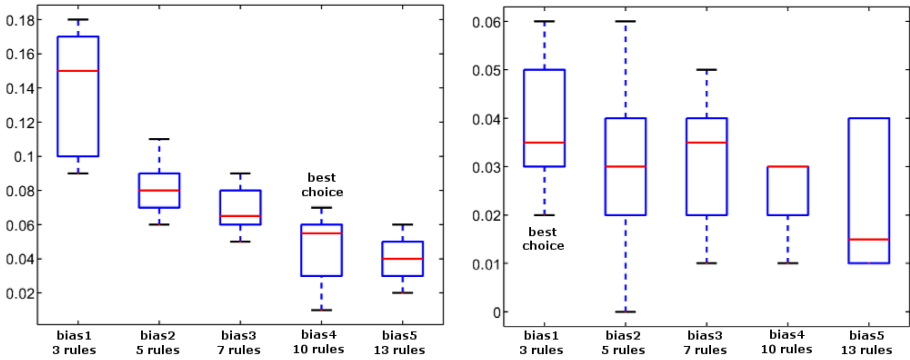


Fig. 1. Box and whiskers plots describing the variation of the classification errors, evaluated on two different datasets according to the selected bias configuration

As an example, figures 1 summarises the results of the 10-fold cross validation sessions performed on a couple of datasets. The box and whiskers plots report the variation of the classification error for each of the five bias configurations. It can be easily argued that $Bias_4$ and $Bias_1$ represent the best parameter configuration (in terms of the trade-off between accuracy and comprehensibility) for tackling the tasks. Due to limited space, we cannot report all the results obtained during the analysis of each task. However, it is important to underline how the empirical evidence showed that datasets, describing tasks in the same domain, have been best learned by the employment of the same bias configuration. To a certain extent, this represents a first confirmation that different categories of problems should be tackled with the adoption of different biases.

At the end of the base-learning phase, the meta-training set T_{meta} can be assembled by exploiting the extracted meta-features and the process of performance assessment. The obtained meta-training set represents the starting point for organising the high-order knowledge and for initiating the meta-learning process.

3.2 Meta-learning Phase and Meta-knowledge Evaluation

Once the meta-training set has been properly defined, the MINDFUL system is ready to enter the meta-level phase of the learning strategy. In this case the neuro-fuzzy strategy is applied to solve a particular kind of regression problem. Indeed, the output values to be predicted by MINDFUL correspond to the numerical bias components. In order to derive usable meta-knowledge, expressed in the form of a fuzzy rule base, the hybrid learning process has been performed by differently employing the same five bias configurations previously adopted. The fuzzy rule base embedding the meta-knowledge derived at the end of the meta-learning experiments is composed by 4 fuzzy rules.

To evaluate the benefits deriving from the employment of the meta-learning strategy, we verified the bias indications provided by the meta-knowledge by investigating new base-level tasks. To this aim, we considered some datasets related to tasks that did not contribute in determining the meta-training set.

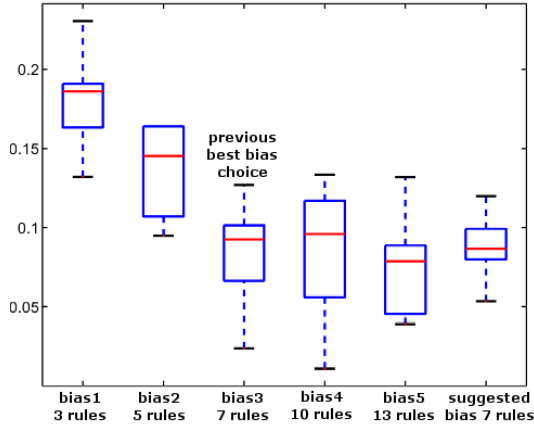


Fig. 2. Box and whiskers plot describing the variation of the classification errors, evaluated on a particular dataset according to the selected bias configurations

The corresponding meta-features have been extracted and used as input of the fuzzy meta-rule base, in order to infer a suggested bias configuration for each task. In most cases, the inferred bias values are in agreement with those which produced the best results during the base-learning experiments for tasks in the same domain. This can be read as a confirmation of the discriminating power of the chosen meta-features, when evaluated for tasks in a common environment. The appropriateness of the inferred results has been also assessed by executing a 10-fold cross validation session of experiments, involving both the suggested bias configurations and the previously considered parameter settings $\{Bias_i\}_{i=1}^5$. For the sake of illustration, we report the results concerning the analysis of a dataset which refers to a particular (previously unseen) task inside one of the synthetic domains. As it can be seen in figure 2, the inferred bias value helps to efficaciously learn the task at hand, and the obtained results in terms of accuracy are comparable with those produced by the bias previously labelled as the “best choice” inside the particular domain of reference.

4 Conclusive Remarks

The MINDFUL system offers the opportunity to put in practice a meta-learning strategy, in order to overcome the limitations of base-learning approaches. By diverging from the widely proposed approaches of model combination, our methodology focuses on the learning behaviour of a single model, thus qualifying a peculiar idea of meta-learning. The neuro-fuzzy integration realised by the KERNEL of the MINDFUL system allows to exploit the learning capabilities of neural networks to perform a process of knowledge extraction from data. By arranging the derived information in form of fuzzy rule base, it is possible to build up a comprehensible knowledge which is useful to tackle base-level tasks. The definition of

suitable dataset meta-features and the process of performance assessment constitute crucial steps to construct a meta-training set, which represents the basis for deriving a kind of meta-knowledge to be applied in cross-task applications.

The experimental sessions presented in this paper constitute a first attempt to ascertain the capabilities of the system over an ensemble of synthetic domains, purposely designed for this preliminary validation. However, these kind of simulations furnished an empirical evidence of the discriminating power of the chosen meta-features, when evaluated for tasks in a common environment. Moreover, the experimental results confirmed that different categories of learning problems should be tackled with the adoption of different biases.

The obtained results encourage the further application of the MINDFUL system in more complex real-world problems. Moreover, future research efforts should be addressed to the development of suitable life-long learning strategies, enabling a form of self-adapting learning. This could take place by continuously updating the meta-training set as novel base-level tasks are encountered.

References

1. Wolpert, D. H., Macready, W. G.: No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation* **1**(1) (1997) 67–82
2. Thrun, S., Pratt, L., (eds.): *Learning to Learn*. Kluwer Academic Publisher (1998)
3. Vilalta, R., Drissi, Y.: A perspective view and survey of meta-learning. *Artificial Intelligence Review* **18** (2002) 77–95
4. Giraud-Carrier, C., Vilalta, R., Brazdil, P.: Introduction to the special issue on meta-learning. *Machine learning* **54** (2004) 187–193
5. Kalousis, A., Hilario, M.: Model selection via meta-learning: a comparative study. In: *Proc. of the 12th Int. IEEE Conference on Tools with AI* (2000)
6. Ortega, J., Koppel, M., Argamon, S.: Arbitrating among competing classifiers using learned referees. *Knowledge and Information Systems* **3** (2001) 470–490
7. Soares, C., Brazdil, P., Kuba, P.: A meta-learning approach to select the kernel width in support vector regression. *Machine learning* **54** (2004) 195–209
8. Castellano, G., Castiello, C., Fanelli, A. M., Mencar, C.: Knowledge discovery by a neuro-fuzzy modeling framework. *Fuzzy Sets and Systems* **149** (2005) 187–207
9. Castiello, C.: *Meta-Learning: a Concern for Epistemology and Computational Intelligence*. PhD Thesis. University of Bari - Italy (2004)
10. Michie, D., Spiegelhalter, D. J., Taylor, C.: *Machine learning, neural and statistical classification*. Ellis Horwood Series in Artificial Intelligence (1994)
11. Castiello, C., Castellano, G., Fanelli, A. M.: Meta-data: characterization of input features for meta-learning. In: Torra, V., Narukawa, Y., Miyamoto, S. (eds.): *Modeling decisions for Artificial Intelligence*. *Lecture Notes in Artificial Intelligence*, Vol. 3558. Springer-Verlag, Berlin Heidelberg New York (2005) 457–468

Learning Cooperation from Classifier Systems

Trung Hau Tran, Cédric Sanza, and Yves Duthen

Paul Sabatier University – Toulouse III, 118 route de Narbonne,
31062 Toulouse Cedex 4, France
hau@irit.fr

Abstract. This paper deals with cooperation for virtual reality applications. In a multi-agent system, cooperation between agents is an important element to solve a common task, which is very difficult or impossible for a single agent or a group of agents without cooperation. Hence we focus on cooperation in the predator-prey problem where a group of programmed and learning predators coordinates their actions to capture the prey. These actions of a learning predator are dynamically weighted by a behavioral system based on motor schemas and classifier systems. At each instant, the system must modify the weights in order to enhance the strategies of the group, as surrounding a prey. Thanks to the classifier system the learning predator learns situations and gradually adapts its actions to its environment. First encouraging results show that coupling such systems gives very efficient performances in dynamic environments.

1 Introduction

The most common phenomenon we can observe in the nature is a group of predators chasing a prey. Simulating such a problem requires a set of cooperative agents that (1) coordinate their behavior, (2) adapt their behavior to the environment, and (3) anticipate the behavior of their partners to efficiently capture the prey, or generally to perform a common task better. Moreover, each one has a specific role in a group, depending on its skill and its situation. For example, a predator which moves at slow speed should block the escape of the prey; another predator which moves at high speed should pursue the prey or prevent it at the other side and cooperates with the first predator to surround it. If they move at the same speed as the prey, they might follow it while the prey continues to escape without any danger lying in front of it. Finally, the predators in pursuit of the prey without cooperation would not be able to catch it. Thus, an alliance between the predators is necessary to achieve better chases. Various methods have already achieved the goal of learning cooperative behaviors such as classifier systems [15], neural networks [19], evolutionary algorithms [13]...

Agents have to be able to memorize situations to trigger the corresponding actions to obtain cooperation. A first approach could consist in evolving some parameters to control the behaviors. Evolving these parameters can generate complex behaviors, for instance strategic positions of the agents. Arkin uses this method in the motor schema [1] which enabled him to simply and efficiently control the navigation of a mobile robot. The core of his approach is a behavioral fusion of the outputs of all motor schemas in an only vector to guide the robot moving. The main advantage is the

possible parallelization of the motor schemas for a running cycle in real time. One difficulty is to dynamically modify the schema parameters corresponding to each situation rather than to keep the fixed ones in order to increase the efficiency of the robot. So, combining this approach with a rule-based system will help the learning agent to learn situations and cooperatively work with the others, and to use its corresponding schema parameters.

Our research aims at developing anticipatory cooperative systems. Indeed, a group develops an anticipatory tacit coordination by anticipating the behaviors of the others after a learning stage to perform the strategies of group. Nevertheless, the results reported in this paper are not well entered into the anticipatory tacit coordination, but a coevolution of the agents is developed. Indeed, an agent adjusts its behaviors by considering its two neighbors; besides a learning agent needs to learn how to build its model of the world and how to choose its actions in situations through the rule-based system in order to cooperate with its group.

The structure of this paper is organized as follows. At first, the related works in the cooperation will be presented. In section 3 we review briefly the methods used such as Arkin's motor schemas and classifier systems; and describe the predator-prey simulation settings, the behavioral predators and our proposal. Section 4 contains the results of the simulation. A discussion of future work concludes this paper.

2 Related Work

[7] developed the coordination by learning from evaluations. The learning agent uses coordination signals [6], which are received from its two neighbors, programmed agents, to learn how to coordinate its actions in order to perform the pursuit problem of which four agents (one learning agent and three programmed ones) capture a prey. The action selection of the learning agent bases on the Exploration Buckets algorithm [5] which encourages the balance between exploration and exploitation. Some actions, ignored in a moment because of their high prediction errors, are still selected to improve their performances.

[13] developed punctuated anytime learning (PAL) to genetic algorithms (GAs), i.e. GA is executed every G generations, called punctuated generations, instead of each one. In the predators-prey scenario a GA is applied to four subpopulations representing four predators. In [2] instead of all chromosomes in a subpopulation N ones randomly selected are tested against all ones in the other subpopulations. The representative chromosome from each subpopulation from the last punctuated generations is used during non-punctuated one. This process reduced computation time required to find an accurate solution. A GA is replaced by cyclic genetic algorithm (CGA) in which a chromosome is composed of two cycles of nine genes. The first cycle defines the motions of the predators before seeing a prey, and the second cycle defines their motions after seeing it. Each gene has two parts $(T_i R_i)$ where T_i is an action and R_i is the number of repetitions of this action. The CGA is perfectly fit for learning cycle behavior strategy as surround a prey [2].

[19] proposed the Multi-Agent ESP (Enforced Subpopulations) of which the agents are controlled by the neural networks. This method is to coevolve them in separate subpopulations to control a team of predators to chase a prey. This approach is proved to be more efficient and robust than a central controller for all agents.

3 Behavioral System

3.1 Motor Schemas

The motor schemas [1] are reactive schema-based robot navigation. They are a variation of the potential fields' method. A motor schema corresponds to a primitive behavior, e.g. avoid-obstacle; move-to-goal; stay-on-path... Its output is a force to command the robot moving, each active motor schema contributes somewhat to the final robotic action. Complex observable behaviors emerge from the combination of these forces. This combination is controlled by the schema parameters. In [1], they are given by the programmers while [14] determined these parameters autonomously by using the GA. [4] developed a learning momentum which is reinforcement learning. His mobile robot sees itself in one of the predefined situations and adjusts the schema parameters corresponding to this situation. It is attracted by our attention. Our learning agent will learn situations instead of predefined ones.

3.2 Classifier System

The classifier system (CS) introduced by Holland [8] is a reactive system. It is composed of three parts: a set of rules [P] called classifiers interacting with the environment, an evolutionary algorithm and a credit assignment. A rule has a condition part, an action part, and a strength evaluating its performance. An agent receives a binary encoded message from the environment via detectors, and then it is compared with the condition part of each classifier in order to determine appropriate classifiers which are classed in a current match set [M]. The winner classifier chosen from [M] is selected through the efficiency of its strength. The action part of the winner performs an indicated behavior in the base of behaviors via effectors. After that a credit assignment is used to evaluate the usefulness of the selected classifier to update its strength. The CS increases its performances by using a GA with genetic operators like crossover, mutation to modify and generate new classifiers to adapt to any situations. The applications of the CS can be found in the soccer simulation [15], in robotics [9]... The extensions of the CS are such as XCS [18], Anticipatory Classifier System (ACS) [16], Neural Classifier System [11]...

3.3 Predator-Prey Simulation Settings

In our simulation of the predator-prey problem based on [10], one prey has the primitive behaviors such as: find a nearest food source, go to a target, eat a food source, and escape predators. The prey moving is programmed as a deterministic function of food sources and there is any randomness involved. The prey finds a nearest food source, goes to it and eats it. In facing a predator, the prey will choose an opposite direction in order to evade it; and in the case there are many predators around the prey, it calculates the opposite direction for each predator, combines ones to form one final direction, and then takes it. When there are any predators detected in the perception of the prey, it pursues its goal, i.e. a nearest food source, or it will continue to escape the predators during N simulation steps if its last action is an escape action. The prey is captured if positions around it in a small rayon R are occupied by all predators.

There are two types of predator such the programmed and the learning one. The programmed predators perform a circulating motion strategy around prey. However, when a programmed predator (called predator “parent”) is absent or dead, a second type of predator (called predator “child”) learns and plays its role. We can say the child replace their parents in order to guarantee the collective strategy against prey. Only the learning predators are controlled by the classifier systems that memorize a pair of situation/action called rule, generate and evolve rules to obtain better cooperation in a group of predators. The predators can visually detect the prey from any distance. All entities can move in any direction on the plan of the 3D environment. In the experiments described in section 4, one prey, four programmed predators and one learning predator are used.

The perception of a predator is an encoded message of positions between the predators relative to the prey. In Fig. 1(a), we divide the circle into twelve sectors, the center of the circle is the position of the prey, the symbol ‘x’ in a sector represents a predator in the perception of the prey, with sensor code 1; blank sectors have sensor code 0; hence an encoded message of this situation is 010001010010. The left-hand first bit is always sensor code of the first sector (noted ‘1’ in Fig. 1(a)), with the remainder corresponding to sensor codes of the sectors proceeding counterclockwise.

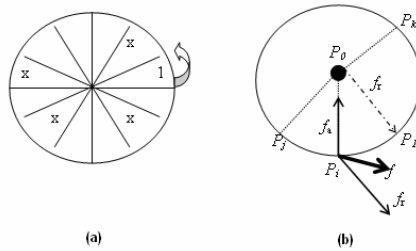


Fig. 1. (a) Perception of a predator, (b) Outputs of 2 motor schemas f_a, f_r

Each predator calculates two motor schemas (behaviors). The output of the motor schema **Go-to-target** is a force $f_a (\in \mathfrak{R}^3)$ which is attracted by a target. The output of the motor schema **Evade-target** is a repulsive force $f_r (\in \mathfrak{R}^3)$ in order to repel from a target. The final force $f = \delta_a f_a + \delta_r f_r$ guides the predator moving. The schema parameters $\delta_a, \delta_r (\in \mathfrak{R})$ correspond to two motor schemas **go-to-target** and **evade-target**, respectively. Their values are predefined to each programmed predator, but are adjusted by the CS to the learning predator. That is the difference between a programmed and a learning predator. In Fig. 1(b), P_i is a position of a predator, P_j, P_k are positions of its two neighbors, P_0 is a target where the predator P_i must go to, P_1 is calculated based on the mean of the difference between two angles $\overrightarrow{P_j P_0 P_i}, \overrightarrow{P_i P_0 P_1}$, $f_a = \frac{\overrightarrow{P_i P_0}}{\|P_i P_0\|} g(\|P_i P_0\|)$, $f_r = \frac{\overrightarrow{P_0 P_1}}{\|P_0 P_1\|} g(\|P_0 P_1\|)$, $g(\cdot)$ is the value of the potential function (e.g. $g(x) = \frac{1}{1+x}$) and the final force $f = \delta_a f_a + \delta_r f_r$ drives the predator moving P_i .

3.4 Learning Predator

Our proposal is to let the autonomous agent learn how to model the environment via the interaction between itself and the other agents using a classifier system (CS). This system learns situations, decides a corresponding action and triggers it to obtain cooperation with the other agents. A classifier has the form of c_i/a_i , endowed with several attributes such as strength, fitness... The classifier condition c_i of length twelve is initially generated with random values from the ternary alphabet $\{0, 1, \#\}$, where # denotes a “don’t care” symbol and acts as 0 or 1. Nine possible actions in the classifier action are (see Table 1). Each action is corresponding to the adjustment values of the schema parameters δ_a, δ_r . Each value indicates the amount of change for the given schema parameter. For example, the condition part of some classifiers such as 010#010#001#/a₃, #1000#010#10/a₇, 01#00101#####/a₉... matches the encoded message 010001010010 given in section 3.4.

Table 1. 9 actions in the action part of each rule

Action a_i	δ_a	δ_r
Action 1	+0.1	+0.1
Action 2	+0.1	0.0
Action 3	+0.1	-0.1
Action 4	0.0	+0.1
Action 5	0.0	0.0
Action 6	0.0	-0.1
Action 7	-0.1	+0.1
Action 8	-0.1	0.0
Action 9	-0.1	-0.1

Our CS bases on YCS [3]. Each classifier is endowed with three principal attributes: a prediction p_j evaluating its strength, a prediction error ϵ_j indicating an estimate of the error in each classifier prediction and a fitness f_j used in the GA. In the opposite of the traditional CS [8], the classifier prediction here is used in the action selection and the classifier fitness in the reproduction. When the CS receives a reward P from the environment, we update three attributes of the appropriate classifiers with learning rate β as in [3]:

$$\epsilon_j \leftarrow \epsilon_j + \beta (\|P - p_j\| - \epsilon_j) \tag{1}$$

$$p_j \leftarrow p_j + \beta (P - p_j) \tag{2}$$

$$f_j = \frac{1}{1 + \epsilon_j} \tag{3}$$

We execute the GA every twenty five generations. When called, roulette wheel selection chooses two parent classifiers based on their fitness and offspring are produced through two genetic operators: mutation (probability μ) and crossover (single point with probability χ). The GA is not used in exploit tests. Integrating the classifier system into motor schemas is shown in Fig. 2.

The reward plays an important aspect in the learning. It reinforces actions of learning agents. Here, the programmed predators send some signals ([6] called coordination signals) to the learning predator who compares them with its reasoning

decision gotten from the combining systems. We define a signal as a vector guiding the predator moving. The difference between those signals and the reasoning decision is interpreted as reward to update the appropriate classifiers advocating the last action.

$$reward = \cos(angle(\vec{v}_{signal}, \vec{v}_{combining_systems})) \tag{4}$$

where v_{signal} is a signal received from the programmed predator; $v_{combining_systems}$ is the output of the combining system and $angle(.)$ calculates the angle between these vectors. Since the programmed predators use a same algorithm to calculate a signal, the learning predator will receive two identical signals from its neighbors. So, we decide to take into account just one of those signals to reduce the computation time.

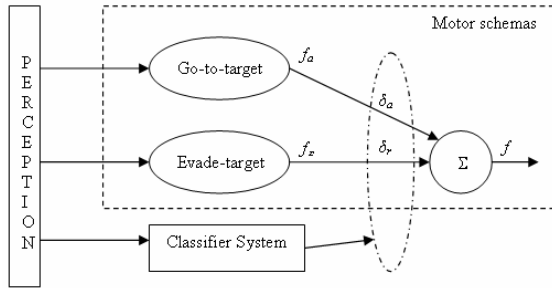


Fig. 2. Integrating the classifier system into motor schemas

4 Simulation Results

We run 3 different tests for 100 rounds each, 15000 generations per round. We only record results from exploit rounds, and then calculate the mean of the 3 tests. A capture rate [7], i.e. fraction of the rounds which ends in a capture, measures the performance of the systems. The CS of the learning predator has 12 detectors, 9 effectors, a population size $N_{pop_size}=800$, $\beta=0.2$, $\mu=0.002$ and $\chi=0.5$. Fig. 3 shows the average fitness of its population from explorer rounds. During this time, its performance gradually increases. In the other words, the learning predator learns and adapts step by step its actions to the programmed ones in cooperation. This average fitness is recorded every 25 generations in each round.

Fig. 4(a) shows the capture rate from exploit rounds (90%), which is close to the capture rate of a group of programmed predators with cooperation (92%) (Fig. 4(b))

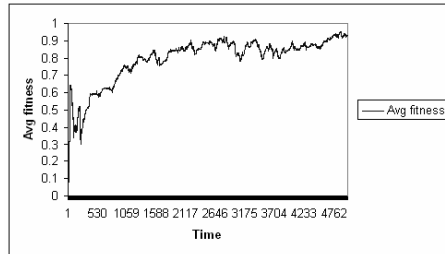


Fig. 3. Avg fitness of the population in the CS of the learning predator from explorer rounds

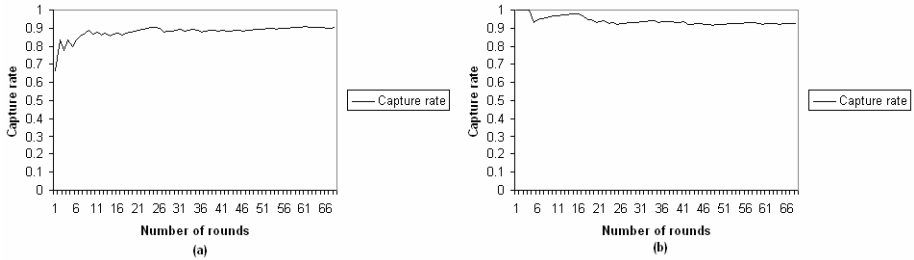


Fig. 4. Capture rate (a) from exploit rounds of a group of four programmed and one learning predators with cooperation, (b) of a group of five programmed predators with cooperation

and higher than the one of a group of programmed predators without cooperation (15%) (not shown). During this first time, the learning predator continues to learn cooperation in order to obtain its accurate classifiers. After about 50 exploit rounds, the learning is stable, the classifiers are accurate, and the prey is captured in major case. We can see the capture rate in the last 50 rounds in Fig. 4(a) which is close to the one in Fig. 4(b).

Our system is hard to compare with the system of [7]. A first difference is that our programmed predator uses the motor schema approach to guide the predator moving in cooperation instead of choosing a moving (left, right, up, down, stand by) by maximizing received coordination evaluations [7]. There is also a difference in the definitions of the reward function.

5 Conclusion

The result in this paper revealed a cooperative strategy of a group of programmed and learning predators to surround a prey. We proposed a combination of the motor schema approach and the classifier system to learn how to use the schema parameters. Hence, our approach is proved to be close to the performance of a group of programmed predators with coordination and more efficient than the approach of which the programmed predators individually pursue their target without cooperation. The performance of the systems of the learning predator depends on signals sent by the programmed predators. These signals are used to evaluate actions of the learning predator. So, we can extend our systems to perform other strategies simply by modifying the algorithm defined in the programmed predators.

Further, we will accommodate each learning predator with the anticipation to look ahead and to act according to the future states in order to obtain a desired state or avoid undesired one at present; for example anticipations of trajectories of its prey [17], of its partners to achieve a foraging task [12]... We will also investigate the anticipatory tacit coordination, i.e. a predator anticipates the behavior of the other predators in a group to adapt its behaviors at present.

References

1. Arkin, R.C. (1992). "Behavior-based robot navigation for extended domains". *Adaptive Behavior*, 2(1):201–225.
2. Blumenthal, H.J., and Parker, G.B. (2004). "Co-Evolving Team Capture Strategies for Dissimilar Robots". *The AAAI 2004 Symposium on Artificial Multiagent Learning*, October 2004, Washington, DC.

3. Bull, L. (2003). "A Simple Accuracy-based Learning Classifier System". UWELCSG03-005.
4. Clark, R.J., Arkin, R.C., and Ram, A. (1992). "Learning Momentum: On-line Performance Enhancement for Reactive Systems". *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*. Nice-France, May 1992.
5. De Jong, E.D. (1997a). "An Accumulative Exploration Method for Reinforcement Learning". *Notes of the AAI'97 Workshop on Multiagent Learning*, AAI technical report WS-97-03.
6. De Jong, E.D. (1997b). "Multi-Agent Coordination by Communication of Evaluations". *Proceedings of the 8th European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, MAAMAW'97, 1997.
7. De Jong, E.D. (1999). "Coordination Developed by Learning from Evaluations". J.A. Padget (ed.) *Collaboration between Human and Artificial Societies*, Springer-Verlag LNAI Vol. 1624.
8. Holland, J.H. (1986). "Escaping Brittleness: The Possibilities of General-Purpose Learning Algorithms Applied to Parallel Rule-Based Systems". In Mitchell, Michalski, and Carbonell, editors, *Machine Learning, an Artificial Intelligence Approach*. Volume II, chapter 20, pages 593-623. Morgan Kaufmann, 1986.
9. Katagami, D., and Yamada, S. (2000). "Real robot learning with human teaching". *Fourth Japan-Australia Joint Workshop on Intelligent and Evolutionary Systems*, pages 263-270.
10. Miller, G., and Cliff, D. (1994). "Co-evolution of pursuit and evasion I: Biological and game-theoretic foundations". *Technical Report CSRP311*, School of Cognitive and Computing Sciences, University of Sussex, Brighton, UK.
11. Nakano, R. (2000). "Efficient Learning of Behavioral Rules". From Animals to Animats 6: SAB2000 Proceedings Supplement of the Sixth International Conference on Simulation of Adaptive Behavior, p178-184. Paris, France (2000).
12. Panatier, C., Sanza, C., and Duthen, Y. (2000). "Adaptive Entity thanks to Behavioral Prediction". From Animals to Animats 6: SAB2000 Proceedings Supplement of the Sixth International Conference on Simulation of Adaptive Behavior, p295-303. Paris, France (2000).
13. Parker, G.B. (2002). "Punctuated Anytime Learning for Hexapod Gait Generation". In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2664-2671. (IROS 2002).
14. Ram, A., Arkin, R.C., Boone, G., and Pearce, M. (1994). "Using genetic algorithms to learn reactive control parameters for autonomous robotic navigation". *Adaptive Behavior*, 2(3):277-304.
15. Sanza, C., Panatier, C., and Duthen, Y. (2000). "Communication and Interaction with Learning Agents in Virtual Soccer". *VW'2000, 2nd International Conference on Virtual Worlds*. LNCS VOL. 1834. Juillet 2000, Paris, France.
16. Stolzmann, W. (1998). "Anticipatory Classifier Systems". In Koza, John R. et al. (editors). *Genetic Programming 1998: Proceedings of the Third Annual Conference, July 22-25, 1998, University of Wisconsin, Madison, Wisconsin*. San Francisco, CA: Morgan Kaufmann. 658-664.
17. Tran, T.H., Sanza, C., and Duthen, Y. (2004). "Study of the anticipatory system in simulation". *3IA'2004, 7th International Conference on Computer Graphics and Artificial Intelligence*. May 2004, Limoges, France.
18. Wilson, S.W. (1995). "Classifier Fitness Based on Accuracy". *Evolutionary Computation* 3(2):149-76.
19. Yong, C.H., and Miikkulainen, R. (2001). "Cooperative Coevolution of Multi-Agent Systems". *Technical Report AI-01-287*, February 2001.

Location Management Using Hierarchical Structured Agents for Distributed Databases

Romeo Mark A. Mateo, Bobby D. Gerardo, and Jaewan Lee

School of Electronic and Information Engineering, Kunsan National University,
68 Miryong-dong, Kunsan, Chonbuk 573-701, South Korea
{rmmateo, bgerardo, jwlee}@kunsan.ac.kr

Abstract. Location-based services (LBS) depends on data gathered from mobile and ubiquitous devices and use to provide services like getting the appropriate location of a mobile user presented in physical and logical maps. The main operations of location management in LBS are updating and searching or paging. Some studies to improve these were presented by using optimal sequential paging and location area schemes. Different LBS means variety of methods on accessing data that leads to complexity of providing services. In this paper, we use an approach of hierarchical structured agents for the method of locating a mobile object in a location-based service. This study focuses on location management by using agents. Agents were used for accessing the distributed databases on LBS. It also introduces a hierarchical searching method that uses a nearest neighbor technique for fast searching. The result of using the technique shows an improved searching method in the location management.

1 Introduction

One of the most important functions of mobile computing environment is determining the mobile user's current location. Knowing the location of a certain object or mobile user is necessary information for location-based services or LBS [1] [2] [3]. The LBS has the ability to find the geographical location of the mobile device and provide services based on its location. Many people expect a high potential of location-based services such as city guides or navigation systems for m-commerce scenarios.

Information about the moving objects is important and not all location services have the same database scheme to process mobile objects. Software agents [4] can solve the problem in the distributed location services. The agents will share its data with the other parts of the system so the information of mobile objects can be known to other location service.

In this paper, we present an approach of using agents for handling the data within the distributed environment. The architecture shows the design of a distributed location-base services managed by location agents to process the queries. The location management of the proposed system presents a hierarchical process of updating, searching and other concerns to provide efficiency. It also introduces a hierarchical searching method with the use of nearest neighbor technique for fast searching.

2 Related Works

2.1 Cellular Networks

Location management keeps track of an active mobile station in cellular networks [5]. There are many research works which tackle down the methods of locating a mobile object in cellular networks like in [6], [7] and [8]. The main operation that deals with researches is on location updating and paging of a mobile station. There has always tradeoff between the frequency of location updates and paging costs. Some researches introduce schemes to minimize the cost of paging and updating of locations. Optimal sequential paging [6] presents a polynomial-time algorithm to solve the problem of minimizing the average paging costs. A predictive distance-based mobile tracking scheme is presented in [7] and dynamic location update schemes in velocity-based scheme [8] are also presented.

2.2 Distributed Databases

A distributed database is an integrated database which was built on top of a computer network rather than on a single computer. Databases may involve different database management systems, running on different architectures that distribute the execution of transactions. This was solved by distributed database management systems discussed in [9]. The distributed databases create distribution transparency to the users. Use of hierarchical distributed location database for tracking mobile objects was discussed by Pituora et. al., [12]. Here the nodes are networked in a tree-like structure.

2.3 Software Agents

Software agents are autonomous being self-contained and capable of making independent decisions and taking actions to satisfy internal goals based upon their perceived environment. A technique uses mobile agents [10] that can access databases in different nodes which can be a solution in the distributed database. A research of hierarchy of agents in the network of nodes that acts as mobile location databases that replicate to other base-station [11] can be a model of the location management using agents.

3 Architecture of the Proposed System

The goal of the proposed system is to provide an efficient location management from distributed databases on location-based services. Figure 1 shows the architecture of the proposed system. A location-based service has location agent which can access the database and communicates with its parent agent. They are arranged in a hierarchical tree-like structure of networks. In the first process, mobile objects are stored to the database of LBS. The location agents will copy and store the mobile object and pass the value to its parent agent. The update process to parent agents will continue until it reaches the root node of the agents. The next process is searching for the mobile object in which the process is done in the local database. If the data is not found in the node then the location agent will verify it on the other location agent until it

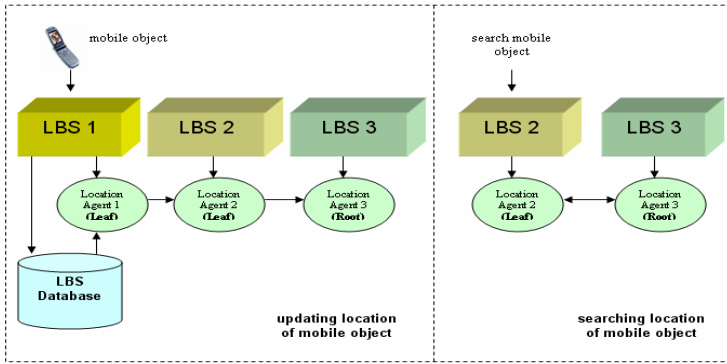


Fig. 1. Proposed location management

reaches the root agent. If the mobile object is found on the other location agents then it will return the value of the node location where the mobile object is registered.

3.2 Location Agents

In [4] the location system uses software agents to solve data management problem. The location agent will listen to all location data of all available location services. The proposed system includes a location agent in each location service which could access database of location service and can communicate to other agents.

3.3 Location Management

The proposed system used hierarchical structures of location agent in which the agents will communicate with its parent node in case of updating or searching for mobile objects. This will avoid the sequence of many lookups in each databases of the network by replicating the object in each node.

3.3.1 Updating and Searching Method

A child node will gather the mobile object data and it will pass the value to its parent node by the location agents. The process of updating will repeat until it reaches the root node. The root node will hold all the data objects under its sub nodes.

A search method is shown in Figure 2 that consists of 4 root nodes. A mobile object data at node V will be stored on agent V and it will be passed on node S until Root 4. A query in node W searches for mobile ID 345789. If the mobile ID is not found in node V then it will communicate to its parent (node T). The search method will be repeated until it reaches R4. The end point of the search within the nodes is in the root and if it is still not found it will redirect the query to other roots (i.e., R1, R2, and R3).

3.3.2 Node Failure and Recovery

The failure of nodes is managed in the proposed system. In the case of the leaf nodes, agents know its parent’s parent so that if there is no communication between the parents, the agent will look for it to establish the link. If the root node fails the link with

its sub nodes, then the sub nodes will look for the available root and pass the data to the new root. If the node recovers, then the sub nodes will establish link to its original parent. All data will be updated on its sub nodes and delete all the obsolete data.

4 Performance Analysis

4.1 Nearest Neighbor Algorithm

The nearest neighbor (NN) feature method is that distance between the groups is defined as the distance between the closest pair of objects, where pairs consisting of one object from each group are concerned. In this case we have a set of nodes in a mesh network and each node has specific distance to each other. The NN algorithm will be used to locate the nodes which have the least distance from the parent node which is presented by equation 1 below:

$$\sum dist(n_i, n_x) < dist(n_i, p_i) \text{ where } x = 1, 2, \dots, n \tag{1}$$

The n represents a node, n_i is the source node and p_i indicates its parent node of n_i . The algorithm will be used to locate the nodes which have the least distance from the parent node. It will identify all the near nodes and determine the count of the nodes.

4.2 Improved Search Method

The improved search method uses a NN algorithm which detects a near location node to execute the search method. First, it will execute the NN algorithm and if the query is not found in the near nodes, it will send the query to the parent node. If the query is valid then it will return the value of the query back to the user.

In Figure 2, we compare the search methods and there is only one hop count on the improved search method and the latter has 3 hop counts. Also, we calculate the average database cost and compare with the improved search method in Figure 4 and shows that it is less than the normal search. The pseudo-code is presented in Figure 3.

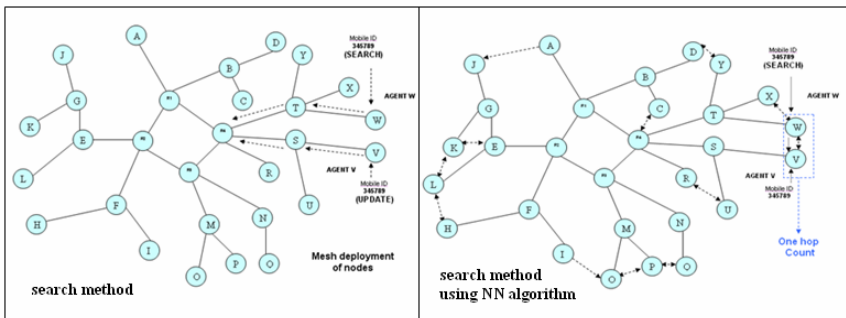


Fig. 2. Comparison of search methods

```

//search the nearest neighbor node
List all Node = dist(Node, SourceNode) < dist(ParentNode, SourceCode)
i = 0
For all List Do
    Search (MobileID, Node(i))
    If BackOriginalNode = True Then End
    i = i + 1
Loop
//search the parents up to the root
GoTo ParentNode

Function Search (MobileID, Node) //Search function
{
GoTo Node
//Search the list of mobile in Location Agent
Do all List of mobileID in LocationAgent
    If LocationAgent.MoblieID(List) = MobileID Then Return Node Location
    BackOriginalNode = True
Loop While not end of data
    
```

Fig. 3. Pseudo-code of search with NN algorithm

5 Experimental Evaluation

The simulation used a database consisting of 34 nodes, 80 mobile ID and 372 data replicated by agents were stored within the network. The platforms used in the re-search were IBM compatible PC with Windows and Linux OS, CORBA, and Java.

The experiment utilized a network of computers to implement the simulation. CORBA was used for the simulation to perform interoperability of the computer’s operating system. The improved technique used the NN algorithm which enhanced the efficiency and provide fast search if the mobile object is near the node being searched.

It was observed that there were an additional number of hops when implementing the improved algorithm which depends on the numbers of nearest nodes but the search was faster. Figure 4 shows hop count results. Two cases were described in the situation, first one was the search having nearest node but the mobile object is not found there. Second, the search is found on one of the nearest node. It shows that if the search is found in the nearest node then the hop counts would be minimized and if the search is not found in the nearest node then the value of hop count would increase. This implies that the former will perform faster searching.

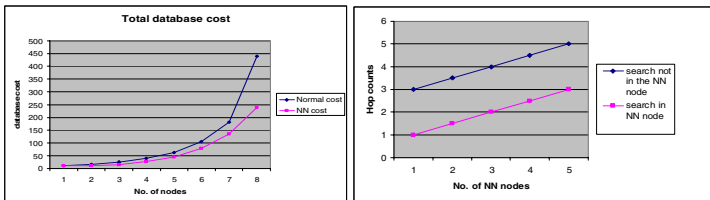


Fig. 4. Graphical results

6 Conclusion and Future Work

In this paper, we used an approach of hierarchical structured agents for location management of location-based services. It used location agents on nodes that perform data replication. A hierarchical structure of updating and searching mobile objects was also provided. It also improved the location management by implementing the nearest neighbor algorithm. The searching does not need to go to all of the parent nodes when the mobile object's identification is found on the nearest nodes of the source node which implies faster searching. The graphical result implies that the technique of using NN algorithm can minimize hop counts and the database cost will be lessened.

The research is only limited on mobile tracking, it does not include physical and logical mapping. The additional functions of the LBS would be studied in the future.

References

1. Maneesh Prasad.: Location Based Services. Available at <http://www.gisdevelopment.net/technology/lbs/techlbs003.htm>.
2. Smita Sengupta.: Open Standards in Location Based Services. Available at www.gisdevelopment.net/technology/lbs/techlbs002.htm
3. Jorg Roth.: Flexible Positioning for Location-Based Services. IADIS International Journal on WWW/Internet, Vol. 1, No. 2, pp. 18-22, 2004.
4. Peter Jaric.: An Agent-Based Location System. Uppsala, unpublished Master's Thesis in Computing Science 141. Available at <http://www.jaric.org/thesis.pdf>.
5. Jingyuan Zhang.: Location Management in Cellular Networks. Available at www.cse.fau.edu/~jie/teaching/fall_2004_files/locationmanagement.pdf.
6. Bhaskar Krishnamachari, Rung-Hung Gau, Stephen B. Wicker, and Zygmunt J. Haas.: Optimal Sequential Paging in Cellular Networks. Available at ceng.usc.edu/~bkrishna/research/papers/OptimalSequentialPaging.pdf.
7. Ben Liang and Zygmunt J. Haas.: Predictive Distance-based Mobility Management for PCS networks. Proceedings of the Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, IEEE INFOCOM, '99, vol. 3, pp.1377-84, 1999.
8. G. Wan and E. Lin.: A Dynamic Paging Scheme for Wireless Communication Systems. Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, MOBICOM 1997, pp.195-203.
9. M. Tamer Ozsu.: Distributed Databases. Available <http://db.uwaterloo.ca/~ddbms/publications/ozsu/Distdb/distdb.pdf#search=Distributed%20Databases'>.
10. Russell P. Lentini, Goutham P. Rao, Jon N. Thies, and Jennifer Kay.: EMAA: An Extendable Mobile Agent Architecture. Available at www.cse.unsw.edu.au/~sjha/papers/icc04lee.pdf.
11. Kevin Lee, Hong Wing Lee, Sanjay Jha, and Nirupama Bulusu.: Adaptive, Distributed Location Management in Mobile, Wireless Networks.
12. Evangelia Pitoura and Ioannis Fudos.: Distributed Location Databases for Tracking Highly Mobile Objects. The Computer Journal, Vol. 44, No.2, 2001.
13. Roy Want, Andy Hopper, Veronica Falcao, and Jonathan Gibbons.: The Active Badge Location System. Available at <http://www-lce.eng.cam.ac.uk/publications/files/tr.92.1.pdf>.
14. Jianliang Xu and Dik Lun Lee.: Querying Location-dependent Data in Wireless Cellular Environment: Available at http://www.w3.org/Mobile/posdep/query_xujl.html.Jeoung

On_line Measurement System of Virtual Dielectric Loss Based on Wavelets and LabVIEW and Correlation Technics

BaoBao Wang and Ye Wang

School of Computer Science and Engineering, Xidian University,
Xi'an 710071, China

bbwang@mail.xidian.edu.cn, wangye_110263.net

Abstract. This paper presents the principles and instrument structures of on_line measurement system for dielectric loss of virtual capacitive_type equipment. The system based on wavelets, LabVIEW, GPRS network and SPC correlation technics, which can efficiently solve the problem of eliminating noises from signals, removing the electromagnetism interference to measure the results of influence, global area and full auto on_line measurement, to attain the higher precision of measurement.

1 Preface

This analysis technics effectively removes noises from signals with wavelets, utilizing related analyzes that measure the value of $\tan\delta$, its accuracy and stability are superior to the traditional methods. By applying LabView, improving the measurement performance and intelligences and providing convenience.

2 The Principle of Measurement for Dielectric Loss

$\tan\delta$ is tested through synchronously high-speed data acquisition of the secondary voltage of the voltage and the test of the electronic leaking of the equipment. Thus, in order to ensure the accuracy of the measurement, wavelets shift should first be applied to get rid of the noises of the collected signals, relevant analysis technologies should be applied to make out the value of $\tan\delta$.

2.1 Wavelets Eliminating Noise

The re-arrangement of the signals after multi-analysis of the signals and threshold treatment of the wavelets parameters will make it possible to get rid of the noises and keep the real signals, the basic steps are:

(1) Selecting the suitable wavelet and decomposition level, decomposing signal that is mixed with the noises.

According to the theory of multi-analysis of the signals:

The signal f is the sum of g_1, g_2, \dots, g_m and f_m , i.e:

$F = g_1 + g_2 + \dots + g_m + f_m \in V_0$, for any $j(j=1, \dots, M)$, all have $f_j(t) \in V_j$, $g_j(t) \in W_j$, and W_j is the orthogonal space of $V_j, V_j, W_j = V_j + 1$, scale function

$\varphi(t)$ is the generate element of V_j , wavelet function $\varphi(t)$ is the W_j , and existing sequence P_k and q_k satisfy:

Hypothetically, the sequences a_k and b_k are the discretization projection of V_j and W_j on f , then having equation below:

$$C_k^{j-1} = \sum_l a_{i-2k} c_l^j \qquad d_k^{j-1} = \sum_l b_{l-2k} c_l^j \qquad (1)$$

Decomposition process as Fig. 1. The acquisition signal is C^0 , the signals after multi-analysis of the signals are high-frequency, d_1, d_2, \dots, d_m and low-frequency component: C^M .

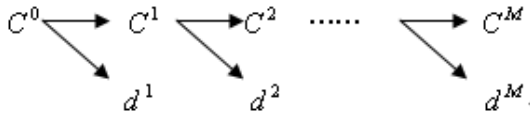


Fig. 1. The acquisition signal is C^0 , the signals after multi-analysis of the signals are high-frequency, d_1, d_2, \dots, d_m and low-frequency component: C^M

(2) The threshold quantization of high frequency coefficient of wavelets decomposition.

In decomposition process above, d^1, \dots, d^M and C^1, \dots, C^M , corresponding to high-frequency and low-frequency coefficient of wavelets decomposition of the signals. The noises of signals are always included in high-frequency component, but the useful signals in are low-frequency component. Therefore the spectrum component each dimensions generated by the noise is assigned to zero, namely putting d^1, \dots, d^M to zero or choosing a threshold quantization. This paper has selected the previous processing mode.

(3) The signal reconstruction

After high-frequency coefficient processing of step (2), the remained spectrum is real signal(the fundamental component is 0-50Hz low-frequency signal). Then making use of the reconstruction algorithm of wavelets transform reconstructs original signals, obtaining the signals of eliminating noises from signals. The reconstruction equation is as follows:

$$C_k^j = \sum_k \left[p_{k-2i} c_l^{j-1} + q_{k-2i} d_l^{j-1} \right] \qquad (2)$$

The reconstruction sketch as Fig. 2:

This paper select wavelets Sym8, decomposition level $M = 6$.

The voltage and current of removing noises from signals are 0-50Hz useful signals, but other high-frequency harmonic wave or noise component is very minimum or zero, then:

$$U_l(t) = A \sin(2\pi f t + \theta), \quad I_l(t) = B \sin(2\pi f t + \theta + \delta)$$

$f=49-51\text{Hz}$; $U_l(t)$ and $I_l(t)$ are fundamental wave voltage and current.

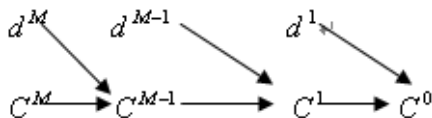


Fig. 2. The reconstruction sketch

If k as the sampling number of the signals is in full period, then the equation of correlation function $R_{UI}(\tau)$ of $U_l(t)$ and $I_l(t)$ when $\tau = 0$ is:

$$\hat{R}_{UI}(0) = \frac{1}{k} \sum_{n=0}^{k-1} U_l(n)I_l(n) = \frac{AB}{2} \cos\delta \tag{3}$$

When $\tau = 0$, the equation of correlation function $R_u(\tau)$ of $U_l(t)$ and $I_l(t)$ is:

$$\hat{R}_U(0) = \frac{1}{k} \sum_{n=0}^{k-1} U_l^2(n) = \frac{A^2}{2} \tag{4}$$

$$\hat{R}_I(0) = \frac{1}{k} \sum_{n=0}^{k-1} I_l^2(n) = \frac{B^2}{2} \tag{5}$$

Thus we can get:

$$\tan\delta = \sqrt{\frac{R_U(0) \bullet R_I(0)}{\hat{R}_I(0)}} - 1 \tag{6}$$

$$\delta = \arccos(2\hat{R}_{UI}(0) / \sqrt{2R_U(0)}\sqrt{2R_I(0)}) \tag{7}$$

3 System Structure

On-line measurement system for dielectric loss of virtual capacitive type equipment applies the construction of virtual instrument(Fig. 3). Monitoring center sends instructions to every monitoring node by using GPRS network, monitoring nodes returning sampling results according to prescriptive data protocol, calculating and analyzing them on monitoring center computers.

4 Software Structure

Whole on-line measurement system mainly includes: general hardware, software and system's main body software module. The former is solidified in general hardware. The latter is a host software for realizing whole functions of the system, applying LabView , by using them you could easily design virtual instrument panels. And there are powerful library functions of data analysis, apparatus drives and serial communications and so on. Relevant testing programs are written in terms of the principle mentioned: wavelets transform, by using correlativity of acquisition voltage and acquisition current to obtain phase shift and SPC.

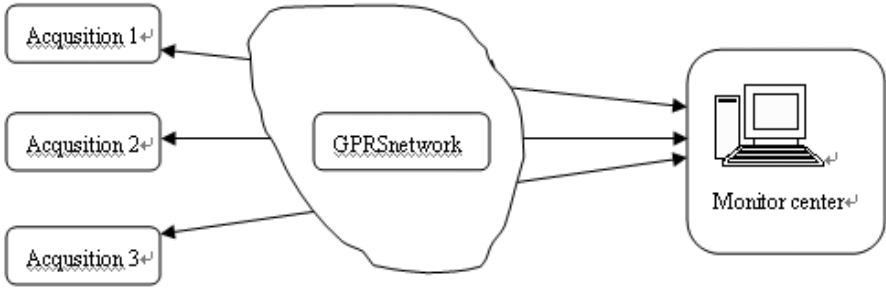


Fig. 3. The monitoring system

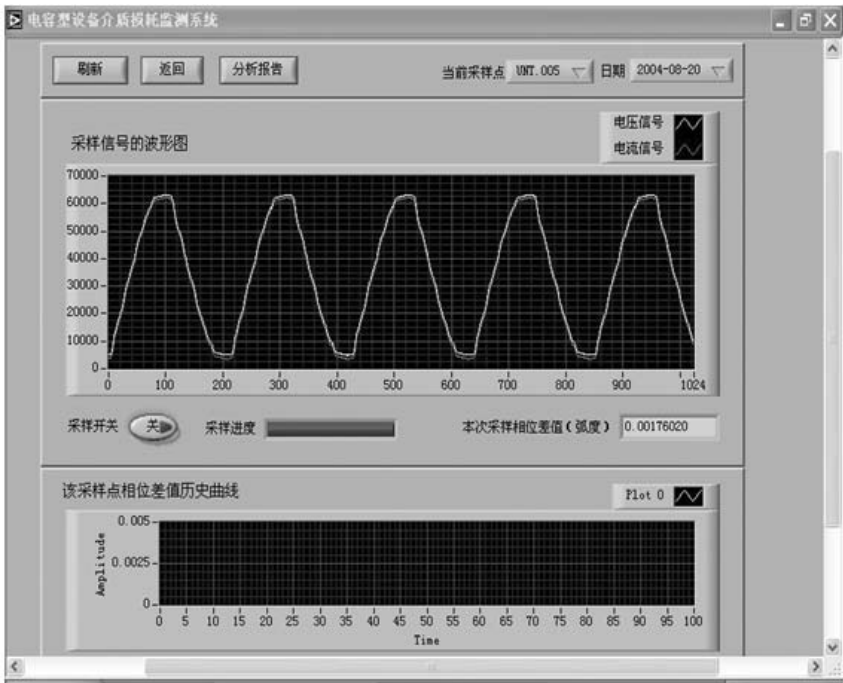


Fig. 4. The virtual instrument panel

5 Testing Result

Laboratory identifications have been made. The pictures Fig. 4 and Fig. 5 are the virtual instrument panels displayed after system having run.

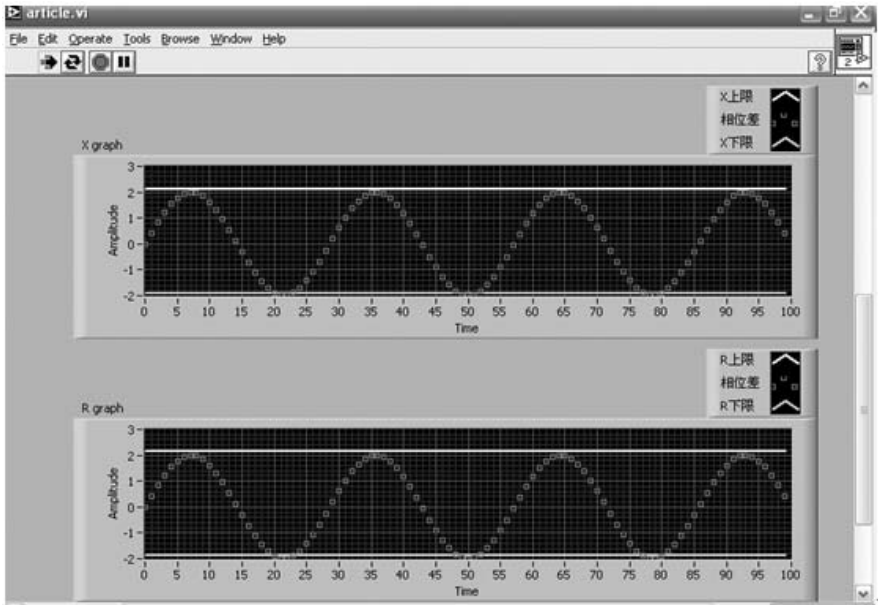


Fig. 5. The SPC system panel

6 Conclusions

- adopting the method of combining wavelets noise elimination and relevant measurement methods to measuring $\tan\delta$, improving the accuracy and stabilization greatly.
- virtual apparatus improves the degree of intelligence and testing performance, and it is very favorable to electric device and network management.
- the design of real-time data acquisition module resolve the key problem of applying virtual instrument to test $\tan\delta$. It works independently, accomplishing high speed acquisition of dual-way signal, synchronously acquisition task. The acquisition data is deposited in inner automatically. The structure is simple, convenient for operation on the spot.

References

- Anderson W. E., Davis R. S.(National Bureau of Standards, Washington, DC.): Measurements on Insulating Materials at Cryogenic Temperatures. Sponsor: Brookhaven National Lab., Upton, NY. Report: NBSIR-79-1950 (1979) 165
- Bai F., Hang Xin-hong (Xi'an Jiaotong Univ), Yan Z.: On-line measuring dielectric loss angle of capacitive-type insulation by using digital measurement technique. Source: International Symposium on Electrical Insulating Materials Proceedings (1995) 275-278 (CODEN: 003175)

3. Bromley J.C, McDer mid W., Raghuv eer M.R., Wang P. (University of Manitoba): A digital technique for the on_line measurement of dissipation factor and capacitance. Source: IEEE Transactions on Dielectrics and Electrical Insulation, v 8, n 2, April, 2001, p 228-232, ISSN: 1070-9878 CODEN: ITDIES , Publisher: Institute of Electrical and Electronics Engineers Inc.
4. <http://digital.ni.com/worldwide/china.nsf/main?readform>, LabVIEW user book
5. Nie Peng, Pu Zhaobang, Sun Heyi (Harbin Inst. of Technol.): New interference reduction method in dielectric loss tangent δ , on_line monitoring for capacitive equipment. Source: Dianli Xitong Zidonghua/Automation of Electric Power Systems, v 28, n 3, Feb 10, 2004, p 67-70 Language: Chinese

Model Checking Temporal Logics of Knowledge and Its Application in Security Verification

Lijun Wu¹, Kaile Su^{1,2}, and Qingliang Chen¹

¹ Department of Computer Science and Technology,
Sun Yat-sen University, Guangzhou 510275, China

² Institute of Electronics and Information Engineering,
He'nan University of Science and Technology, Luoyang 471003, China
gzsudocwlj@yahoo.com

Abstract. Model checking has been used mainly to check if a system satisfies the specifications expressed in temporal logic and people pay little attention to the problem for model checking logics of knowledge. However, in the distributed systems community, the desirable specifications of systems and protocols have been expressed widely in logics of knowledge. In this paper, based on the SMV, by the semantics of knowledge and set theory, approaches for model checking logics of knowledge and common knowledge are presented. These approaches make SMV's functions extended from temporal logics to temporal logics of knowledge. We will illustrate in an example the applications to security verifications for a cryptographic protocol.

1 Introduction

Model checking is an automatic technique for verifying finite state concurrent systems and is mainly used to verify the specification expressed in temporal logics such as CTL (Computation Tree Logic) in the case of SMV [1] and LTL (Linear Temporal Logic) in the case of SPIN and FORSPEC [2]. Comparatively little attention in the model checking community has been given to epistemic logics. However, the temporal logics of knowledge have widely been applied in the distributed systems and the specifications of protocols. And so model checking the temporal logic of knowledge is a new important research domain. Ron van der Meyden and N.V.Shilov discussed the problem of model checking knowledge and time in systems with perfect recall [3], but the complexity is too high. Wiebe van der Hoek presented the algorithm based on the logic of local propositions, which can model check the temporal logic of knowledge. But it was limited to the logics CKL_n that combine LTL and knowledge components and the theoretical foundations of their work still need to be further investigated [4].

In the paper, we give approaches for model checking the logic that combines CTL and knowledge components. Based on the SMV, by the set theory and the semantics of knowledge, we further verify the specifications in knowledge and common knowledge and discuss the complexity. So we make functions of SMV be expanded from model checking CTL to model checking the temporal logic of knowledge. We will show an example the applications to security verification.

2 SMV and Temporal Logics of Knowledge

SMV is an efficient model checker and can verify the specification expressed in CTL [1]. But it cannot verify the specifications such as $K_i\varphi$ and $K_jK_i\varphi$ and $C_G\varphi$. In this paper, based on the set $T(T = \{s|s \in S, (M, s) \models \varphi\})$ of states that SMV outputs finally, we further verify knowledge and common knowledge, and thus can model check specifications $K_i\varphi$ and $K_jK_i\varphi$ and $C_G\varphi$.

The temporal logics of knowledge are the logics that combine temporal and epistemic logics. Researchers have developed the temporal logics of knowledge on the basis of LTL, for example, Wiebe van der Hoek has verified the logics that combine LTL and epistemic logics (Halpern and Vardi's logic CKL_n [5]).

In the paper, we mainly focus on the logic CKK_n that combines CTL and epistemic logic. By the semantics of knowledge and set theory, based on the output of SMV, we can further model check knowledge and common knowledge. The formulas of CKK_n are composed of proposition operators, path quantifiers, temporal operators and knowledge operators. So the set of formulas of CKK_n can be defined as follows:

$$\langle CKK_n \rangle ::= p | \neg \langle CKK_n \rangle | \langle CKK_n \rangle \vee \langle CKK_n \rangle | EX \langle CKK_n \rangle | EG \langle CKK_n \rangle | E[\langle CKK_n \rangle U \langle CKK_n \rangle] | K_i \langle CKK_n \rangle | C_G \langle CKK_n \rangle$$

Because SMV mainly model checks the formulas in EX , EG and EU , we pay main attention to the model checking specifications such as $K_i\varphi$ and $C_G\varphi$ (here, φ is a formula of CTL).

3 The Algorithm for Model Checking the Temporal Logics of Knowledge Based on SMV

We assume that Kripke structure $M = (S, R, L, K_1, \dots, K_n)$, here S is a set of states, R is a transition relation that must be total (that is, for every state $s \in S$, there is a state $s' \in S$ such that $R(s, s')$), L is a map $L : S \rightarrow 2^{AP}$ (AP is an atomic proposition set), and K_i is an equivalence relation of M concerned with agent i . According to semantics of knowledge and common knowledge in [6], we can define semantics of knowledge and common knowledge in M as follows: $(M, s) \models K_i\varphi$ iff $(M, t) \models \varphi$ for all t such that $(s, t) \in K_i$. $(M, s) \models C_G\varphi$ iff $(M, s) \models E_G^k\varphi$, for $k = 1, 2, \dots$; for $\forall s \in S$ and for agent i , we define the direct set of agent i at state s as $T(s, i) = \{t | t \in S, (s, t) \in K_i\}$. Let φ be a CTL formula, we denote the output the state set of SMV to be $T = \{t | t \in S, (M, t) \models \varphi\}$.

Proposition 1. *If $s' \in T(s, i)$ and $K_i (i = 1, \dots, n)$ is an equivalence relation, then $T(s', i) = T(s, i)$.*

Proof: Let $t \in T(s', i)$, then $(s', t) \in K_i$. Since $s' \in T(s, i)$, it follows that $(s, s') \in K_i$. And since K_i is an equivalence relation, it follows $(s, t) \in K_i$, and so $t \in T(s, i)$. Hence $T(s', i) \subseteq T(s, i)$. By the same reason, we can prove that $T(s, i) \subseteq T(s', i)$. Therefore $T(s', i) = T(s, i)$. ■

3.1 Model Checking $K_i\varphi$

Let $T_{K_i} = \{s | s \in S, (M, s) \models K_i\varphi\}$, we have following propositions for T_{K_i} and T .

Proposition 2. *If $K_i(i = 1, \dots, n)$ is an equivalence relation and φ is a formula of CTL, then $T_{K_i} \subseteq T$; If $s \in T$, then $T(s, i) \subseteq T_{K_i}$ if and only if $T(s, i) \subseteq T$.*

Proof: Let $s \in T_{k_i}$, then $(M, s) \models K_i\varphi$. By the semantics of knowledge, it follows that $(M, s') \models \varphi$ for any s' such that $(s, s') \in K_i$. Since K_i is an equivalence relation, $(s, s) \in K_i$. Hence $(M, s) \models \varphi$, and so $s \in T$, therefore $T_{k_i} \subseteq T$.

Assume $T(s, i) \subseteq T$, because $T(s, i) = \{t | t \in S, (s, t) \in K_i\}$ and $T(s, i) \subseteq T$, it follows that $(M, t) \models \varphi$ for all t such that $(s, t) \in K_i$. From semantics of knowledge, it follows that $(M, s) \models K_i\varphi$, and hence $s \in T_{k_i}$. Now we prove $T(s, i) \subseteq T_{k_i}$. Let $s' \in T(s, i)$, from Proposition 1, it follows that $T(s', i) = T(s, i)$, and so $T(s', i) \subseteq T$. From the above proof, we must have $s' \in T_{k_i}$, and thus $T(s, i) \subseteq T_{k_i}$. For the other direction, assume $T(s, i) \subseteq T_{k_i}$ and let $t \in T(s, i)$. It follows that $(M, s) \models K_i\varphi$. So by $t \in T(s, i)$, we must have $(M, t) \models \varphi$, that is, $t \in T$, and hence $T(s, i) \subseteq T$. ■

Therefore, in order to model check $K_i\varphi$ (that is to calculate T_{K_i}), we only need to traverse all states of T . At every step, we take out a state s from T , if $T(s, i) \subseteq T$, then we put $T(s, i)$ into the result set T_{K_i} (the initial value of T_{K_i} is null) and delete $T(s, i)$ from T to reduce T , otherwise, we do not put $T(s, i)$ into the result set T_{K_i} but delete $T(s, i) \cap T$ from T to reduce T . And then we take out another state from T and repeat the above process until finishing traversing all states of T and T becomes null. The resulting set T_{K_i} that we obtain finally is just the set of states that satisfy specification $K_i\varphi$ in the system. In the algorithm, T is the set of states that satisfy specification φ and is obtained by model checker SMV.

3.2 Model Checking $K_jK_i\varphi$

Assume $T_{K_{ji}} = \{s | s \in S, (M, s) \models K_jK_i\varphi\}$. Then model checking $K_jK_i\varphi$ is to calculate $T_{K_{ji}}$.

Proposition 3. *If $K_i(i = 1, \dots, n)$ is an equivalence relation and φ is a formula of CTL, then $T_{K_{ji}} \subseteq T_{K_i}$; If $s \in T_{K_i}$, then $T(s, j) \subseteq T_{K_{ji}}$ iff $T(s, j) \subseteq T_{K_i}$.*

Proof: Assume $s \in T_{k_{ji}}$, it follows $(M, s) \models K_jK_i\varphi$. By the definition of knowledge, we must have $(M, s') \models K_i\varphi$ for all s' such that $(s, s') \in K_j$. Since K_j is an equivalence relation, it follows $(s, s) \in K_j$ and $(M, s) \models K_i\varphi$. And so $s \in T_{k_i}$, therefore $T_{k_{ji}} \subseteq T_{k_i}$. The latter part can be proved in a similar way to the corresponding part in proposition 2. ■

Therefore, in order to model check $K_jK_i\varphi$ (that is to calculate $T_{K_{ji}}$), we can first calculate T_{K_i} and then traverse all states of T_{K_i} . At every step, we take out a state s from T_{K_i} , if $T(s, j) \subseteq T_{K_i}$, then we put $T(s, j)$ into the resulting set $T_{K_{ji}}$ (the initial value of $T_{k_{ji}}$ is null) and delete $T(s, j)$ from T_{K_i} to reduce

T_{K_i} , otherwise, we do not put $T(s, j)$ into the resulting set $T_{K_{j_i}}$ but delete $T(s, j) \cap T_{K_i}$ from T_{K_i} to reduce T_{K_i} . And then we take out another state from T_{K_i} and repeat the above process until finishing traversing all states of T_{K_i} and T_{K_i} becomes null. The resulting set $T_{K_{j_i}}$ that we obtain finally is just the set of states that satisfy specification $K_j K_i \varphi$ in the system.

3.3 Model Checking $C_G \varphi$

Assume G is a group of agents of agent 1, ..., agent n . $C_G \varphi$ expresses the common knowledge of agents in G . And let $T_c = \{s | s \in S, (M, s) \models C_G \varphi\}$, and $T_k = \{s | s \in S, (M, s) \models E_G^k \varphi\} (k = 1, 2, \dots)$. Proofs for proposition 4 and 5 are using similar approaches as in proposition 2 and 3 and so are omitted here.

Proposition 4. *If $K_i (i = 1, \dots, n)$ is an equivalence relation and φ is a formula of CTL, then $T_c \subseteq T$ and $T_{k+1} \subseteq T_k (k = 1, 2, \dots)$.*

Proposition 5. *If $K_i (i = 1, \dots, n)$ is an equivalence relation and $s \in T_k$, then $s \in T_{k+1}$ iff $T(s, i) \subseteq T_k (i = 1, 2, \dots, n)$.*

Proposition 6. *There is $m_0 (1 \leq m_0 \leq |S|)$ such that $T_j = T_{m_0}$ for any $j \geq m_0$ ($|S|$ is the number of states in S).*

Proof: Let $|S| = m$. By proposition 4, we have that $\{T_k\}$ is a degressive sequence of states. Since $|T_k| \leq m$ for any natural number k and $\{T_K\}$ is a degressive sequence of states, it is clear that there must be a m_0 in $\{1, 2, \dots, m, m+1\}$ such that $T_{m_0+1} = T_{m_0}$. Now we prove that for any natural number k , if $T_{k+1} = T_k$, then $T_{k+2} = T_{k+1}$. Assume $T_{k+1} = T_k$ and $s \in T_{k+1}$, then it follows $(M, t) \models E_G^k \varphi$ for all t such that $t \in T(s, i) (i = 1, 2, \dots, m)$. Hence $t \in T_k$, and so $t \in T_{k+1}$. It follows $(M, t) \models E_G^{k+1} \varphi$, and thus $(M, s) \models K_i E_G^{k+1} \varphi (i = 1, 2, \dots, m)$, therefore $(M, s) \models E_G^{k+2} \varphi$, that is, $s \in T_{k+2}$, and hence $T_{k+1} \subseteq T_{k+2}$. By proposition 4, it follows $T_{k+1} \supseteq T_{k+2}$, and so $T_{k+1} = T_{k+2}$. By the arbitrariness of k and the above process of proof, we have that $T_j = T_{m_0}$ for any $j \geq m_0$. ■

Proposition 7. *There is a natural number m_0 such that $m_0 \leq |S|$ and $T_c = T_{m_0}$.*

Proof: Followed immediately by Proposition 6. ■

In order to model check $C_G \varphi$ (that is to calculate T_c), we can first calculate T_1 and T_2 . If $T_2 = T_1$, then by proposition 6 and proposition 7, it follows that T_1 is just the set T_c of states that satisfy the specification $C_G \varphi$. If $T_2 \neq T_1$, we continue to calculate T_3 and repeat the above process. By proposition 7, it follows that T_c can be calculated at most m_0 steps.

Now we discuss how to calculate T_{k+1} from T_k at every step of above process. By proposition 5, in order to calculate T_{k+1} from T_k , we need only to traverse all states of T_k . At every step, we take out a state s from T_k , if it follows $T(s, i) \subseteq T_k$ for all i such that $i \in \{1, 2, \dots, n\}$, then we put s into the resulting set T_{k+1} (the initial value of T_{k+1} is null) and delete s from T_k , otherwise, we do not put s into the resulting set T_{k+1} but delete s from T_k . And then we take out another state from T_k and repeat the above process until finishing traversing all states of T_k and T_k becomes null. The resulting set T_{k+1} that we get finally is just what we want.

4 Complexity of the Algorithms

Here we consider the complexity of our algorithms. Assume Kripke structure $M = (S, R, L, K_1, \dots, K_n)$.

Proposition 8. *The time complexity of model checking $K_i\varphi$ is $O(|S|^2)$; The time complexity of model checking of $K_jK_i\varphi$ is $O(|S|^2)$. The time complexity of model checking $C_G\varphi$ is $O(m * |S|^3)$. Here $|S|$ is the number of states in S and m is the number of agents in G , that is, they are all polynomial.*

Proof: Here we just present the case for computing $K_i\varphi$, the others are similar. Firstly, to calculate $T(s, i)$ needs at most $|S|$ steps. At every step during traversing T , to decide $T(s, i) \subseteq T$ needs at most $|T(s, i)| * |S|$ steps and to delete $T(s, i)$ from T needs also at most $|T(s, i)| * |S|$ steps. Assume that $T(s, i)$ is expressed in $S(j)$ at j th step and T becomes null after k steps (the algorithm ends), then it follows that $T = \cup_{j=1}^k S(j)$ and $S(i) \cap S(j) = \emptyset$ for any $i, j (1 \leq i, j \leq k)$. Hence, to finish traversing T needs $\sum_{j=1}^k (|S| + 2 * |S(j)| * |S|)$ steps. And it is clear that $\sum_{j=1}^k (|S| + 2 * |S(j)| * |S|) \leq k * |S| + 2 * |S|^2 \leq 3 * |S|^2$, that is, polynomial. ■

5 Applications to Security Verifications

5.1 TMN Cryptographic Protocol

We use TMN cryptographic protocol [7] as an applications of our algorithms to security verifications. TMN cryptographic protocol is a key distribution protocol for mobile communication systems. The protocol is as follows:

$$\begin{aligned} A &\rightarrow S : B, \{N_a\}_{K_S} \\ S &\rightarrow B : A \\ B &\rightarrow S : A, \{N_b\}_{K_S} \\ S &\rightarrow A : B, \{N_b\}_{N_a} \end{aligned}$$

Here, A is the initiator, B is the responder, S is a trusted server, K_S is the public key of S , N_a is the nonce that A issues, N_b is the nonce that B issues, N_b is also the session key between A and B . The purpose of the protocol is to build a session key between A and B , which is used later when they communicate.

5.2 The Specification of the System

In the protocol, the initiator A and the responder B attempt, with the aid of the trusted central server S , to obtain a shared secret for use as a session key. So system must satisfy two security properties. Firstly, after the protocol runs, A and B should know the session key (expressed with K_{ab}) each other, that is, "the session key between A and B is K_{ab} " is common knowledge between A and B . Secondly, the intruder must not know "the session key between A and B is K_{ab} ". Assume we express "the session key between A and B is K_{ab} " in φ , express " A knows φ " in $K_A\varphi$, express " B knows φ " in $K_B\varphi$, express

" I knows φ " in $K_I\varphi$, and express "the common knowledge between A and B " in $C_G\varphi$ (here $G = \{A, B\}$). Thus the specification of system can be expressed as $C_G\varphi \wedge \neg K_I\varphi$. With the aid of SMV and the program of our algorithm, we have verified that the system (TMN cryptographic protocol) does not satisfy the specification $C_G\varphi \wedge \neg K_I\varphi$.

6 Conclusion

In the paper, based on the SMV, according to the semantics of knowledge and set theory, we present algorithms with polynomial time complexity and make the model checking tool of SMV be expanded from CTL to the temporal logics of knowledge with one and two level nested knowledge and common knowledge. We go a further step than this kind of work in [4]. Our future work will focus on how to apply the method to checking more epistemic specifications for rational agents such as BDI model [8].

Acknowledgement. This work is supported by Science Foundation of China grants 60496327, 10410638 and 60473004, German Research Foundation grant 446 CHV113/240/0-1, and Guangdong Provincial Natural Science Foundation grant 04205407.

References

1. O.Grumberg E.M. Clarke and D.A. Peled. *Model Checking*. The MIT Press, Cambridge, MA, 2000.
2. M.Y. Vardi. Branching vs. linear time: final showdown. In T. Margaria and W.Yi, editors, *Proc. 7th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2001)*, volume LNCS No. 2031, pages 1–22, Berlin, 2001. Springer.
3. R. van der Meyden and N.S. Shilov. Model checking knowledge and time in systems with perfect recall. In *Proc. Conf. on Foundations of Software Technology and Theoretical Computer Science*, Springer LNCS No 1738, pages 262–273, Berlin, 1999.
4. W.vander Hoek and M.Wooldridge. Model checking knowledge and time. In *Proc. 19th Workshop on SPIN(Model Checking Software)*, Grenoble, April,2002.
5. J. Halpern and M. Vardi. The complexity of reasoning about knowledge and time: extended abstract. In *Proc. 18th Annual ACM Symposium on Theory of Computing*, pages 304–315, 1986.
6. R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about knowledge*. MIT Press, Cambridge, MA, 1995.
7. Makoto Tatebayashi, Natsume Matsuzaki, and David B. Newman Jr. Key distribution protocol for digital mobile communication systems. In *Proceedings of Advances in Cryptology - CRYPTO '89*, pages 324–334.
8. Kaile Su, Abdul Sattar, Kewen Wang, Xiangyu Luo, Guido Governatori, and Vineet Padmanabhan. Observation-based model for BDI-agents. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-05)*, 2005.

A Computational Approach for Belief Change

Shangmin Luan and Guozhong Dai

Institute of Software, Chinese Academy of Sciences, China
shangmin@iscas.cn

Abstract. In this paper, we combine the syntax-based belief change approach and model-based approach, and present a computational approach for belief change. We introduce functions to revise or contract a belief set, as well as functions to revise or contract a belief base. We also show properties of the revision functions and the contraction functions. The implementation of the revision functions and the contraction functions are also considered, and algorithms to revise a belief set or contract a sentence from a belief set are also given. Compared with related works, the main characteristic of our approach is that the functions can be implemented by algorithms.

1 Introduction

Any intelligent agent has to account for a changing environment and the fact that its own beliefs might be inaccurate. For this reason, belief change is a task central for any kind of intelligent behavior[1, 2], and has been extensively studied during the last twenty years. The most influential work is given by Alchourron, Gärdenfors and Makinson[3]. Belief states are modeled by logically closed sets of sentences, called belief sets. Rational postulates and belief change functions are proposed[3, 4]. A belief set is difficult to be represented by the existing knowledge representation methods given so far, so finite sets are particularly interesting, and this has led to a number of different approaches using sets of sentences which are not closed under logical consequence[5, 6]. Another important method for revision of a propositional knowledge base is model-based approach [7, 8]. Dixon[9] give an implementation of the most conservative entrenchment by assigning each formula a natural number as its rank. Williams[10] provides a computational model for belief base revision. Delgrande[11, 12] presents a general, consistency-based framework for expressing belief change. Other related works are given in [13, 14]. The former works introduced focus on a single revision of a belief set by a formula, however, one would be interested in sequences of revisions, called iterated belief revision[15, 16].

In this paper, we combine the model-based and syntax-based approach, and introduce an approach for belief change. The rest of the paper is organized as follows. In section 2, we define the contraction and revision functions for belief sets and belief bases, show their properties. In section 3, the implementation of our approach is considered. In section 4, related works are compared with our results, and conclusion is also given. For notations and terminologies, please refer to [17].

2 A Computational Approach for Belief Base Change

This section introduces our approach to belief change. Revision and contraction of logically closed set of sentences(called belief set change) are considered, as well as revision and contraction of a belief base.

2.1 A Computational Approach for Belief Change

Definition 1. (*Revision of a Belief Set*) Suppose Γ is a belief set and A is a sentence, Δ is the base of Γ , V_1 is a truth assignment satisfying Δ , V_2 is a truth assignment satisfying A . We construct a truth assignment V as follows: For an atom L , if $L \in Var(A) \cap Var(\Delta)$, $V(L) = V_2(L)$; if $L \in Var(A) - Var(\Delta)$, $V(L) = V_2(L)$; if $L \in Var(\Delta) - Var(A)$, $V(L) = V_1(L)$. Then, we define $\Gamma + A = Cn((\Delta|V) \cup \{A\})$.

Theorem 1. *The revision function $+$ defined above satisfies the AGM's postulates $(+1)$, $(+2)$, $(+4)$.*

Proof. By the definition, $\Gamma + A$ is a closed theory set. Hence, the revision function given above satisfies $(+1)$. We also know $A \in \Gamma + A$ by the definition, so the revision function satisfies $(+2)$. If $\neg A \notin Cn(\emptyset)$, i.e., A is not an axiom, then we know V is a model satisfying $\Gamma + A$ by the definition of V . $\Gamma + A$ is consistent. Hence, the revision function satisfied $(+4)$.

Example 1. The above revision function $+$ does not satisfy AGM's postulate $(+3)$. Suppose, $\Gamma = Cn(\Delta = \{\neg A \vee B, \neg A \vee C, \neg A \vee D\})$, $A \vee B$ is a sentence. $\neg(A \vee B) = \neg A \wedge \neg B \notin Cn(\Gamma)$. The truth assignment $V_1(A) = 0, V_1(B) = 0, V_1(C) = 0, V_1(D) = 0$ satisfies Δ , and the truth assignment $V_2(A) = 1$ satisfies $A \vee B$. By the definition 1, $V(A) = 1, V(B) = 0, V(C) = 0, V(D) = 0$. $\Gamma + (A \vee B) = Cn(\{A \vee B\}) \neq Cn(\Gamma \cup \{A \vee B\})$. But there exists a truth assignment $V_1(A) = 1, V_1(B) = 1, V_1(C) = 1, V_1(D) = 1$ satisfying Δ and a truth assignment $V_2(A) = 1$ satisfying $A \vee B$ such that the resulted truth assignment $V(A) = 1, V_1(B) = 1, V_1(C) = 1, V_1(D) = 1$ satisfies $\Gamma + (A \vee B) = Cn((\Delta|V) \cup \{A \vee B\}) = Cn(\Gamma \cup \{A \vee B\})$. For the general case, we have the following theorem.

Theorem 2. *If $\neg A \notin Cn(\Gamma)$, then there exists a model V such that $\Gamma + A = Cn((\Delta|V) \cup \{A\}) = Cn(\Gamma \cup \{A\})$, i.e., the revision function satisfies the postulates $(+3)$.*

Proof. If $\neg A \notin Cn(\Gamma)$, then A is consistent with Δ which is the base for Γ . So, there exists at least one model V such that $V \models \Delta$ and $V \models A$. Hence, $\Gamma + A = Cn(\Delta|V \cup \{A\}) = Cn(\Delta \cup \{A\})$, i.e., the revision function satisfies the postulates $(+3)$.

Example 2. The above revision function $+$ does not satisfy AGM's postulate $(+5)$. Suppose, $\Gamma = Cn(\Delta = \{\neg A \vee B, \neg A \vee C, \neg A \vee D\})$. $Cn(A \vee B) = Cn(\neg A \rightarrow B)$. The truth assignment $V_1(A) = 0, V_1(B) = 0, V_1(C) = 0, V_1(D) = 0$ satisfies Δ . The truth assignment $V_2(A) = 1$ satisfies $A \vee B$. And the truth assignment $V_3(B) = 1$ satisfies $A \vee B$. This results in two truth assignment,

$V_4(A) = 1, V_4(B) = 0, V_4(C) = 0, V_4(D) = 0$ and $V_5(A) = 0, V_5(B) = 1, V_5(C) = 0, V_5(D) = 0$. $\Gamma + (A \vee B) = Cn(\Delta|V_4 \cup \{A \vee B\}) \neq Cn(\Delta|V_5 \cup \{A \vee B\}) = \Gamma + (\neg A \rightarrow B)$. In fact, V_2 is also a model satisfying $\neg A \rightarrow B$, and V_3 is also a model satisfying $A \vee B$. If we use one of the two models to construct the model V defines in the definition 1, then $\Gamma + (A \vee B) = Cn((\Delta|V) \cup \{A \vee B\}) = Cn(\Delta|V \cup \{\neg A \rightarrow B\}) = \Gamma + (\neg A \rightarrow B)$. Similarly, $+$ does not meet $(\dagger 6)$. For the general case, we have the following theorem.

Theorem 3. *If $Cn(A) = Cn(B)$, then there exists a model V such that $\Gamma + A = Cn((\Delta|V) \cup \{A\}) = \Gamma + B = Cn((\Delta|V) \cup \{B\})$, i.e., the revision function satisfies the postulate $(\dagger 5)$.*

Proof. If $Cn(A) = Cn(B)$, then a model for A is also a model for B , and vice versus, i.e., a model for B is also a model for A . So, if the models used in constructing $\Gamma + A$ are the same as the models used in constructing $\Gamma + B$, then $\Gamma + A = \Gamma + B$, i.e., the revision function satisfies the postulate $(\dagger 5)$.

Theorem 4. *If Γ is closed, then there exists a model V such that $(\Gamma + A) \cap \Gamma = Cn(\Delta|V) = \Gamma - \neg A$, i.e., the revision function satisfies the postulate $(\dagger 6)$.*

Proof. If the models used in constructing $\Gamma + A$ are the same as the models used in constructing $\Gamma - \neg A$, then $(\Gamma + A) \cap A = \Gamma - \neg A = Cn(\Delta|V)$, i.e., the revision function satisfies the postulate $(\dagger 6)$.

Definition 2. *(Contraction of a Belief Set) Suppose Γ is a belief set and A is a sentence, Δ is the base of Δ , V_1 is a truth assignment satisfying Δ , V_2 is a truth assignment satisfying $\neg A$. We construct a truth assignment V as follows: For an atom L , if $L \in Var(A) \cap Var(\Delta)$, $V(L) = V_2(L)$; if $L \in Var(A) - Var(\Delta)$, $V(L) = V_2(L)$; if $L \in Var(\Delta) - Var(A)$, $V(L) = V_1(L)$. Then, we define $\Gamma - A = Cn(\Delta|V)$.*

For the contraction function, we have similar theorems. For the limitation on length of the paper, these theorems are omitted. Similarly, operators revising or contracting a belief base are defined as follows:

Definition 3. *(Revision of a Belief Base) Suppose Γ is a belief base and A is a sentence, V_1 is a truth assignment satisfying Γ , V_2 is a truth assignment satisfying A . We construct a truth assignment V as follows: For an atom L , if $L \in Var(A) \cap Var(\Gamma)$, $V(L) = V_2(L)$; if $L \in Var(A) - Var(\Gamma)$, $V(L) = V_2(L)$; if $L \in Var(\Gamma) - Var(A)$, $V(L) = V_1(L)$. Then, we define $\Gamma \dot{+} A = (\Delta|V) \cup \{A\}$.*

Definition 4. *(Contraction of a Belief Base) Suppose Γ is a belief set and A is a sentence, V_1 is a truth assignment satisfying Γ , V_2 is a truth assignment satisfying $\neg A$. We construct a truth assignment V as follows: For an atom L , if $L \in Var(A) \cap Var(\Gamma)$, $V(L) = V_2(L)$; if $L \in Var(A) - Var(\Gamma)$, $V(L) = V_2(L)$; if $L \in Var(\Gamma) - Var(A)$, $V(L) = V_1(L)$. Then, we define $\Gamma \dot{-} A = \Delta|V$.*

The properties of the functions for revising or contracting a belief base are similar to that of the functions for revising or contracting a belief set. For the limitation on length of the paper, we omit these theorems.

3 Implementation of Belief Change Functions

We use triple $\langle \Gamma, \Delta, A \rangle$ to denote a scenario, where Γ is a belief set, Δ is its base. Γ is revised by A . For a general propositional sentence, it is difficult to find a truth assignment satisfying the sentence. But many algorithms for finding a model satisfying a formula have been presented. It is also well known that a general sentence can be transformed into a set of clauses. So, we can first transform a general sentence into a set of clauses, and then find a model satisfying the clause set. The resulted model is also a model satisfying the general sentence.

It is also well known that equivalent sentences have the same standard CNF form. So, we first transform a sentence into its standard CNF form, and use $\text{Transform_into_CNF}(A)$ to denote the algorithm. And then an existed algorithm is used to find a model of a set of clauses, and $\text{Find_Model}(\mathcal{C})$ is used to denote the algorithm, where \mathcal{C} is a set of clauses. \mathcal{C}_1 and \mathcal{C}_2 are used to denote the standard CNF of general propositional sentence C_1 and C_2 , respectively.

We also need an algorithm to find a model satisfying $\mathcal{C}_1 \cup \mathcal{C}_2$. The resulted model satisfies \mathcal{C}_2 , the sentences in \mathcal{C}_1 is maximal. It is easy to give such an algorithm, and $\text{Find_Max}(\Delta, A)$ is used to denote the algorithm. An revision algorithm is given as follows.

```

Revision( $\langle \Gamma, \Delta, A \rangle$ )
   $\mathcal{C}_1 = \text{Transform\_into\_CNF}(\Delta)$ ;
   $\mathcal{C}_2 = \text{Transform\_into\_CNF}(A)$ ;
   $V = \text{Find\_Max}(\mathcal{C}_1, \mathcal{C}_2)$ ;
   $\Gamma = \text{Cn}((\Delta|V) \cup \{A\})$ ;
   $\Delta = \Delta|V \cup \{A\}$ ;
End(Revision)

```

Theorem 5. *The revision function defined by the above procedure satisfies (+1) – (+6).*

Proof. By theorem 1, we only show that the revision function defined by the above procedure satisfies (+3), (+5) and (+6). For $\mathcal{C}_1 = \mathcal{C}_3$ and $\mathcal{C}_2 = \mathcal{C}_4$, $V_1 = \text{Find_Max}(\mathcal{C}_1, \mathcal{C}_2)$ and $V_2 = \text{Find_Max}(\mathcal{C}_3, \mathcal{C}_4)$, then $V_1 = V_2$. Hence, we know that the revision function satisfies (+5) and (+6). If $\mathcal{C}_1 \cup \mathcal{C}_2$ is satisfied, $V_1 = \text{Find_Max}(\mathcal{C}_1, \mathcal{C}_2)$, then $V_1 \models \mathcal{C}_1$ and $V_1 \models \mathcal{C}_2$, hence, the revision function satisfies (+3).

The contraction procedure is similar to the revision procedure.

```

Contraction( $\langle \Gamma, \Delta, A \rangle$ )
   $\mathcal{C}_1 = \text{Transform\_into\_CNF}(\Delta)$ ;
   $\mathcal{C}_2 = \text{Transform\_into\_CNF}(\neg A)$ ;
   $V = \text{Find\_Max}(\mathcal{C}_1, \mathcal{C}_2)$ ;
   $\Gamma = \text{Cn}(\Delta|V)$ ;
   $\Delta = \Delta|V$ ;
End(Contraction)

```

Theorem 6. *The contraction function defined by the above procedure satisfies $(\dot{-}1) - (\dot{-}6)$.*

The proof of this theorem is similar to that of theorem 5.

4 Related Works and Conclusion

Previous works on implementing belief change can be divided into two groups, essentially consisting of implementations of non-base revision and of base revision. The former group typically have good formal properties (for example, the AGM postulates) but with inefficient implementations, while the latter group may violate some postulate while being expected to perform reasonably well.

Dixon[9] presented a prototype for belief revision. He discussed the implementation of the entrenchment approach. He first supposed that theory base was finite, then he presented a method to determine the degree of a belief as following: the degree of evidence for a belief is represented as a natural number known as rank of the belief. The higher the rank, the more evidence there is for the belief, and the logical theorems have the highest rank of all. Suppose the logical theorems have rank n , and let the partitions be $\Gamma_1, \Gamma_2, \dots, \Gamma_n$, some of which may be empty. When A is incorporated into Γ , all the beliefs whose ranks are not higher than that of input belief A are removed from (if contradictions occur. Williams[10] improved Dixon's method as following: the rank of a belief remains unchanged, unless there are evidences to change it. But the these revision operators may delete some beliefs which are not related to the inconsistency.

The recent work[11, 12] substitute all the atoms occurrences in a knowledge base. It is not possible for applications to substitute all the atoms occurrences in a knowledge base.

This paper presents a computational approach for belief change by combining the syntax-based belief change approach and model-based approach. We introduce functions to revise or contract a belief set, as well as functions to revise or contract a belief base. We also show properties of the revision and the contraction functions. The implementation of the revision functions and the contraction functions are also considered, and algorithms to revise a belief set or contract a sentence from a belief set are given. Compared with related works, the main characteristic of our approach is that the functions can be implemented by an algorithm.

Acknowledgement. This work is supported by the National Grand Fundamental Research 973 Program of China under Grant No.2002CB312103 and Natural Science Foundation of China under Contact No.60310213 and 60325206.

References

1. H. van Ditmarsch, W. van der Hoek and B. Kooi: Public Announcement and Belief Revision. *Advances in Modal Logic* 2004, University of Manchester, UK (2004) 62-73
2. J.W. Roorda, W. van der Hoek and J.J.Ch Meyer: Iterated Belief Change in Multi Agent Systems, *Logic Journal of the IGPL* 11 (2003) 223-246

3. Alchourron C E, Gardenfors P and Markinson D.: On the Logic of Theory Change Partial Meet Contraction and Revision Functions. *Journal of Symbolic Logic* 50 (1985) 510-530
4. Gardenfors P and Makinson D.: Revisions of Knowledge Systems Using Epistemic Entrenchment. *Proceedings of the Second Conference on Theoretical Aspects of Reasoning About Knowledge, Asilomar, California, U.S.A. (1988)* 83-95
5. B. Nebel.: Syntax based approaches to belief revision. In (P. Gärdenfors ed.), *Belief Revision*, Cambridge University Press (1992) 52-88
6. Doyle J.: Rational Belief Revision, in *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning (1991)* 163-174
7. Dalal M.: Investigations into a Theory of Knowledge Base Revision: Preliminary Report, *Proceedings of the Seventh National Conference on Artificial Intelligence, Saint Paul, Minnesota, U.S.A. (1988)* 475-479
8. Winslett M.: Reasoning about Action Using Possible Models Approach. *Proceedings of the Seventh National Conference on Artificial Intelligence, Saint Paul, Minnesota, U.S.A. (1988)* 89-93
9. Dixon S E.: *Belief Revision: A Computational Approach*. PhD.Thesis, University of Sydney (1994)
10. Williams M A.: Towards a Practical Approach to Belief Revision: Reason-Based Approach. *Proceedings of the Fifth International Joint Conference on Principles of Knowledge Representation and Reasoning, Morgan Kaufmann (1996)* 412-421
11. James Delgrande and Torsten Schaub: A Consistency-Based Approach for Belief Change, *Artificial Intelligence* 151 (2003) 1-41
12. James P. Delgrande, Torsten Schaub, Hans Tompits and Stefan Woltran: On Computing Solutions to Belief Change Scenarios, *Journal of Logic and Computation* 14 (2004) 801-826
13. S. Luan, G. Dai and W. Li.: A Programmable Approach to Maintenance of a Finite Knowledge Base, *Journal of Computer Science and Technology* 18 (2003) 102-108
14. Shangmin Luan, Guozhong Dai: Fast Algorithms for Revision of Some Special Propositional Knowledge Bases, *Journal of Computer Science and Technology* 18 (2003) 388-392
15. Darwiche A and Pearl J.: On the Logic of Iterated Belief Revision. *Artificial Intelligence* 89 (1997) 1-29
16. Boutilier C.: Revision Sequences and Nested Conditionals. *Proceedings International Conference on Artificial Intelligence, Toronto, Ontario, Canada (1993)* 519-525
17. Gallier J H.: *Logic for Computer Science, Foundation of Automatic Theorem Proving*. New York: John Wiley&Sons (1987)

Feature Selection by Fuzzy Inference and Its Application to Spam-Mail Filtering

Jong-Wan Kim¹ and Sin-Jae Kang¹

School of Computer and Information Technology, Daegu University,
Gyeongsan, Gyeongbuk 712-714, South Korea
{jwkim, sjkang}@daegu.ac.kr

Abstract. We present a feature selection method by fuzzy inference and its application to spam-mail filtering in this work. The proposed fuzzy inference method outperforms information gain and chi squared test methods as a feature selection method in terms of error rate. In the case of junk mails, since the mail body has little text information, it provides insufficient hints to distinguish spam mails from legitimate ones. To address this problem, we follow hyperlinks contained in the email body, fetch contents of a remote web page, and extract hints from both original email body and fetched web pages. A two-phase approach is applied to filter spam mails in which definite hint is used first, and then less definite textual information is used. In our experiment, the proposed two-phase method achieved an improvement of recall by 32.4% on the average over the 1st phase or the 2nd phase only works.

1 Introduction

Spam mails, which are unsolicited commercial emails, flood mailboxes, exposing young people to unsuitable content, and wasting network bandwidth. The spam filtering problem can be seen as a particular case of the text categorization problem. Several information retrieval (IR) techniques are well suited for addressing this problem, and in addition it is a two-class problem: spam or non-spam. A variety of machine learning algorithms have been used for email categorization task on different meta-data. Sahami et al. [1] focuses on the more specific problem of filtering spam mails using a Naïve Bayesian classifier and incorporating domain knowledge using manually constructed domain-specific attributes such as phrasal features and various non-textual features. In most cases, support vector machines (SVM) developed by Vapnik outperforms conventional classifiers and therefore has been used for automatic filtering of spam mails as well as for classifying email text [2]. In particular, the best result was obtained when SVM was applied to the header with feature selection. Accordingly, we can conclude that SVM classifier is slightly better in distinguishing the two-class problem.

For selection of important features or terms representing documents such as mails or news well, assigning them weights are the same problem that the existing linear classifiers such as Rocchio and Widrow-Hoff algorithms [3] finding representative vectors of an example document collection. Both of these algorithms use TF (Term Frequency) and IDF (Inverse Document Frequency) for re-weighting terms but they

do not consider term co-occurrence relationship within feedbacked documents. To resolve this drawback, the computation of term co-occurrences between these representative keywords and candidate terms within each example document is required. Three factors of TF, DF (Document Frequency), and IDF have essentially inexact characteristics, which are used to calculate the importance of a specific term. Since fuzzy logic is more adequate to handle intuitive and uncertain knowledge, we combine the three factors by the use of fuzzy inference. We calculate weights of candidate terms by using the method [4] that it is known to give superior performance to the existing representative keyword extraction methods and assign a priority to select representative keywords with the weights of candidate terms.

In this paper, we present the feature selection by fuzzy inference is a little superior to the conventional methods such as information gain and chi square in pornography or porn mails filtering. A two-phase filtering system for intercepting spam mails based on textual information and hyperlinks is also given.

2 Feature Selection in Training Phase

Since the body of a spam mail has little text information recently, it provides insufficient hints to distinguish spam mails from legitimate mails. To resolve this problem, we utilized hyperlinks contained in the email body and extracted all possible hints from original email body and the fetched webpage. These hints are used to construct SVM classifiers. We divided hints into two kinds of information: definite information and less definite textual information. Definite information for filtering spam mails is sender's information, such as email id and URL addresses, and definite spam keyword lists such as "porno," "big money" and "advertisement". There are many particular features of email, which provide evidence as to whether an email is spam or not. For example, the individual words in the text of an email, domain type of the sender, receiving time of an email, or the percentage of non-alphanumeric characters in the subject of an email are indicative of less definite textual information in a spam mail [1]. In the two-phase approach, we first classified the spam mail by using the definite information, and then used the less definite information.

Feature selection from less definite textual information involves searching through all possible combination of features in the candidate feature set to find which subset of features works best for prediction. A few of the mechanisms designed to find the optimum number of features are document frequency threshold, information gain, mutual information, term strength, and chi square. In comparing learning algorithms, Yang and Pedersen found that, except for mutual information, all these feature selection methods had similar performance and similar characteristics [5]. To select features having high discriminating power, we compared the fuzzy inference method [4] with information gain and chi square because information gain and chi square were known effective in text categorization. Information gain is frequently employed as a term goodness criterion in the field of machine learning. It measures the number of bits of information obtained for category prediction by knowing the presence or absence of a term in a document. The chi square measures the lack of independence between a term t and a category c and can be compared to the chi squared distribution with one degree of freedom to judge extremeness. The chi square statistic has a natural value of zero if t and c are independent.

In the fuzzy inference, TF, DF, and IDF of each term are calculated from the pre-processed email documents and are normalized. The normalized term frequency NTF, the normalized document frequency NDF, and the normalized inverse document frequency NIDF are used as fuzzy input variables. Membership functions of the fuzzy input and output variables should have been fuzzified to the form suitable for fuzzy inference. First, we will define the membership functions (μ) of three fuzzy input variables NTF, NDF, and NIDF as the following expressions in which the meaning of each fuzzy term {S, M, L} is corresponding to Small, Middle, and Large, respectively:

$$\begin{aligned} \mu_S(x) &= \max(0, 1 - x/0.75) \text{ and } \mu_L(x) = \max(0, 1+(x-1)/0.75) \text{ for } x = \text{NTF variable,} \\ \mu_S(y) &= \max(0, 1 - y/0.35), \\ \mu_M(y) &= \min\{\max(0, 1+(y-0.5)/0.35), \max(0, 1-(y-0.5)/0.35)\}, \text{ and} \\ \mu_L(y) &= \max(0, 1+(y-1)/0.35) \text{ for } y = \text{NDF variable or NIDF variable.} \end{aligned}$$

Similarly the fuzzy output variable TW (Term Weight) that represents the importance of each term has the following membership functions. At this time, the meaning of each fuzzy term {Z, S, M, L, X, XX} is corresponding to Zero, Small, Middle, Large, Xlarge, XXlarge, respectively.

$$\begin{aligned} \mu_Z(TW) &= \max(0, 1-TW/0.2), \mu_{XX}(TW) = \max(0, 1+(TW-1)/0.2), \\ \mu_S(TW) &= \min\{\max(0, 1+(TW-0.2)/0.2), \max(0, 1-(TW-0.2)/0.2)\}, \\ \mu_M(TW) &= \min\{\max(0, 1+(TW-0.4)/0.2), \max(0, 1-(TW-0.4)/0.2)\}, \\ \mu_L(TW) &= \min\{\max(0, 1+(TW-0.6)/0.2), \max(0, 1-(TW-0.6)/0.2)\}, \text{ and} \\ \mu_X(TW) &= \min\{\max(0, 1+(TW-0.8)/0.2), \max(0, 1-(TW-0.8)/0.2)\}. \end{aligned}$$

Table 1 gives 18 fuzzy rules to inference the term weight TW, where NTF is considered as primary factor, NDF and NIDF as secondary ones. See in [4] to refer explanation in detail. Finally, the terms with higher TW values are selected as feature vectors to classify mail messages by fuzzy inference.

Table 1. Fuzzy inference rules are composed of 2 groups according to NTF value

NTF = S	NIDF \ NDF	S	M	L
	S	Z	Z	S
	M	Z	M	L
	L	S	L	X
NTF = L	NIDF \ NDF	S	M	L
	S	Z	S	M
	M	S	L	X
	L	M	X	XX

3 Experiments

The email corpus used in the experimental evaluation contained a total of 4,792 emails and 4 categories: 2,218 for legitimate mail, 1,100 for porn spam, 1,077 for financing spam, and 397 for shopping spam. To select important features, we used the

weka.attributeSelection package provided by WEKA [6]. WEKA is a workbench designed to aid in the application of machine learning techniques to real world data sets. WEKA contains a number of classification models. The SVM classifier used in this experiment was also provided by WEKA. SVM is tested with its default parameters settings within the WEKA.

In email filtering, it is extremely important that legitimate emails are not filtered out. In comparison, a user may be satisfied if some spam-email was not filtered, in order not to miss any good email. Error rate (E) represents the ratio of the incorrect predictions over total mails [7]. Thus, good email filtering should indicate low error rate. Error rate is defined as:

$$\text{Error rate} = \frac{\text{number of incorrect predictions}}{\text{total number of emails}} \quad (1)$$

We used ten-fold cross validation to reduce random variation in the experiments. Email corpus was randomly partitioned into ten parts, and each experiment was repeated ten times, each time reserving a different part for testing, and using the remaining nine parts for training. Results were then averaged over the ten runs. Figure 1 compared the performance of the fuzzy inference and the conventional ones such as information gain and chi square in selecting features for filtering pornography spam. Almost 7,600 morphemes were extracted by eliminating stop words and redundant words. These morphemes are the candidate features to be used training porn mails. In this work, we selected 200, 338, 485, 681, and 838 features for information gain and chi square by the WEKA and for the fuzzy inference by our system among these 7,600 morphemes. When compared with the experimental results by Yang [5], it gave almost same results. As you can see in Figure 1, the fuzzy inference method improved about 6% and 10% over information gain and chi squared test in terms of the average error rate, respectively. It is more important to reduce the average error rate than to increase other performance measures such as accuracy and F-measure. Therefore, the proposed fuzzy inference is regarded as a good and stable feature selection method regardless of the number of selected features.

To evaluate the filtering performance on the email document corpus, we use the recall (R) and precision (P) commonly employed in the information retrieval field. The 4,335 emails among 4,792 ones are used for training SVM classifiers and the remaining 457 are used for testing the proposed system's performance. Testing emails are used to determine whether the mails are spam or not using the information and classifiers constructed during the training phase. We already divided hints into two kinds of information: definite information and less definite textual information. In case that an email contains one of the definite information, there is no need to perform machine learning algorithms, since it has a very high probability of being spam mails. In other case that the email has no definite information, it is evaluated using the SVM classifiers. That is to say, if an email contains one of the definite information, it is regarded as a spam mail. Otherwise, it is passed to the next SVM applying phase. SVM classifier for porn spam mails is applied first. If an email is classified as a spam mail, the second applying phase is over. If not, it is passed to the next SVM classifier for financing spam. When the email is classified as a financing spam mail, the second applying phase is over too. Like the above two SVM classifiers, the last SVM classifier for shopping spam is performed in sequence if needed.

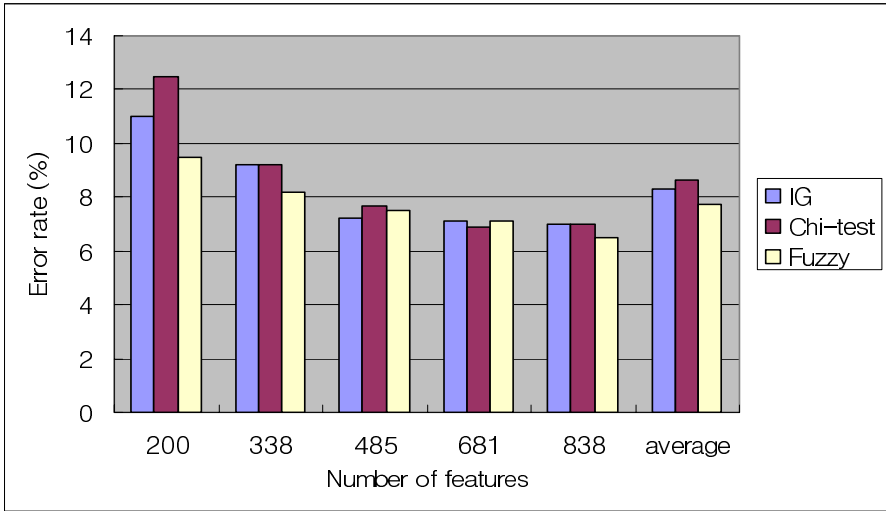


Fig. 1. Error rates of three feature selection methods

We found from Table 2 that the proposed two-phase method was more effective than the method applying each phase separately, since the 1st phase undertook some portion of the 2nd phase’s workload with very high precision. Compared the 1st phase or the 2nd phase only works, there is little change of precision, but recall was improved a lot when using hyperlinks. That is, the two-phase method improved the performance of recall by 68.6% and 9.2% over the 1st phase or the 2nd phase only works, respectively. When we consider the average of the 1st phase only and the 2nd phase only work, the proposed two-phase method improved 32.4% in terms of recall. We can recognize from these results that fetching web pages plays an important role in collecting more features and then deciding ambiguous mails.

Table 2. Performance of the proposed method (%)

Applying phase	Recall	Precision
1 st phase only	43.6	100
2 nd phase only	67.3	99.4
1 st + 2 nd phase	73.5	99.5

4 Conclusion

In this paper, we performed a comparative experiment on feature selection in pornography mails categorization. As you can see in Figure 1, the proposed fuzzy inference method lowered the average error rate over information gain and chi-squared test by the 6% and 10%, respectively. In general, it is very important to reduce the average error rate than any other measures in spam filtering domain. Though the 6 or 10% improvement of error rate by the proposed fuzzy inference looks not significant,

improving the error rate by only 1% is not easy and meaningful to email users. In also, we proposed a two-phase method for filtering spam mails based on textual information and hyperlinks. The proposed two-phase method achieved an improvement of recall by 32.4% on the average over the method of the 1st phase or the 2nd phase only work. We discovered that fetching hyperlinks is very useful in filtering spam mails, and the two-phase method is more effective than the method using machine learning algorithm only, blacklists, or keyword-based filters. This research is very important in that our system can prevent young people from accessing pornography materials on spam mails by chance, and save valuable time by lightening the email checking work. We will do further research on how to find more features by considering images in email messages and constructing ontology on spam keywords. Therefore we will improve the filtering performance.

Acknowledgement

This research was supported in part by the Daegu University Research Grant, 2005.

References

1. Sahami, M., Dumais, S., Heckerman, D., and Horvitz, E.: "A bayesian approach to filtering junk e-mail," In AAAI-98 Workshop on Learning for Text Categorization (1998) 55-62
2. Drucker, H., Wu, D. and Vapnik, V.: "Support Vector Machines for Spam Categorization," IEEE Trans. on Neural Networks, Vol.10(5) (1999) 1048-1054
3. Lewis, D. D., Schapire, R. E., Callan, J. P., and Papka, R.: "Training algorithms for linear text classifier," Proc. of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval (1996) 298-306
4. Kim, J. W., Kim, H. J., Kang, S. J., and Kim, B. M.: "Determination of Usenet News Groups by Fuzzy Inference and Kohonen Network," Lecture Notes in Artificial Intelligence, Vol.3157, Springer-Verlag (2004) 654-663
5. Yang, Y, and Pedersen, J. P.: "A comparative study on feature selection in text categorization," in Fourteenth International Conference on Machine Learning (1997) 412-420
6. Witten, I. H. and Frank, E., Data Mining: Practical machine learning tools and Techniques with java implementations, Morgan Kaufmann (2000)
7. Androutsopoulos, I., Koutsias, J., Chandrinou, V., Spyropoulos, D.: "An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages", In the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2000) 160-167

Design of Multiagent Framework for Cellular Networks

A.K. Sharma¹ and Dimple Juneja²

¹ YMCA Institute of Engineering, Faridabad(Haryana), India
ashokkale2@rediffmail.com

² Career Institute of Technology and Management, Faridabad(Haryana), India
parents_juneja@yahoo.com

Abstract. The paper begins with the in-depth dwelling of challenges posed by cellular network communication and further provides solution to those different challenges. A network that is adaptable to the demands of user is desired. For this reason it is necessary to provide agent based architecture, which is flexible, dynamic and can react proactively according to the services desired and demanded. The purpose of this research is to determine the best and an intelligent multi-agent framework for cellular networks by integrating various cooperative agents and wireless networks. Moreover the work justifies the agentification of the cellular network by discussing positive and negative aspects of the proposed feasible solution.

Keywords: Cellular Networks Communication, Multi-agent Framework and Intelligent Agents.

1 Introduction

The term “Cellular Networks” [1] refers to Advance Mobile Phone Systems (AMPS) [1] that consist of three major components: Mobile phones, Base Stations and Mobile Switching Center (MSC). In order to establish a call these three components communicate using the existing protocols in hierarchical fashion. It is impractical to have a human network operator or service provider at all times, working at the same pace and directly for all the users on network. This shortcoming has become the bottleneck with the growing size and heterogeneity of cellular networks. This work overcomes the listed shortcoming in the upcoming sections by automating the service offering area of cellular networks.

The work proposes a multi-agent framework for specifying and building distributed systems exploiting the technology of DAI and MAS. It expands our previous work [6] to propagate messages among components (Agents) that need dynamic coordination for detecting, upgrading and providing a broad range of operational scenarios. The ultimate aim is to reduce the overhead and load of a switching system assisting it with an *Intelligent Framework*. Crucial analysis of cellular networks raises the important disputes of cellular industry and subsequently provokes the use of agent technology to meet these disputes as agents are reusable functional units that need to be designed once and can be reused indefinitely.

The paper is broadly divided into three sections. Section 1 throws light on the work of eminent researchers. Sections 1.1 discover challenges and list the objectives to be

attained. Section 2 presents the proposed multi-agent framework for cellular networks. Consecutively in Section 2.1 section 2.2 the ecology and the supporting algorithms for the working of agents is presented. The next section 3 discusses benefits and limitations of proposed architecture. Finally in the end the conclusions and future work is provided.

1.1 Related Work

Related work explores the possibilities of implementation of software intelligent agents in cellular networks and thereby changing the face of cellular networks to Intelligent Cellular Networks (ICN) or Agent-Oriented Cellular Networks (AOCN). i.e. “The agent controlled cellular network and architecture that add value to conventional telecom and cellular system shall be referred to as Agent-Oriented Cellular Networks (AOCN)”.

Kazandzhiev et.al have described different approaches for building multi-agent architectures for intelligent networks and applying multi-agent technologies to e-learning area [7]. The proposed architecture implements business-logic of services in application part and SS7 data transfer specification is realized in system part.

“Current research trends are focused towards decentralization and cooperation” [8]. This area of research has not been applied to cellular networks directly but has the potential to automate or assist in the tasks being performed thereof.

Xiaocong investigate different ways of using information and knowledge in agent teamwork setting [9]. Also the paper describes the tool that facilitate human to dynamically manipulate information dependence.

Goal of MAGENTA (Mobile Agents for Telecommunication Applications) is to design a comprehensive framework for the agent-based telecommunication application. The project emphasizes the basis for the long-term development of an agent development environment.

WAVE a computational framework promises that agents provide self-organization and self-recovery from complex failures. Moreover these can do local data processing, exchange information between subsystems and between themselves.

Tannenhouse and Wetherall [10] stress that active networks allow users to inject customized programs into the nodes of network. Authors in their papers have identified number of research questions to be addressed and proposed solutions to few questions only.

Downes [11] is of the view that industry needs to shift from the idea that the network of information needs to be organized to the idea that ultimate requirement is self-organizing network of information. The author intelligently puts this is as that create resources, describe resources and use resources to the extent where very little human intervention is required.

Baldi et al. [12] calls attention to the approach for standardization and advocates for agents based management. In his work Baldi et al. showed that agent or code mobility is essential in enabling a better use of bandwidth resources and a high degree of flexibility and reconfigurability in telecommunication networks.

Critical look at the available literature reveals that number of researchers have raised the issues and also presented the solutions to some of the problems prevailing in telecom but none have explored the disputes prevailing in cellular networks. Such unrevealed issues and challenges in cellular networks are discussed in the next section.

1.2 Challenges and Objectives

Conventional cellular networks were service dependent telecommunication network i.e. service logic was hardwired in switching systems as depicted in Fig.1. Therefore network operators either had to compromise or negotiate the serving features and provide the services before the generic release of features. *The main issue here was that once the services are implemented they couldn't be easily modified to meet individual customer's requirement* [14].

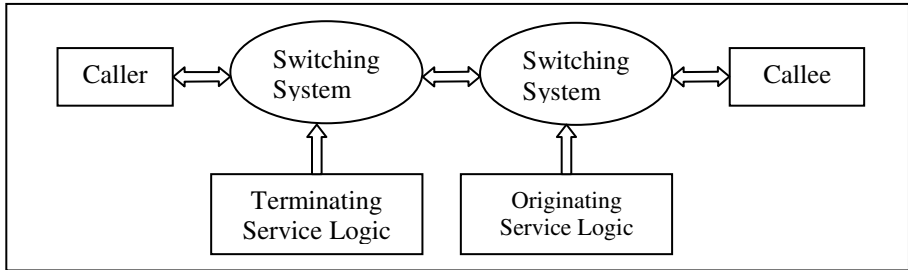


Fig. 1. Conventional Call Establishment System

In the present scenario we have Stored Program Control (SPC) switching systems in which the service logic is programmable but nevertheless they suffer from following shortcomings:

- *service logic concept is not modular*
- *Vendor dependent.*
- *It is more complicated to add new services and the service specific logic. Moreover this service specific logic that is used for one service can't be used for any other service.*

The proposed multi-agent framework shall meet the challenges imposed and subsequently shall attain the following desired objectives:

- *Generate services that are rapidly deployed in the network.*
- *Shall provide vendor independence. Service providers shall be able to integrate commercially available software and their applications.*
- *Shall have the capability to provide new services or modify existing services throughout the network without physical intervention.*
- *Service providers can change the service logic rapidly and efficiently in order to meet the individual needs of users.*

The proposed AOCN integrates the already embedded base of stored program controlled switching systems; agent based service specific logic and SS7 network. The agent technology allows for the separation of service specific functions and data from other resources. This feature reduces the dependency on switching system vendors and also service providers have more freedom to create and customize services.

2 The Proposed Multiagent Framework

Instead of one centralized and usually very large system that assumes the complete control and intelligence of network, a number of smaller systems or agents can be used to help manage the network in a cooperative manner. Consequently, this multi-agent system in cellular networks is being proposed. The architecture of the proposed system is depicted in Fig. 2. It may be noted that this framework introduces an intelligent framework instead of two conventional switches used for establishing a call. This framework consists of number of agents possessing the abilities already discussed.

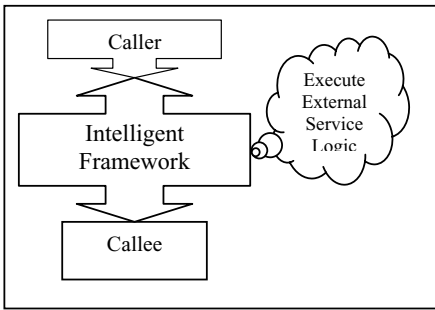


Fig. 2. Proposed Call Establishment System

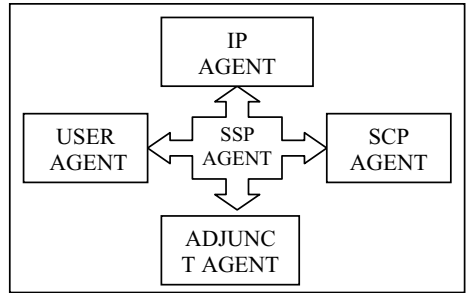


Fig. 3. Multi-Agent Framework for Cellular Networks

2.1 Ecology of Agents in Proposed Multi-agent Framework

Primarily, there are five software agents employed as User Agent (UA), Service Switching Point Agent (SSPA), Service Control Point Agent (SCPA), Intelligent Peripheral Agent (IPA), and Adjunct Agent (AA) that altogether constitutes the intelligent framework. However there can be numerous External Service Agents (ESA) basically serving as Task Agents [13]. The interaction among various agents employed is as shown in Fig 3.

Since agents in any system follows a hierarchy [3] here it is the user agent sitting on top and acts as root. SSP agent acts as interface agent that gathers information about the user whereas SCP agent is a controller or manager that not only manages the ways of processing a call but also controls the working of ESAs. IPA and AA are siblings of SCPA. All ESAs follow orders of SCPA but can cooperate among themselves to perform a desired task. It may be noted that each agent offers services to it parent agent. Table 1 lists the responsibilities of agents employed in architecture.

2.2 Working of Proposed Framework

User agent gets activated when a caller places a call and immediately interacts with SSP Agent for appropriate processing of call. SSP Agent is a capable switching agent that not only provides end users with access to network but also provides access to special capabilities of intelligent framework. SSP Agent is intelligent as it can

recognize when call processing at SCP Agent is required. SSPA and SCPA are the most interacting agents and the interface between SSP Agent and SCP Agent is represented in Fig 4.

Table 1. Agents and their Responsibilities

Agent	Responsibilities
<i>User Agent (UA)</i>	Parent Agent provides Interface between user and intelligent MAF, requests a call to be established
<i>Service Switching Point Agent (SSPA)</i>	Gathers information about user and recognizes when call processing at service control point agent (SCPA) is required
<i>Service Control Point Agent (SCPA)</i>	Invokes ESA and Analyzes Route for processing the call
<i>External Service Logic (ESA)</i>	Lowest Level Agents that actually performs the task requested.
<i>Intelligent Peripheral Agent (IPA)</i>	Provides tones, announcements, and voice recognition
<i>Adjunct Agent (AA)</i>	Special Supportive Agent functionally similar to SCPA

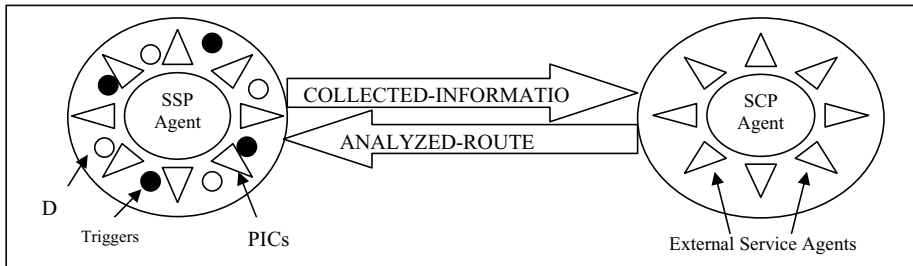


Fig. 4. SSP Agent-SCP Agent Interface

The SSP Agent usually interacts with SCP Agent in order to analyze the route for processing the call appropriately. SSP Agent is implemented in two different states. Points_in_Call (PICs) in Fig 4 represent the normal switching activities or states that call go through from origination to termination whereas Trigger_Detection_Points (TDPs) are added to see if there are any active triggers.

The triggers in SSP Agent are classified to be of three types as

- Originating Triggers - operate at the originating switching system during call set-up
- Mid-call Triggers - operate during speech phase of a call
- Terminating Triggers - operate at the terminating switching system during call set-up

Usually a trigger is activated when a user demands services other than normal call processing. If a trigger is activated, normal call processing is suspended until SSPA

and SCPA complete communication. At this instant of time SSPA assembles an information- collected message and sends it to the SCPA over SS7 Network. SCPA on reception of this message invokes a particular ESA. After ESA processes the message, the SCPA sends an analyze-route message that tells SSPA how to handle the call before going to next PIC. SSP agent can communicate with other network systems as defined by the individual services.

Intelligent Peripheral Agent (IPA) handles specialized interactions between the user and the AOIN. It has resources such as tones, announcements, and voice recognition e.g. SCPA may instruct SSPA to route a call to an IPA. The IPA will interact with the user to obtain more information. This information is forwarded to the SCPA for further processing. To increase the reliability, fault tolerance and robustness of the system, Adjunct Agent (AA) is provided as a special supportive agent that gets activated rarely in case SCPA is busy. It is functionally similar to SCPA and has direct interaction with SSPA.

2.3 Implementation Details

The logical design of this MAF is simple because the complexity of the middleware has been abstracted from the user. The complete algorithms required for the working IP Agent, SCP Agent, User Agent and SSP Agent are given in Fig 5(a), 5(b), 5(c) & 5(d) respectively. Since Adjunct agent is functionally similar to SCP Agents therefore no separate algorithm for it is being provided. SCPA may be reused when and where required.

```

IP_Agent ( )
{
  repeat
  {
    wait(flag_sufficient;

    signal(SSP_Agent
    triggers IPA);

    collected_information
    = User_agent
    (caller_phone_no);

    SSP_Agent( collected_
    information);

  }
  forever;
}

```

```

SCP_Agent ( )
{
  repeat
  {wait(SSPA forwards collected_information);
  signal(busy);
  if(flag_sufficient)
  ES_Agent=SM_Agent(collected_information);
  switch(ES_agent)
  {case "Service_Creation_Template ":
    screen_outgoing_call;
    Route to callee;
    break;
  case "Digit Extension Dialing" :
    determine the real telephone number;
    Route to callee;
    break;
  case "Disaster Recovery Service" :
    route to normal business location or to
    alternate business location;
    break;
  case "Do-not-Disturb-Service" :
    Plays special announcements
    break;
  default : Reject the call; break;
  }else IP_Agent(); }forever;
}

```

Fig. 5(a). Algorithm for IP Agent

Fig. 5(b). Algorithm for SCP

```

User-Agent ( )
{Repeat
{Wait (any_caller_calling);
Caller_phone_no=read phone no. of
the caller;
Signal (some_caller_calling);
Flag=authenticate
(caller_phone_no);
If (flag)
{Call_group=KB_Agent
(caller_phone_no);
Switch (call_group)
{case "ordinary": accept the call;
break;
case "special" : signal
(special_caller_calling);
collected_information=ssp_agent(s
pecial);
break;
case "others" : Reject the call;
break;
} } }forever;

```

Fig. 5(c). Algorithm for User Agent

```

SSP Agent ( )
{
repeat{
wait(user_agent
forwards_any_call);
signal(special_caller_calling
);
switch(special_caller)
{ case "PIC": accept the call;
break;
case "DP": signal(trigger);
collected_information=
Trigger_Agent(special_caller)
;
route_analyzed=
SCP_agent(collected_informati
on); break;
default : Reject the call;
break;
} }forever;
}

```

Fig. 5(d). Algorithm for SSP Agent

3 Benefits and Pitfalls of Proposed System

The services demonstrated in algorithm of SCP Agent are limited but offered by AOCN are listless. This defines the potential of Agent-oriented Cellular Networks. It is reusability that has proved to be most beneficial as network provider reuses embedded plants in switching systems. Agents designed are smaller in size and therefore these communicate fast using any agent communication language. Agents-oriented systems are themselves robust and fault tolerant therefore service providers provide services to the market place more reliably. AOCN is a sophisticated framework since the approach followed is completely modular. Being restricted in information accessibility and limited in knowledge for interpreting information, it is critical for The agents to be able to effectively collaborate with each other. Agents designed are static and weak. For communication and cooperation among themselves these are language dependent.

4 Conclusions

This research has introduced major breakthrough in the technology applied for the operation of switching system in cellular networks. Injecting intelligence into cellular networks will make them self-sustaining and decision-making. In order to achieve operational simplification and business objectives, the proposed framework perhaps more than any other will enable organizations around the world to be more responsive and competitive into the upcoming years. Enabling customers to customize the services to meet their specific needs may extend the work. Adding mobility among agents may extend the Proposed work. Similarly the concept of *ants* shall enhance the performance of this system.

References

1. <http://www.iec.org>," Web Proforum Tutorials on GSM,SS7, Cellular Communication, Intelligent Networks.
2. Salah Aidarous, Thomas Plevyak, " Telecommunication Network Management : technologies and Implementations", IEEE series on Network Management, IEEE Press.
3. Costas Tsatsoulis and Leen_kiat Soh " Intelligent Agents in Telecommunication Networks", CRC Press 2000.
4. Shaw Green, Leon Hurst, Brenda Nangle, Dr. Padraig Cunningham, fergal somers, Dr. Richard Evans, Software Agents: a Review", May 1997.
5. Mike Wooldridge and Nick Jennings, "Intelligent Agents: Theory and Practice", Knowledge Engineering Review, v10n2, June 1995.
6. A.K.Sharma, Dimple Juneja, Charu Bishnoi " Intelligent Agents In Call Management System", Proc. of IEEE International Conference Knowledge Intensive Multi-Agent System KIMAS-2005, pp-9-14,18th –21st April'05, Waltham, MA, USA.
7. Alexendar Kazandzhiev, Ivan Momtchev, Lozka Popova, Dimiter Shikalanov, " Distributed Multi-Agent Based Approaches", Proc. of IEEE International Conference Knowledge Intensive Multi-agent System KIMAS 2005, pp-3-8, Weltham , MA, USA
8. W.Pedrycz and A.V. Vsilakos "Computational Intelligence in Telecommunication Networks", CRC Press 2000.
9. Xiaocong Fan, Rui Wang, Bingjun Sun, Shaung Sun, John Yen, " Multi-Agent Information Dependence", Proc. of International Conference Knowledge Intensive Multi-Agent System KIMAS-2005, pp-41-52,18th –21st April'05, Waltham, MA, USA.
10. David L.Tannenhouse , David J. Wetherall, "Towards an Active Agent Architecture", LCS, MIT, USA.
11. JIME: Downes, "The Intelligent Networks", downes-2004-5-disc-15.html
12. Baldi,M., S.Gai, and G.P.Picco, "Exploiting Code Mobility in Decentralized and Flexible Network Management", in Proc. Of First International Workshop on Mobile Agents (MA'97), April ,Berlin, Germany.13-26.
13. Peter Store, Nicholas R. Jennings, Amy Greenwald, University of Texas, University of Southampton, Brown University, "Agent & Markets", IEEE Intelligent Systems, March 2003.
14. www.MobileIN.com, "Mobile Basics".

Transitive Dependence Based Formation of Virtual Organizations

Bo An¹, Chunyan Miao², Zhiqi Shen²,
Yuan Miao³, and Daijie Cheng¹

¹ College of Computer Science, Chongqing University, China
anbolangzhong@sohu.com, djcheng@cqu.edu.cn

² School of Computer Engineering, Nanyang Technological University, Singapore
ascymiao@ntu.edu.sg

³ School of Computer Science and Mathematics,
Victoria University of Technology, Australia
Yuan.Miao@vu.edu.au

Abstract. This paper first proposes a novel transitive dependence theory. Based on the proposed transitive dependence theory, a dynamic virtual organization formation framework has been worked out which includes service discovery, transitive dependence based reasoning for organization partners search and organization resolution.

1 Introduction

Virtual organizations (VOs) are composed of a number of autonomous entities each of which has a range of problem solving capabilities. Sometimes, one or more of the entities may realize there are potential benefits to be obtained from pooling resources. When this potential is recognized, the relevant entities go through a process of trying to form a new VO. In this paper, we extend direct dependence theory which is used in social reasoning [1], and propose a novel transitive dependence theory (section 3). Transitive dependence relations allow an entity to know which of its goals are achievable and which of its plans are feasible at any moment. With the notion of transitive dependence, a novel framework for formation of virtual organizations in service oriented Grid is worked out (section 4). Moreover, this research presents a transitive dependence based method for searching organization partners including dependence tree generation, dependence tree reduction, plan optimization and action optimization.

2 Formal Description of VO Formation

The formal description of VO formation scenarios VOFS is a 4-tuple $\langle AG, A, D, P \rangle$: 1) AG is the set of agents in a service oriented Grid market, for example, service providers, organizations, Yellow Page agents, QoS agents, organization management agents, etc. 2) A is the set of actions agents can perform. In VOFS, a service an agent can offer is regarded as an action the agent may perform. 3) G is the set of goals agents hold. A goal can be a consumer's request for certain service. 4) P is a set of plans agents have for the set of goals G . Each plan for a specific goal consists of a set of actions (services), i.e., $P : G \rightarrow \{a_1, a_2, \dots, a_n\}$, where $a_i \in A, i = 1, 2, \dots, n$.

In order to illustrate the formal description of VO formation sceneries VOFS, consider the following scenario. Suppose that someone visits Beijing in 2008 for the Olympic Games. Whilst in Beijing she wishes to make use of her PDA to take full advantage of various multimedia services. Typically, there can exist many service providers who offer the services that she requires, so our visitor will need to find the potential providers, select an optimal package, and then keep tracking the ever-changing market for possible better deals.

From the perspective of VOFS, all the service providers are regarded as agents, and the services these agents can offer are the actions these agents can perform. The user's wishes are her goals and there may be several plans for each goal. For a specific goal, the user can request one agent to act as the organization manager to form an organization to provide the services he asks for, and the organization manager can employ the transitive dependence based VO formation mechanism to reason out an optimal organization.

In order to make the creation of mutually beneficial organizations possible, we make the following two assumptions: 1) we assume that agents' information about the other members is complete. 2) we assume that each agent has personal rationality.

3 Transitive Dependence Theory

Definition 1: (Strong dependence) Suppose an agent ag_i tries to achieve a goal g . $p(ag_k, g) = \{a_1, a_2, \dots, a_n\}$ is a plan of the agent ag_k for the goal g . The agent ag_i has no ability to achieve an action $a_i \in p(ag_k, g)$, but it believes that the agent ag_j has ability to achieve the action a_i , then the agent ag_i strongly depends on the agent ag_j about the action a_i , i.e., $Sdep(ag_i, ag_j, p(ag_k, g), a_i)$.

Definition 2: (Weak dependence) An agent ag_i tries to achieve a goal g . There is a plan of an agent ag_k for the goal g , $p(ag_k, g) = \{a_1, a_2, \dots, a_n\}$. The agent ag_i can achieve the action $a_i \in p(ag_k, g)$ by itself, but it also believes that the agent ag_j has ability to achieve the action a_i if it pays $offer_{ag_i \rightarrow ag_j}(a_i)$ to the agent ag_j . If the agent ag_i achieves the action a_i by itself, it should spend $cost_{ag_i}(a_i)$, and $cost_{ag_i}(a_i) > offer_{ag_i \rightarrow ag_j}(a_i)$, then the agent ag_i weakly depends on the agent ag_j about the action a_i , i.e., $Wdep(ag_i, ag_j, p(ag_k, g), a_i)$.

A simple example of weak dependence is the Tieworld [2], where every agent can move tiles, and agents can save cost through cooperation.

Definition 3: Suppose the agent ag_i can achieve the action a_i , but it wants any other agent who depends on it about the action a_i to do an action a_j for it, and it has no ability to achieve the action a_j , then the agent ag_i has an action dependence on the action a_j about the action a_i , i.e., $Adep(ag_i, a_i, a_j)$.

Let A be a set of actions. The agent ag_i can achieve the action a_i . If it wants any other agent who depends on it about the action a_i to do all the actions in A , we call this kind of action dependence *and-action dependence*. If it wants any other agent who depends on it about the action a_i to do any one action in A , we call this kind of action dependence *or-action dependence*.

Now we discuss transitive dependence. For the agents ag_i , ag_j and ag_k , there are two dependence relations: $Dep(ag_i, ag_j, p(ag_m, g_a), a_i)$ and $Dep(ag_j,$

$ag_k, p(ag_q, g_b), a_j)$, where Dep can be $SDep$ or $WDep$. Although the agent ag_j can achieve the action a_i , but it has an action dependence on the action a_j about the action a_i , i.e., $Adep(ag_j, a_i, a_j)$. We can find that the agent ag_i transitively depends on the agent ag_k about the action a_j .

We call the dependence relations in definition 1 and 2 direct dependence.

Dependence chain is used to describe the transition process of transitive dependence relations. A dependence chain has a head and a tail. For the agents ag_i, ag_j and ag_k , $Dep(ag_i, ag_j, p(ag_m, g_a), a_i)$, $Dep(ag_j, ag_k, p(ag_q, g_b), a_j)$, and $Adep(ag_j, a_i, a_j)$, the dependence chain from the agent ag_i to the agent ag_k is $Dpc = ag_i \xrightarrow{p(ag_m, g_a), a_i} ag_j \xrightarrow{p(ag_q, g_b), a_j} ag_k$.

For a dependence chain Dpc , $Head(Dpc)$ represents the agent at the head of the dependence chain, and $Tail(Dpc)$ represents the agent at the tail of the dependence chain. For an agent ag_i in a dependence chain Dpc , $ToDep_act(ag_i, Dpc)$ and $Deped_act(ag_i, Dpc)$ represent the actions that the agent ag_i depends on and is depended upon respectively.¹ For the dependence chain Dpc in the last paragraph, $Head(Dpc) = ag_i$, $ToDep_act(ag_i, Dpc) = a_i$, $Tail(Dpc) = ag_k$, and $Deped_act(ag_k, Dpc) = a_j$.

Definition 4: Transitive dependence can be described as $Tdep(Depender, Dependee, Dependence\ chain)$. The direct dependence is regarded as a kind of special transitive dependence, i.e., the agent ag_i 's dependence on the agent ag_j about the action a_i that belongs to the plan $p(ag_k, g)$ can be described as $Tdp(ag_i, ag_j, ag_i \xrightarrow{p(ag_k, g), a_i} ag_j)$. Transitive dependence can be recursively defined as:

For the agents ag_i, ag_j and ag_k , $Tdep(ag_i, ag_j, Dpc_1) \wedge Tdep(ag_j, ag_k, Dpc_2) \wedge Adep(ag_j, Deped_act(ag_j, Dpc_1), ToDep_act(ag_j, Dpc_2)) \Rightarrow Tdep(ag_i, ag_k, Dpc_1 + Dpc_2)$, where $Dpc_1 + Dpc_2$ represents the connection of Dpc_1 and Dpc_2 .

4 The VO Formation Framework

The formation of virtual organizations consists of three stages: service discovery, transitive dependence based reasoning, and organization resolution.

4.1 Service Discovery Phase

Agents must have some information about other members before reasoning about possible organization partners. This kind of information is acquired during an initial service discovery phase and can be acquired and updated dynamically.

4.2 Transitive Dependence Based Reasoning Phase

Assume an agent ag_i has a goal g . If the agent ag_i has no ability to achieve an action $a_i \in p(ag_k, g)$, the agent ag_i strongly depends on other agents about the action a_i that it can't achieve; If the agent ag_i has ability to achieve the action a_i but it has to spend more than what it pays for some other agents when it asks for the help of them, the agent ag_i weakly depends on other agents about the

¹ An agent may appear more than once in a dependence chain.

action a_i . The kind of direct dependence and transitive dependence in solving a goal are represented with a dependence tree.

Definition 5: A dependence tree DPT is an ordered triple $(V(DPT), E(DPT), \Psi(DPT))$ consisting of a nonempty set $V(DPT)$ of nodes, a set $E(DPT)$, disjoint from $V(DPT)$, of edges and an incidence function $\Psi(DPT)$ that associates with each edge of DPT an ordered pair of vertices of DPT :

1. The set $V(DPT) = Root \cup V_{plan}(Root) \cup V_{ag}(DPT) \cup V_{act}(DPT)$ is the union of four disjoint sets. $Root$ is the root of the dependence tree, i.e., the depender and its goal to achieve, $V_{plan}(Root)$ is the set of plans the agents may use to achieve the goal in the root node $Root$, $V_{ag}(DPT)$ is the set of agents, and $V_{act}(DPT)$ is the set of actions.
2. The set $E(DPT)$ is a set of edges.
3. The function $\Psi_{DPT} : E(DPT) \rightarrow V(DPT) \times V(DPT)$ is defined as follows:
 - (a) $\Psi_{DPT}(e_1) = (Root, p_i)$ associates an edge e_1 with an ordered pair of vertices $(Root, p_i)$, and represents that the goal in the root node $Root$ can be achieved by the plan p_i .
 - (b) $\Psi_{DPT}(e_2) = (p_i, a_i)$ associates an edge e_2 with an ordered pair of vertices (p_i, a_i) , and represents the fact that the plan p_i needs the action a_i and the action can't be achieved by the depender or the depender can achieve it but has to spend more.
 - (c) $\Psi_{DPT}(e_3) = (a_i, ag_i)$ associates an edge e_3 with an ordered pair of vertices (a_i, ag_i) , and represents the fact that the action a_i can be performed by the agent ag_i .
 - (d) $\Psi_{DPT}(e_4) = (ag_i, a_i)$ associates an edge e_4 with an ordered pair of vertices (ag_i, a_i) , and represents the fact that the agent ag_i has an action dependence on the action a_i , i.e., $ADep(ag_i, a_k, a_i)$, where the action a_k is the origin action node of the agent ag_i .

Here we define two notations for the dependence tree. Let a node x be any node but the $Root$ node in a dependence tree DPT . $father(x)$ represents the father node of the node x , and $sons(x)$ represents the set of nodes whose father node is the node x . Each node x has only a father node $father(x)$, and may have more than one son node, i.e., $|sons(x)| \geq 0$.

To illustrate the dependence tree, an example is given and the corresponding dependence tree is in Fig.1.² An agent ag_1 has a goal g_1 and it has two alternative plans, $p_{11} = \{a_0, a_1, a_2\}$ and $p_{12} = \{a_0, a_3\}$, for g_1 . Suppose the agent ag_1 can only perform the action a_0 , and the actions a_1 and a_2 can be performed respectively by the set of agents $\{ag_2, ag_3\}$ and $\{ag_4\}$. The agent ag_5 can achieve the action a_3 . An action dependence relation of the agent ag_2 is $ADep(ag_2, a_1, a_4)$. An or-action dependence relation of the agent ag_3 is $ADep(ag_3, a_1, a_5 \vee a_6)$. An and-action dependence relation of the agent ag_5 is $ADep(ag_5, a_3, a_7 \wedge a_8)$. The action a_5 can be performed by the agent ag_6 . The action a_7 can be performed

² The or-dependence relations are represented with dotted lines.

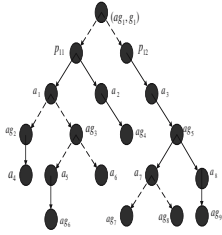


Fig. 1. A dependence tree

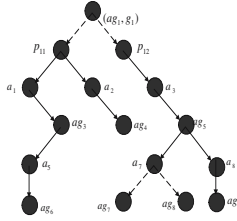


Fig. 2. A reduced dependence tree

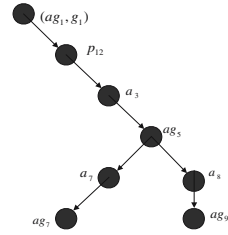


Fig. 3. An organization tree

by the agents ag_7 and ag_8 . The action a_8 can be performed by the agent ag_9 . The actions a_4 and a_6 can't be done by any agent.

Definition 6: (Feasible action) For an action node a_i in a dependence tree, it's a feasible action if it can be achieved, which can be defined by: 1) If the prior node of the action node a_i is a plan node, and the depender in the *root* node has ability to achieve the action a_i , the action a_i is a feasible action. 2) If an edge e associates with an ordered pair of vertices (a_i, ag_j) and the node ag_j is a leaf node, the action a_i is feasible. 3) If an edge e associates with an ordered pair of vertices (a_i, ag_j) , the node ag_j is an "or" node, and there is more than one ordered pair of vertices (ag_j, a_k) , in which the action a_k is feasible, then the action a_i is a feasible action. 4) If an edge e associates with an ordered pair of vertices (a_i, ag_j) , the node ag_j is an "and" node, and for every ordered pair of vertices (ag_j, a_k) , the action a_k is feasible, then the action a_i is a feasible action.

For a plan p_i in a dependence tree, if for every ordered pair of vertices (p_i, a_i) , the action a_i is feasible, the plan p_i is a *feasible plan*.

After deletion of the all actions and plans that are not feasible from a dependence tree, the reduced tree is called a *reduced dependence tree*.

Take the dependence tree in Fig.1 as an example. Assume that the agent ag_2 has no ability to achieve the action a_4 , and the agent ag_3 has no ability to achieve the action a_6 . Fig.2 shows the reduced dependence tree.

For a reduced dependence tree, if the node *Root* is a leaf node, then the goal is not achievable; otherwise, it's a feasible goal.

There may be more than one feasible plan for the goal. For an action node, there may be more than one agent which can achieve the action. Thus, the depending agent should make a decision on choosing the best plan for the goal and best agent for an action. It's obvious that both the two problems are of high complexity and there is a strong dependence between the two problems. To avoid the high complexity of the two problems, here we utilize a solution of bounded rationality. The two problems with mutual influence are divided into two independent problems: *plan optimization* and *action optimization*, which will reduce the computational complexity of finding the best plan and the best cooperation partner, but, accordingly, the organization result is not optimal.

Plan Optimization: For a feasible goal, there may be more than one feasible plan, and we choose the most favorable plan to form organization.

Action Optimization: Similarly, for a feasible action, there may be more than one agent which can achieve it, and the agent that will result in the most favorable outcome is chosen to achieve the action.

For a reduced dependence tree, after plan optimization and action optimization, the reduced dependence tree is called a *organization tree*.

Since to find out the best solution for a specific goal is of high complexity, in this paper, we give an anytime algorithm [3] to find the best solution in the allowed time range.

Algorithm. Suppose $\text{Index}()$ is the index function to evaluate a solution. It returns a numeric value. The smaller number indicates a better solution.

1. if there is no feasible plan/action, return;
2. select a feasible plan/action;
3. compute its index value using index function;
4. if it is the first solution, keep the solution and the index value;
5. else if the index value is less than the kept one, replace the solution with the current one and keep the index value;
6. else go to 1.

We can find that the best solution so far will be chosen as the final solution when the deadline approaches. Thus, at any time, there is a dynamic organization can be formed. For the reduced dependence tree is in Fig.2, let us assume that the agent ag_1 has to pay much more if it adopts the plan p_{11} than that if it adopts the plan p_{12} , and the reserve price of the agent ag_7 is less than that of the agent ag_8 for the action a_7 . Fig.3 shows the organization tree.

4.3 Organization Resolution Phase

In this phase, the depending agent invites all the agents in the organization tree to form an organization. According to assumptions 1 and 2, all the agents in the organization tree will agree to join in the organization for cooperation.

5 Conclusions and Future Work

Compared with other VO formation mechanisms (e.g., cooperative game theory based methods used in coalition formation and supply chain formation), the transitive dependence based one seems more intuitive. With the anytime algorithm, the depending agent can find the best solution so far.

The proposed organization formation framework can be applied in many other application domains, such as, the application in autonomous and service oriented computing, supply chain management, workflow, enterprise integration.

References

1. Sichman, J.S., Conte, R., Demazeau, Y., Castelfranchi, C.: A social reasoning mechanism based on dependence networks. In: Proc. of the 16th ECAI. (1994) 188–192
2. Zlotkin, G., Rosenschein, J.S.: Coalition, cryptography, and stability: Mechanisms for coalition formation in task oriented domains. In: Proc. of AAAI 94. (1994) 432–437
3. Hansen, E.A., Zilberstein, S.: Monitoring and control of anytime algorithms: A dynamic programming approach. *Artificial Intelligence* **126** (2001) 139–157

An Agent Based Education Resource Purvey System

Xiaochun Cheng^{1,2}, Xin He¹, Xiaoqi Ma¹,
Dongdai Zhou², Peijun Duan², and Shaochun Zhong²

¹ Department of Computer Science, The University of Reading,
Reading RG6 6AY, England, UK

{x.cheng, x.he, xiaoqi.ma}@rdg.ac.uk

² Institute of Ideal Information Technology, Northeast Normal University,
Changchun, 130024, China

Abstract. Most current education organizations use books and CDs as the main media, which takes a long time for knowledge updating between education resource providers and the users. The rapid development of the Internet has brought with it the possibility of improving the resource purveying mechanisms. Therefore, we designed an agent based system to purvey education resources from the resource centre to schools through the Internet. Agent technology helps to improve system performance and flexibility. This paper describes the design of our system, details the functions of the main parts of the system, shows the communication methods between agents and finally evaluates the system by experiments.

1 Introduction

Education Resource Purvey Systems (ERPS) provide required educational resources from resource providers to users [1][3][6]. Traditional ERPS uses books and CDs as the main media and could have the following disadvantages: (1) the normal cycle of renewal involves updating books and CDs, which takes a lot of time and money; (2) traditional resources could contain contents different from what the students need. Thus, it is necessary to design a flexible, timely and customized education resource purvey system.

Internet can be deployed to purvey education resources [1][2][3][4][11], since it has become available at most schools. It is feasible to produce an efficient network-based education resources purvey system. Agent technology support autonomous actions in the complex changing environment in order to meet the design objectives [1][6][8]. In a satisfying education resource purvey system, the knowledge and information transmits from the resource centre to remote users, regarding the customized user requirements and dynamically changing resources. We apply agent technology to education resource purvey system design. [1][2][5][6].

Section 2 outlines the system architecture. It explains the options of different agents according to their characteristics and the tasks. Section 3 introduces the implementation of the system and evaluates the system with experiments. Section 4 summarizes the project and provides suggestions for future research.

2 System Architecture

The resource centre manages the education resources and controls the resource purveying. The distributed purvey pattern [1][5][6] adopted in the system helps to reduce the network traffic load.

2.1 System Design

According to the roles and tasks of system administrators and users, the system architecture is designed as shown in Fig. 1.

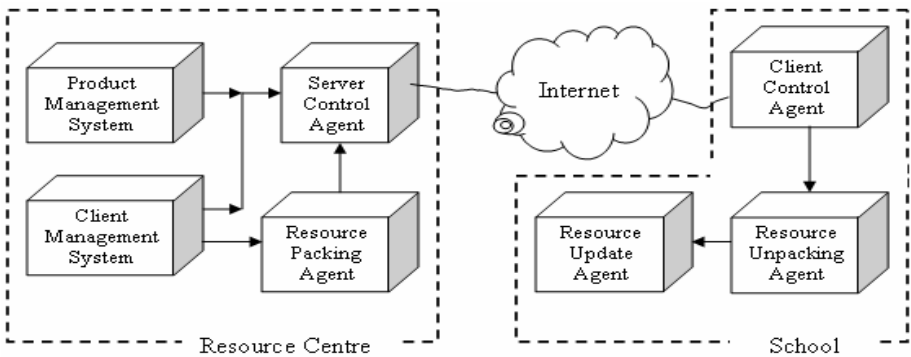


Fig. 1. Agent based education resource purvey system architecture

In the resource centre, the resource packing agent monitors changes of the resources in the centre and the requirements about the resource packing from client systems or the resource centre itself with the assistance of the client management system. If current resources are updated or new resources are introduced into the centre, the resource packing agent will package the updated parts or the newly introduced parts in order to pass them through the Internet. Then, the information about the newly packaged resources is published. Any system client (client control agent) who is interested in the news would contact the server control agent. After the server control agent identifies the user’s status, a permission to download the packaged resources is granted to the client. Resource unpacking agent unpacks the obtained resource packages. Resource update agent monitors all resource updating from the resource centre and updates changes to the local knowledge grids [1][2][3] concerned.

2.2 Analysis of the Agents

Resource Packing Agent. The main functions of the resource packing agent include resource analysis, resource compression, resource encryption, resource packing and resource publishing. Its work flow diagram is shown in Fig. 2.

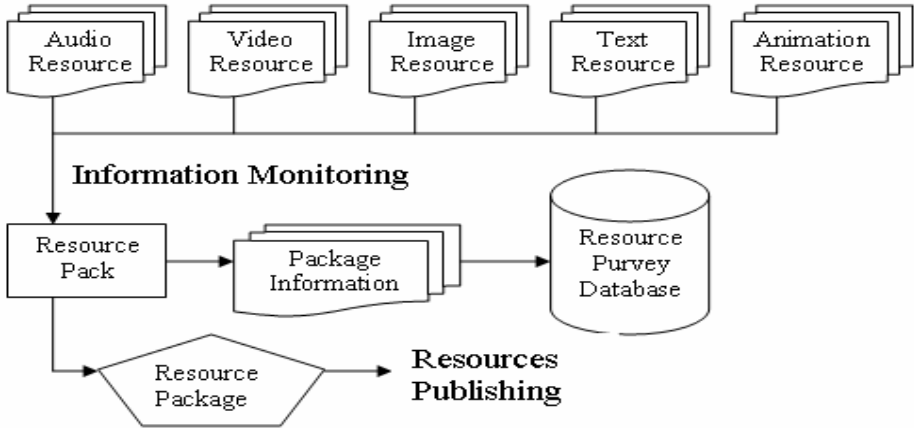


Fig. 2. The work flow diagram of the resource packing agent

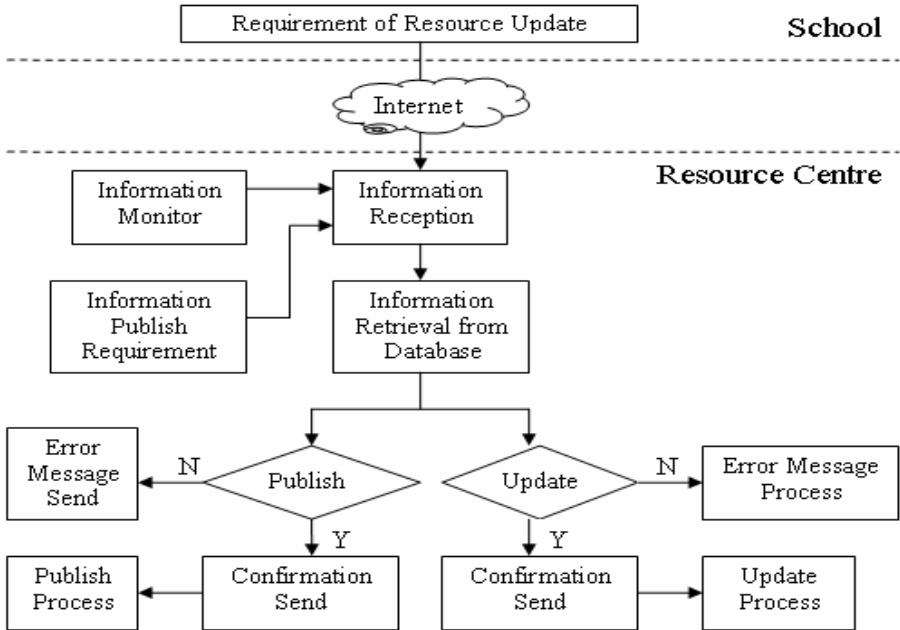


Fig. 3. The work flow diagram of the server control agent

Server Control Agent works at the interface between the resource centre and individual clients. The main functions of the server control agent include checking and developing policies on resource purveying, receiving resource updating information, sending resource updating information, monitoring resource updating requirements from clients, keeping records of resource purveying, communicating with client control agents, dealing with accidents and dynamic assigning network load. Its work flow diagram is shown in Fig. 3.

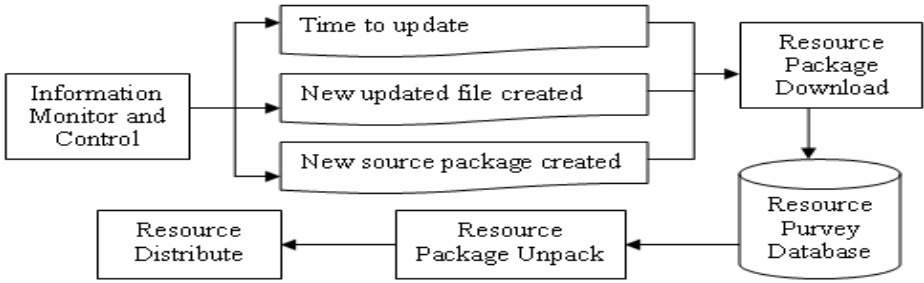


Fig. 4. The work flow diagram of the client control agent

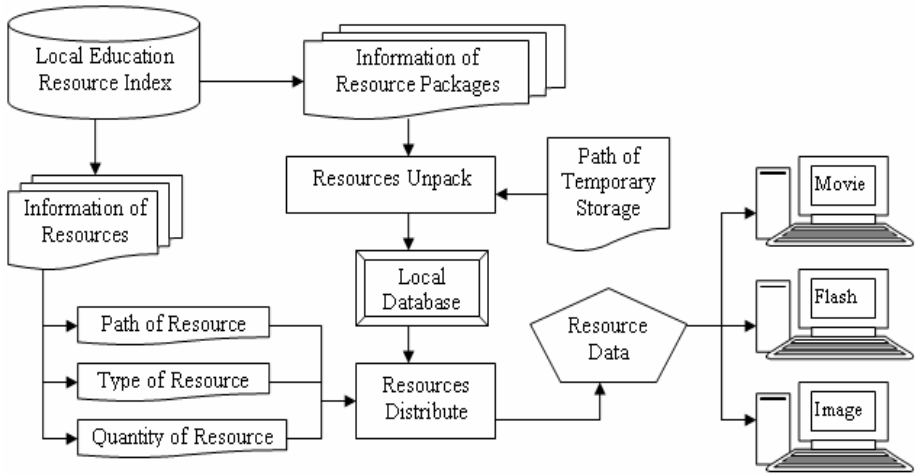


Fig. 5. The work flow diagram of the resource unpacking agent

Client Control Agent is an important part of each client system. The main functions of client control agents include developing policies on resource updating, communicating with the server control agent to find out whether there is a new resource packet published by the resource packing agent, keeping a record of each resource purveying from the resource centre, asking for resource updating information from the resource centre, downloading and verifying the resource packages from the resource centre and keeping a log file for that, maintaining a backup of the local educational resources, updating local educational resources after resource unpacked by the resource unpacking agent, maintaining the local educational resources, submitting the requirements for resource updating and coping with the accidents in the local network with the assistance of the server control agent. Its work flow diagram is shown in Fig. 4.

Resource Unpacking Agent. The main functions of resource unpacking agents include analysis of resources to be downloaded, unpacking resource packages from the resource centre, decrypting the resource packages, distributing and

storing resources in local education resource systems. Its work flow diagram is shown in Fig. 5.

Resource Update Agent starts to work automatically when resource unpacking is completed. The main functions of resource update agents include setting the school resource updating policies, analysis of the local educational resources, performing the resource auto-updating, analysis of the unpacked resources.

3 System Implementation and Evaluations

There are two kinds of communications in our system. The agent communication language (ACL) is employed incorporated for the communications between local agents [8], while point-to-point communication method is used through the internet in order to ensure the security and integrity of the sent messages [7].

To evaluate the efficiency of our system, experiments were conducted first with local clients. Later on, some remote client systems were chosen for testing. During these experiments, downloading was fulfilled successfully and resource packages were verified and unpacked correctly. According to the experiment results, applying agent technology into our education resource purvey system is successful.

4 Conclusion and Future Work

In this paper, we proposed a new agent based education resource purvey system. The system architecture and the specification of its modules are outlined firstly. Detailed agent specifications are given consequently. Finally, the system implementations and evaluations are reported. The experiment results corroborate the expected advantages of this system. Further research to improve the system load balance will be conducted [9]. Some abilities of agents (e.g. inheritance, management of learning, mobility) are supposed to benefit the services [10].

Acknowledgement

Our research has been supported by EC, EPSRC, the National Natural Science Foundation of China, and Hong Kong K C Wang Education Foundation.

References

1. Li Y, Cheng X, Zhong S, Yu X, Agent-based Grid Platform for Active E-Learning, Proceedings of the IEEE SMC UK-RI 3rd Workshop on Intelligent Cybernetic Systems(ICS'04), pp 144-148, Londonderry, Northern Ireland, UK. Sept 2004.
2. Li Y, Cheng X, Li Z., Ouyang D., Cheng M., Yu X. and Zhong S., Semantic Organization to Enhance Active Learning Based on Distributed Multimedia Educational Resources, IEEE International Symposium on Consumer Electronics, 2004, ISBN 0-7803-8527-6, pp 436-441. September 2004. Reading, UK.

3. Zhang D, Zhong S, Cheng X, Actively Supply Personalized Educational Resources for Elementary Schools, Proceedings of the 10th Annual Conference of Chinese Automation and Computing Society in UK (CACSUk 2004). Liverpool, UK. September 2004. ISBN: 0 9533890 6 5, Pages 177-182.
4. He R and Zhong S and Cheng X, The Research about a Chinese Personalization Education Resources Search Engine, Proceedings of the IEEE SMC UK-RI Chapter Conference 2005 on Applied Cybernetics, 147-152, September 7-8, 2005, City University, London, UK.
5. Jin Y, Zhong S, Cheng X, Design of Intelligent Tutoring System Based on Internet, Journal of Guangxi Normal University, Vol. 22, No.3, pp 19-23. September 2004.
6. Duan P, Zhou D, Cheng X, The Research on Individual Service Model of Teaching Resources based on Multi-Agent Technology, Education Technology Research, 2004 (5): 64-67.
7. Wei, X.: An XML Based Agent Communication Framework. Computer Engineering and Application. 18:(10), 108-112, 2001.
8. Wooldridge, M.: An Introduction of Multi-Agent Systems. John Wiley & Sons Inc, 2002.
9. Sun P, Ma X, Cheng X, Zhong, S, Knowledge Management for E-learning System. Proceedings of the 10th Annual Conference of Chinese Automation and Computing Society in UK (CACSUk 2004). Liverpool, UK. September 2004. ISBN: 0 9533890 6 5, Pages 193-198.
10. He R, Zhong S, Cheng X, The Application of Knowledge Engineering Research in E-Course Integration, NENU Science Journal, Vol 35, October, 2003, pp46-50. ISSN 1000-1832.
11. Zhang D, Zhong S, Cheng X, The Research for Teaching Chinese as Foreign Language based on Internet, Journal of Guangxi Normal University, Vol. 22, No.3, pp 24-28. September 2004.

Model of Game Agent for Auction-Based Negotiation

Jun Hu and Chun Guan

Department of Computer Science and Engineering, Nanchang University,
Nanchang 330029, China
Jx_hujun@163.com

Abstract. Some actual multi-agent automated negotiation systems using auction mechanism in e-commerce are inefficient, as buyer negotiation agents are lack of enough rationality and uneasy to determine appropriate bid price automatically according to different circumstance. In order to improve the auction-based negotiation efficiency of multi-agent system in e-commerce, this paper proposes a model of game agent for auction-based negotiation and a bid negotiation algorithm based on game theory, which provide a new effective way to establish buyer game agent for making buyer agent determine price more accurately and bid more rationally. As buyer agents bid in a more rational way, auction agents and buyer agents can finish negotiation more efficiently.

1 Introduction

Software agents are programs to which one can delegate a task, and agent technologies can be applied to any of these areas where a personalized, continuously running, semi-autonomous behavior is desirable. These qualities make agents particularly useful for the information-rich and process-rich environment of electronic commerce. Multi-agent automated negotiation system for e-commerce can be regarded as an electronic virtual market between buyer agents and seller agents for trade of goods or services, agents automatically look for appropriate sellers or buyers in this virtual market and negotiate with them for the optimal trade by market mechanism. Auction is a relatively simple market negotiation mechanism, which is well structure and easy to carry out automatically, provides a favorable market mechanism for bilateral, multilateral and multi-issue negotiation in multi-agent automated system of e-commerce. However, in some actual multi-agent automated negotiation systems using auction mechanism in e-commerce, like Kasbah [1], AuctionBot [2], eMediator [3], etc, buyer negotiation agents are lack of enough rationality, It is uneasy to determine appropriate bid price according to different circumstance and as a result, negotiation is inefficient. In order to improve the efficiency of auction-based negotiation system in e-commerce, this paper proposes a model of game agent for auction-based negotiation and a bid negotiation algorithm based on game theory, which provide a new effective way to establish buyer game agent for making buyer agent determine price more accurately and bid more rationally.

2 Model of Game Agent for Auction-Based Negotiation

Game theory focuses on the study of how people interact and make decisions, which is a special branch of mathematics developed for studying decision-making in complex circumstance [4]. From the opinion of game theory, negotiation is the process in which decision-makers decide optimal strategies for equilibrium. Auction, belonging to incomplete information static game, is an important applied field of the bayes-nash equilibrium [5], thus, incomplete information static game can help negotiation agents to analyze equilibrium bid strategy effectively. During auction-based negotiation, agents are self-interested, auction agents pursuit max profit in selling commodities, however, buyer agents aim to minimize payoff for buying commodities.

If buyer agents bid in equilibrium strategies in the process of auction-based negotiation, auction agents and buyer agents can reach a bayes-nash equilibrium in limited steps, get solutions and finish the negotiation according to incomplete game theory. If buyer negotiation agents can compute and analyze bayes-nash equilibrium strategies automatically, determine bid prices in these strategies, they can finish negotiation more effectively. Thence, a game agent model for auction-based negotiation is proposed in this paper, which can make buyer negotiation agent bid more rationally, get more powerful ability of game analysis, determine price more accurately aiming to get commodities in the lower price, and improve the efficiency of auction-based automated negotiation in multi-agent system. The structure of model is illustrated in Figure 1.

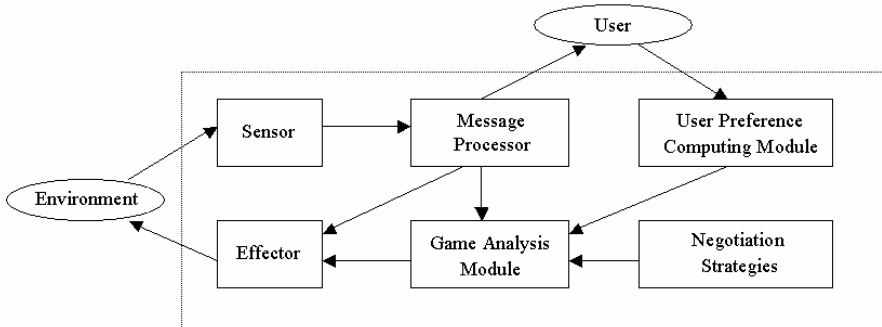


Fig. 1. A model of game agent for auction-based negotiation

The model of game agent for negotiation is composed of sensor, effector, message processor, user preference computing module, game analysis module and negotiation strategies.

Sensor has a responsibility for receiving messages from environment. Receiving thread ceaselessly receive the stream of ASCII, convert stream to messages and save them to the receive message buffer queue.

Effector mainly takes charge of transmitting messages to environment. Sending thread check the transmission message buffer queue continuously, when a new message needs to be transmitted, convert the message to ASCII stream, which can be sent.

Message processor is charge of sorting and processing messages. Its threads check the receive message buffer queue continuously, when a new message arrives, parse this message with the KQML/XML parser, then sort and process message according to multi-issue negotiation protocol based on auction mechanism.

User preference computing module, designed to establish user individual preference model based on commodity content, can establish the membership function of commodity properties and determine weights of commodity properties in GUI. When negotiation agent on behalf of user performs bargain negotiation, It can rationally adjust negotiation price based on the established preference model.

Game analysis module is the core of automated game negotiation agent, which is composed of the automated bid negotiation algorithm based on auction game and the bid negotiation module. Game analysis based on bid strategies can be realized in this module, It is designed to apply the automated bid negotiation algorithm based on game to compute optimal equilibrium strategies for bidding according to user preference, negotiation strategies, auction type and received messages, and send bid messages to the effector for transporting to auction negotiation agent.

Negotiation strategies is a set of strategies used in bid decision-making by negotiation agent, which is composed of risk strategy, demand strategy, preference strategy and combinatorial strategy, etc.

3 Bid Negotiation Algorithm Based on Game

According to incomplete information game theory, in the process of auction-based negotiation, bidder will get bayes-nash equilibrium in limited steps, namely negotiation solution, and can finish negotiation as long as bidder adopt equilibrium strategies to bid. If buyer negotiation agents can compute and analyze bayes-nash equilibrium strategies automatically, determine bid prices by these strategies, they will finish negotiation effectively. Different bid strategies are determined in static game of incomplete information [6], according to different auction ways and different circumstances. In order to realize automated bid game strategy analysis of bidders, enforce buyer agent rationality and improve negotiation efficiency, this paper proposes an automated bid negotiation algorithm based on game theory.

This paper supposes that the number of game agent is N , the number of capable strategies is M , auction is used in the benchmark model. The automated bid negotiation algorithm based on game is followed:

```

Begin
  Bests=0;Bestu=0;
  Evaluating price  $V_i$  according to user preference;
  if English auction then
    bidding in the dominant strategy of English
    auction game;
  else if Second-price auction then
    bidding in the dominant strategy of second-price
    auction game;
  else if Dutch auction then
    bidding in the equilibrium strategy of second-price
    auction game;

```



```

else if First-price auction then
    
$$b(v_i) = v_i - \frac{\int_{v_i}^{v_r} [F(\delta)]^{n-1} d\delta}{[f(v_i)]^{n-1}} ;$$

else
    Begin
        for k:=1 to M do
            Begin
                computing b (sk) ;
            U = (Vi - b(sk)) ∏j≠i prob(bj < b) ;
                if U>Bestu then
                    Begin
                        Bestu=u; Bests=k;
                    End;
                End;
            b (sbests) ;
        End;
    End.

```

The algorithm can be explained as below. Supposes that there are N players in game, each player has M kinds of strategies in the benchmark auction model. Each player evaluates commodity price by the independent private value model, which is a weighted function formed of some influence price factors. Supposes that $i = \{1 \dots n\}$ denotes the No. i game negotiation agent, and $j = \{1 \dots k\}$ denotes the No. j factor in negotiation which affects commodity price, $x_j^i \in [low_j^i, high_j^i]$ denotes the No. i Agent's No. j factor value, and its domain value is $[low_j^i, high_j^i]$, the membership function of this factor is $y_j^i : [low_j^i, high_j^i] \rightarrow [0,1]$, which denotes the degree of preference for player to this factor and can be a single raise or decay function. w_j^i denotes the important degree of No. j factor to game player i, $\sum_{j=1}^k w_j^i = 1, w_j^i > 0$. v_0^i denotes commodity price in the domain knowledge of game player i, a is a constant, then function of independent private value to commodity can be expressed as $v_1^i(x) = \sum_{j=1}^k (w_j^i * y_j^i * a * v_0^i)$. Different equilibrium bid strategies in different type of auctions can be determined in the static game theory of incomplete information.

4 Experiments

In order to validate the efficiency of automated bid negotiation algorithm based on game theory, a negotiation agent system is established, which includes one auction

negotiation agent as A and twenty game negotiation agent as B_i . Agent B_i respectively makes use of the automatic bid negotiation algorithm based on game and normal bid negotiation strategies without game to negotiate for buying commodities with auction agent A. These normal bid negotiation strategies include anxious negotiation strategy, coolheaded negotiation strategy and frugal negotiation strategy used in the Kasbah electronic market by MIT Media Lab.

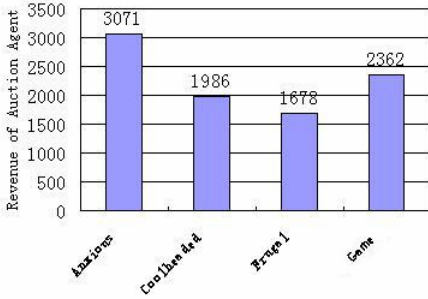


Fig. 2. Revenue of auction agent

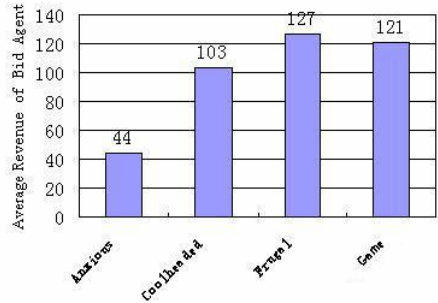


Fig. 3. Average revenue of bid agent

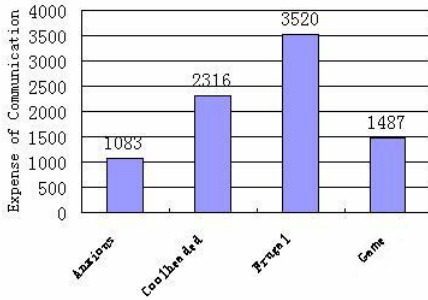


Fig. 4. Expense of communication

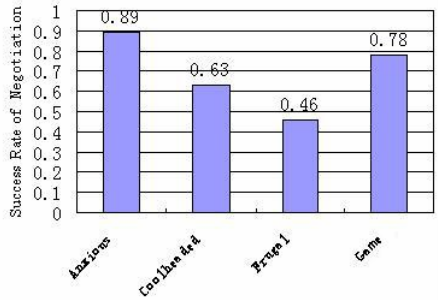


Fig. 5. Success rate of negotiation

In our experiment agent system, agent A sells commodities by auction, twenty buyer agents use the same strategy to bid. In order to compare different strategies, each of negotiation strategies is experimented over 1000 times. The experiment mainly considers the revenue of auction agent, average revenue of bid agent, expense of communication, success rate of negotiation. Experiment results of all negotiation strategies are illustrated in figure 2, 3, 4, 5.

When agents negotiate in the automated bid negotiation algorithm in these experiments, auction agent and bid agents can get better revenue, expense of communication is the least in all strategies, success rate of negotiation is high. As negotiation result is equilibrium, It is uneasy to regret after agreement for each agent. Experiment results show that the game strategy is superior to general negotiation strategies based on individual activities.

5 Conclusions

Some multi-agent automated negotiation systems using auction mechanism in e-commerce are inefficient, as buyer negotiation agents are lack of enough rationality and uneasy to determine appropriate bid price automatically according to different circumstance. In order to improve the efficiency of multi-agent automated negotiation system, make buyer agents bid more rationally and determine price more accurately, this paper proposes a model of game agent for auction-based negotiation and a bid negotiation algorithm based on game theory. The model and the algorithm provide a new effective way to establish game negotiation agent. Because game negotiation agents can bid in a more rational way, It is easy for them to come to Nash equilibrium, maximize their individual benefit and finish negotiation efficiently. And It is simply applied to implement a automated agile supply chain multi-agent system in a small manufacture enterprise to deal with automated negotiation bargain for product, raw materials and transport services. It functions perfectly in this period of time although some improvements are still necessary at the point such as user preference selection of score functions and evaluations of experiments.

References

1. He, M., Jennings, N. R., Leung, H.: On agent-mediated electronic commerce. *IEEE Transactions on Knowledge and Data Engineering* (2003), 15(4), 985-1003.
2. Moukas, A., Guttman, R., Maes, P.: Agent-mediated Electronic Commerce: An MIT Media Laboratory Perspective. *Proceedings of the First International Conference on Electronic Commerce (ICEC-98)*, Seoul, Korea.
3. Sandholm, T.: eMediator: A next generation electronic commerce server. *Computational Intelligence* (2002), 18(4), 656--676.
4. Fudenberg, D., Tirole, J.: *Game theory*. MIT Press (1991).
5. Weibull, J. W.: *Evolutionary game theory*. MIT Press (1995).
6. McAfee, R. P., McMillan, J.: Auctions and bidding. *Journal of Economic Literature* (1987), 25, 699-738.

An Autonomous Mobile Robot Based on Quantum Algorithm

Daoyi Dong, Chunlin Chen, Chenbin Zhang, and Zonghai Chen

Department of Automation, University of Science and Technology of China,
Hefei, Anhui 230027, China
{dydong, clchen}@mail.ustc.edu.cn,
chenzh@ustc.edu.cn

Abstract. In this paper, we design a novel autonomous mobile robot which uses quantum sensors to detect faint signals and fulfills some learning tasks using quantum reinforcement learning (QRL) algorithms. In this robot, a multi-sensor system is designed with SQUID sensor and quantum Hall sensor, where quantum sensors coexist with traditional sensors. A novel QRL algorithm is applied and a simple simulation example demonstrates its validity.

1 Introduction

Autonomous mobile robot is a kind of robot system which perceives the environments through many kinds of sensor systems, and realizes free movement under complex and unknown environment to accomplish certain tasks. With the development of robot technologies, mobile robot has been widely applied in daily life and industrial area. But nowadays the performance of these robots still cannot meet peoples' requirements. There are mainly two reasons: (1) performances of sensors are too limited; (2) MCUs cannot compute fast enough. On the other hand, recently quantum technology, especially quantum information technology is rapidly developing. Quantum sensors using quantum effects have come into use and there are superconduction quantum interference device (SQUID) sensor [1] and quantum Hall sensor [2, 3]. At the same time, quantum algorithms are capable of parallel computing and some learning methods can also be improved by quantum parallelism.

Considering the merits of quantum technology [4], we design a novel autonomous mobile robot which detects faint signal with quantum sensors and accomplishes some learning tasks using quantum reinforcement learning (QRL) algorithms [5]. Compared with traditional mobile robot, it is composed of highly intelligent and complex engineering systems in which traditional algorithms are fused with quantum algorithms. This paper is organized as follows. The system structure of autonomous mobile robot is proposed in Sec.2. Section 3 designs a multi-sensor system based on quantum sensors. According to the task requirements of mobile robot, a QRL algorithm is used to this autonomous mobile robot and a simple simulation example demonstrates its validity in Sec.4. Section 5 gives conclusion and prospect.

2 The System Structure of Autonomous Mobile Robot

The overall hardware structure of the autonomous mobile robot system is shown in Fig.1, which is composed of two parts: Remote computer and mobile robot platform. The two parts are connected using bluetooth module, which works as wireless module.

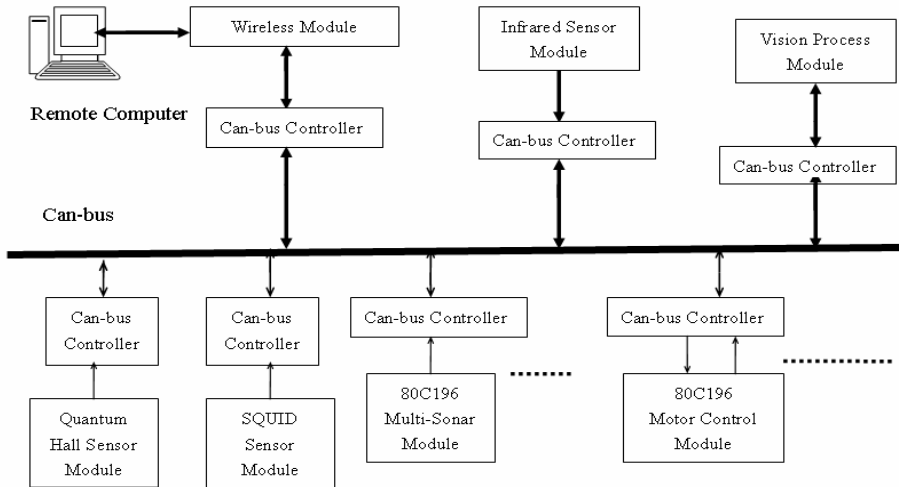


Fig. 1. Hardware structure of mobile robot system

The remote computer receives sensor information from mobile robot platform and sends behavior decisions via classical control algorithms and quantum algorithms. The mobile robot platform works as a carrier for sensors and actuators and is mainly composed of the following modules: (1) Can-bus modules; (2) Motor control modules; (3) Vision process module; (4) Multi-sonar sensor modules; (5) SQUID sensor modules; (6) Quantum Hall sensor modules; (7) Other modules for expanding the system.

The whole system is organized as a multi-agent system (MAS). Sensor agents implement the multi-sensor system based on quantum sensors, responsible for detecting environment and data preprocessing. Motor agent answers for bottom motion control, where wheel speed is calculated according to the kinematics models and sent to motion control modules. Obstacle avoiding agent is a kind of reactive agent and needs quick response to dynamic environment. Mapping agent and planning agent are both crucial deliberate agents and answer for the building of environment map and path planning.

3 Multi-sensor System Based on Quantum Sensors

Sensors are the most important approach for robot to detect the environment. The multi-sensor system used in the autonomous robot introduced above consists of several kinds of sensors, which are listed as Table 1.

Table 1. Sensors used in quantum-classical hybrid autonomous mobile robot

Sensors	Characteristics	Function
Quantum Hall Sensors	High sensitivity and very good temperature stability	Detecting faint electro-magnetic field
Superconduction Quantum Interference Device (SQUID) Sensors	Extremely sensitive magnetic sensor	Detecting super faint electric magnetic field
CCD Camera	The amount of information is large, under illumination environments	Sensing light that the object reflects in the environment
Multi-ultrasonic Sensors	Low price, accurate	Measuring the distance of the barriers
Infrared Sensors	Measurement is fast and accurate	Judging whether the object exists
Touch Switch	Simple, practical	Judging whether the robot collides with object

Among these sensors, SQUID sensors and quantum Hall sensors are the most special ones, so that the detecting system not only holds the common sensing ability, but also can measure super faint electromagnetic field, such as detecting cancer and tumor in biomedicine and sensing the faint signals of changing electromagnetic field in war industry.

- Quantum Hall Sensors: Have high sensitivity and very good temperature stability and can be used to detect faint electric magnetic field under different kinds circumstances
- SQUID Sensors: SQUID sensors are extremely sensitive magnetic sensors that use Josephson junctions. They can detect magnetic fields as small as 10^{-14} Tesla. SQUID sensors are commonly composed of SQUID, gradiometer and superconductive magnetic field. And SQUID sensors excel in high sensitivity.

4 Behavior Planning Based on Quantum Reinforcement Learning

When applied to real mobile robots algorithms should be highly intelligent, which requires MCUs with powerful computing ability. But traditional computers are limited by the integration of chip. One possible approach is to fall back on quantum computation theory and fuse quantum algorithms into some learning methods.

4.1 Quantum Parallel Computation

In quantum information technology, information unit is represented with quantum state (also called quantum bit or qubit), and qubit is an arbitrary superposition state of two-state quantum system:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1)$$

where α and β are complex coefficients and satisfy $|\alpha|^2 + |\beta|^2 = 1$. $|0\rangle$ and $|1\rangle$ are two orthogonal states (also called basis vectors of quantum state $|\psi\rangle$), and they correspond to logic states 0 and 1. $|\alpha|^2$ represents the occurrence probability of $|0\rangle$ when this qubit is measured, and $|\beta|^2$ is the probability of obtaining result $|1\rangle$. The value of classical bit is either Boolean value 0 or value 1, but qubit can be prepared in the coherent superposition state of 0 and 1, i.e. qubit can simultaneously store 0 and 1, so quantum computation has parallel process ability.

Consider a n -qubit cluster and it lies in the following superposition state:

$$|\psi\rangle = \sum_{x=00\dots0}^{11\dots1} C_x |x\rangle, \quad \sum_{x=00\dots0}^{11\dots1} |C_x|^2 = 1 \tag{2}$$

where the length of x is n , C_x is complex coefficients and $|C_x|^2$ represents occurrence probability of $|x\rangle$ measuring state $|\psi\rangle$. $|x\rangle$ has 2^n values, so the superposition state can be looked upon as the superposition state of all integers from 0 to $2^n - 1$. The computing U to function $f(x)$ can give out:

$$U \sum_{x=00\dots0}^{11\dots1} C_x |x,0\rangle = \sum_{x=00\dots0}^{11\dots1} C_x U |x,0\rangle = \sum_{x=00\dots0}^{11\dots1} C_x |x, f(x)\rangle \tag{3}$$

Thus a n -qubit cluster can simultaneously process 2^n states and the quantum parallelism provides an exponential-scale computation space in the n -qubit linear physical space. Therefore quantum computation can effectively increase the computing speed of classical function, and so it can be fused into learning algorithms and be used to autonomous mobile robot to effectively speed up the solution of question.

4.2 Quantum Reinforcement Learning for Autonomous Mobile Robot

The essence of robot learning and planning is to deal with state-action pair $\{State(t), Action(t)\}$. Here the widely used reinforcement learning methods [6] are introduced as an example to show how to improve traditional methods using quantum algorithm.

Firstly, let N_s and N_a be the number of states and actions, then choose numbers m and n , which are characterized by the following inequations:

$$N_s \leq 2^m \leq 2N_s, \quad N_a \leq 2^n \leq 2N_a \tag{4}$$

And use m and n qubits to represent state set $S = \{s\}$ and action set $A = \{a\}$:

$$s : \left[\begin{array}{c|c|c} a_1 & a_2 & \dots & a_n \\ \hline b_1 & b_2 & & b_n \end{array} \right], \text{ where } |a_i|^2 + |b_i|^2 = 1, \quad i = 1, 2, \dots, m$$

$$a : \left[\begin{array}{c|c|c} \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \hline \beta_1 & \beta_2 & & \beta_n \end{array} \right], \text{ where } |\alpha_i|^2 + |\beta_i|^2 = 1, \quad i = 1, 2, \dots, n$$

According to Ref. [5], the procedural form of QRL may be described as follows.

Initialize $|s^{(m)}\rangle = \sum_{s=00\dots0}^{11\dots1} C_s |s\rangle$, $f(s) = |a_s^n\rangle = \sum_{a=00\dots0}^{11\dots1} C_a |a\rangle$ and $V(s)$

Repeat (for each episode)

For all states $|s^{(m)}\rangle = \sum_{s=00\dots0}^{11\dots1} C_s |s\rangle$:

- (1) Observe $f(s)$ and get $|a\rangle$;
- (2) Take action $|a\rangle$, observe next state $|s^{(m)}\rangle$, reward r

Then update: $V(s) \leftarrow V(s) + \alpha(r + V(s') - V(s))$

$$C_a \leftarrow e^{\lambda(r+V(s'))} C_a$$

Until for all states $|\Delta V(s)| \leq \epsilon$.

Now let us use QRL to this robot. Consider the typical rooms with corridor example, gridworld environment of four rooms and surrounding corridors as shown in Fig. 2. The task of the algorithms is to find an optimal policy which will let the robot move from S to G with minimized cost. In this experiment, the initial state and the goal is cell(4,4) and cell(8,8), respectively. The learning performance for QRL is plotted in Fig. 3. We observe that given a proper stepsize (alpha < 0.10) this algorithm learns extraordinarily fast at the beginning phase, and then steadily converges to the optimal policy that costs 25 steps to the goal G . The results show that QRL algorithm excels other RL algorithms in the action selecting policy using probability.

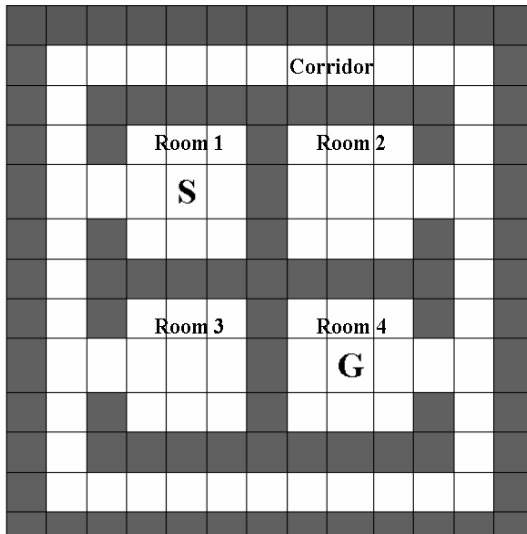


Fig. 2. The example of rooms with corridor is a gridworld environment with cell-to-cell actions (up, down, left and right). The labels S and G indicate the initial state and the goal in the simulation experiment described in the text.

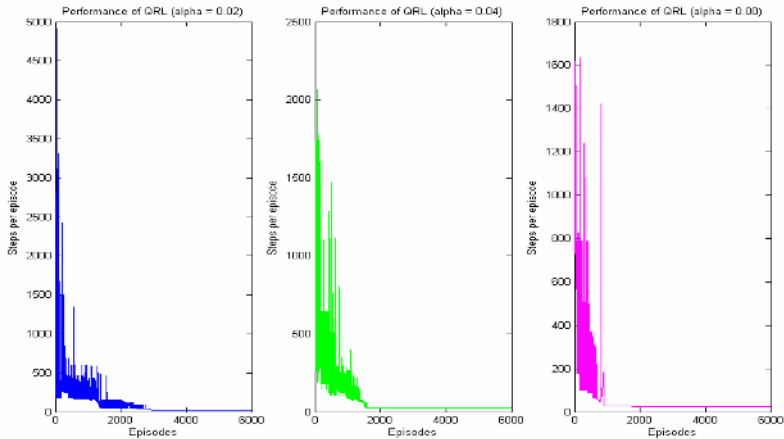


Fig. 3. Performance of QRL in the example of rooms with corridor

5 Conclusion

Considering two bottlenecks: performances of sensors and computation speed of MCUs, we design a novel quantum-classical hybrid autonomous mobile robot. Compared with traditional mobile robot, the robot effectively overcomes the limitation of existing sensors' performance and is able to meet real-time requirement for complex problems, which will make it competent for applications in the fields of military affairs, national defense, aviation, biomedicine, safety engineering, etc.

References

1. Morisawa, J., Otaka, M., Kodama, M., Kato, T., Suzuki, S.: Detection of Intergranular Cracking Susceptibility Due to Hydrogen in Irradiated Austenitic Stainless Steel with A Superconducting Quantum Interference Device (SQUID) Sensor. *J. Nucl. Mater.* 302 (2002) 66–71
2. Behet, M., Bekaert, J., De Boeck, J., Borghs, G.: InAs/Al_{0.2}Ga_{0.8}Sb Quantum Well Hall Effect Sensors. *Sensors and Actuators.* 81 (2000) 13–17
3. Behet, M., Das, J., De Boeck, J., Borghs, G.: InAs/(Al, Ga)Sb Quantum Well Structure for Magnetic Sensors. *IEEE Trans. Magn.* 34 (1998) 1300–1302
4. Nielsen, M.A., Chuang, I.L.: *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, England (2000)
5. Dong, D. Y., Chen, C.L., Chen, Z.H.: Quantum Reinforcement Learning. In: Wang, L., Chen, K., Ong, Y.S. (eds.): *Advances in Natural Computation*. Lecture Notes in Computer Science, Vol. 3611. Springer-Verlag, GmbH (2005) 686–689
6. Sutton, R., Barto, A.G.: *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA (1998)

A MPC and Genetic Algorithm Based Approach for Multiple UAVs Cooperative Search

Jing Tian¹, Yanxing Zheng², Huayong Zhu¹, and Lincheng Shen¹

¹ School of Mechanical Engineering and Automation,
National University of Defense Technology, China
jingtian@nudt.edu.cn

² School of Computer Science, National University of Defense Technology

Abstract. This paper focuses on the problem of cooperative search using a team of Unmanned Aerial vehicles (UAVs). The objective is to visit as many unknown area as possible, while avoiding collision. We present an approach which combines model predictive control(MPC) theory with genetic algorithm(GA) to solve this problem. First, the team of UAVs is modelled as a controlled system, and its next state is predicated by MPC theory. According to the predicted state, we then establish an optimization problem. By use of GA, we get the solution of the optimization problem and take it as the input of the controlled system. Simulation results demonstrate the feasibility of our algorithm.

1 Introduction

Using a fleet of Unmanned Aerial vehicles (UAVs) equipped with special sensors to search an interesting area is one of the excellent applications of UAVs. Many literatures have been proposed for the cooperative search problem using multi-UAVs. Flint[1] proposed a dynamic programming algorithm for it. Beard[2] also addressed the problem by dynamic programming algorithm. k -shortest path algorithm was use by Sujit [3] as search algorithm for multiple UAVs searching based on the uncertainty map of an unknown region.

This paper combines Genetic Algorithm (GA)[4] with Model predictive control(MPC)[5] to provide an effective algorithm for cooperative search. The search area is partitioned to a collection of hexagonal cells, similar with that proposed in[3]. We model the team of UAVs as a controlled system, the state of which is the position of UAVs. MPC theory is then used to predicate the next state of the controlled system, according to which, we define the objective function which maximize the searching area of all the UAVs by choosing the best position of UAVs in the next time step. By use of GA, we get the solution of the optimization problem and take it as the approximate optimal input of the controlled system. Compared with mixed-integer linear programming(MILP) which is often used in the optimization process of MPC [6], [7], [8], [9], GA is a global optimal optimization technique and can solve both linear and nonlinear optimization problems and provides more flexibility. Our approach combines the prediction ability of MPC with the optimization ability of GA, avoiding the search process running into the local optimality.

2 Hexagonal Partition of Search Area

Firstly, we partition the given two-dimensional rectangle search area into a collection of identical hexagonal cells as Sujit did in[3]. Hexagonal cell offers the flexibility to UAV to move in six uniformly distributed directions at each time step and reach a neighboring cell while expending the same amount of energy.

For the given rectangle search area, we partition it to $(M \times N)$ hexagonal cells and index each cell with a two-dimensional coordinate $(i, j), i \in \{1, 2, \dots, M\}, j \in \{1, 2, \dots, N\}$. $(1, 1)$ represents the left-down cell and (M, N) represents the right-top cell. (i, j) represents the cell located in the i th line and j th column.

Assume that each hexagonal cell can be covered by an UAV located in its center and large enough to allow the UAV maneuver inside it. In each time step, an UAV can move from the center of one cell to that of its neighboring cell. To avoid collision, there are no more than one UAV in one cell simultaneously.

3 Problem Formulation

3.1 The State-Space Model

Assume that Nv UAVs to perform the cooperative search mission. Each UAV is modelled as a point mass moving in two dimensions with limited speed. Let $x_i(k) = (p, q), p \in \{1, \dots, M\}, q \in \{1, \dots, N\}$ denotes the index of the hexagonal cell which the i th UAV locates in at instant k . $(x_i(0), x_i(1), \dots, x_i(T))$ is trajectory of the i th UAV. The system state at k is $X(k) = \{x_1(k), x_2(k), \dots, x_{Nv}(k)\}$.

The state-space model of the system has the following description

$$X(k + 1) = X(k) + u(k) \tag{1}$$

Where $u(k) = (u_1(k), \dots, u_{Nv}(k))$ is the system control input at k . $u_i(k) \in U = U_{odd} \cup U_{even}$ changes the position of the i th UAV from current cell to a neighboring cell. For $x_i(k) = (p, q)$, if p is odd, $u_i(k) \in U_{odd}$, else $u_i(k) \in U_{even}$, where $U_{odd} = \{(0, 1), (1, 0), (1, -1), (0, -1), (-1, -1), (-1, 0)\}$, $U_{even} = \{(0, 1), (1, 1), (1, 0), (0, -1), (-1, 0), (-1, 1)\}$. Fig1 illustrates the correspondence between control input and the neighboring cell of current cell.

The condition of collision avoidance is expressed as

$$x_i(k) \neq x_j(k), i \neq j, \text{ for } i, j \in \{1, \dots, Nv\} \tag{2}$$

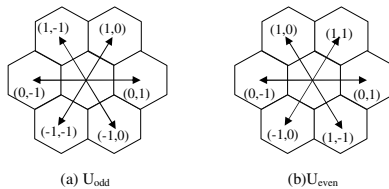


Fig. 1. Odd and Even control set

It is easy to predict system future outputs according to the state-space model.

$$X(k + j|k) = X(k + j - 1|k) + u(k + j - 1|k), j = 1, 2, \dots, H \quad (3)$$

Where H is the planning horizon. $X(k + j|k)$ indicates the prediction of the system state at instant $(k + j)$ calculated at k .

3.2 Optimization Problem

We use a detecting value $V_{ij}(k) \in [0, 1]$ to denote to what extent we know about the cell (i, j) at instant k . $V_{ij}(k) = 0$ means that we know nothing about the situation in it. If $x_i(k) = (p, q)$, the cell (p, q) is visited by the i th UAV and $V_{pq}(k) = 1$. A forgetting factor $\beta \in [0, 1]$ is employed to reflect the dynamic search area. Let matrix $V(k) = [V_{ij}(k)]_{M \times N}$ denotes the detecting values of all the cells at instant k , where

$$V_{ij}(k) = \begin{cases} 1 & \text{if cell}(i, j) \text{ is visited by a vehicle at } k \\ \beta V_{ij}(k - 1) & \text{otherwise} \end{cases} \quad (4)$$

We design a cooperative controller which generates a set of control inputs for each vehicles. At k , the controller operates by solving an optimization problem P_k , whose solution is $(u(k|k), u(k + 1|k), \dots, u(k + H|k))$, and only $u^*(k) = u(k|k)$ is applied to the system. Define

$$\delta_{pq}(x_i(k)) = \begin{cases} 1 & \text{if } x_i(k) = (p, q) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Now we can present the optimization problem P_k formulated at instant k as

$$\max J(k) = \sum_{i=1}^{Nv} \sum_{p=1}^M \sum_{q=1}^N (1 - \delta_{pq}(x_i(k + H)))\beta V_{pq}(k + H - 1) + \delta_{pq}(x_i(k + H)) \quad (6)$$

Where $J(k)$ reflects the detecting values of all the cells. By optimizing P_k constricted by equation(2), we get the approximate optimal control input $u^*(k)$.

4 MPC and GA Based Cooperative Searching Algorithm

Our goal is to find the approximate optimal $u^*(k)$ which maximize $J(k)$. We use GA as the optimization tool, which is a powerful search algorithm based on the mechanics of natural selection and can offer more flexibility for optimization. Fig2(a) shows a chromosome which is composed of $H \times Nv$ real numbers between 1 and 6. The i th line represent the control input $u(k + i - 1|k)$. According to fig2(b) and fig1, the correspondent control input can be figured out.

In GA, The better the solution encoded by a chromosome, the higher the fitness. The fitness function is as follows.

$$F = \begin{cases} 0 & \text{if } (X(k + 1 + j|k)) \text{violates the constraints.} \\ J(k) & \text{otherwise} \end{cases} \quad (7)$$

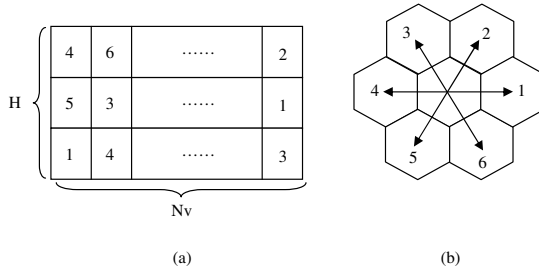


Fig. 2. Chromosome coding

```

(1) k=0
(2) while k < T
(3)   according to X(k), update V
(4)   using GA to solve optimization problem P_k and get u*(k)
(5)   X(k + 1) = X(k) + u*(k)
(6)   k=k+1
    
```

Fig. 3. MPC and GA based cooperative searching algorithm

We use the roulette-wheel selection method in the reproduction process and single-point crossover in the crossover process. Given $M \times N$ hexagonal cells and initial positions of Nv UAVs, each UAV can fly at most T time steps constrained by the fuel. Fig3 presents the cooperative searching algorithm.

5 Simulation Results

For the purpose of simulation, we consider a search area which is partitioned to 10×10 hexagonal cells and 5 UAVs is employed to perform the search. In the simulations, the initial position of the UAVs is generated randomly. Each UAV can fly $T = 40$ time steps constrained by the fuel. The forgetting factor $\beta = 0.9$. The predictive horizon is set to $H = 4$. In the following, we use the detecting value of a hexagonal cell to represent its gray level. The more we know about it, the whiter it shows.

Fig4(a) shows the final search results after 40 time steps, where the whiter the cell, the more information we know about it. Fig4(b) shows the trajectories of five UAVs. We can see that no cell is black now, which indicates that all the cells have been visited at least once.

Finally, in order to fully illustrate the efficiency of our algorithm, we compare its performance with a random search and a greedy strategy. In the random strategy, the vehicle select its next cell for search by generating a uniformly distributed random number which indicates the six different neighboring cells.

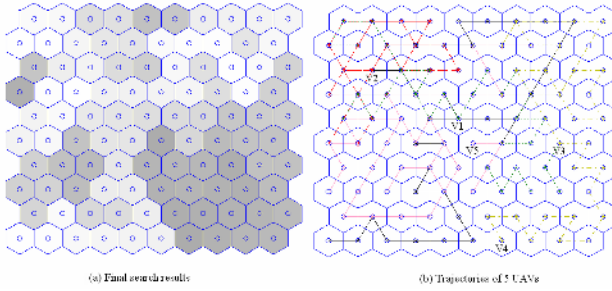


Fig. 4. Cooperative search results

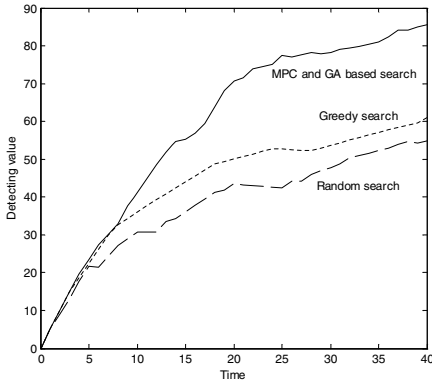


Fig. 5. Cooperative search results

While in the greedy search strategy, the vehicle selects the next cell with the smallest detecting value. The performance of the algorithms is measured by the sum of the detecting values of all the cells. Fig5 shows the increment of detecting values with increasing time for the three search algorithms. It clearly shows the superiority of MPC and GA-based search algorithm over the other two algorithms. This superiority is mainly comes from the prediction ability of MPC and flexibility of GA.

6 Conclusion

In this paper, a MPC and GA based algorithm is presented for multiple UAVs cooperatively searching an unknown area. Our algorithm combines the flexibility of GA with the prediction ability of MPC and provides an effective means for cooperative search problem. First, a hexagonal cell based partition of the search area is introduced. Then, we construct the system state-space model and define the objective function. At each instant, GA is employed to optimize the objective function and get approximate optimal control input for the system. Compared

with MILP algorithm, GA has powerful optimization ability because it is a global optimal optimization technique and can solve both linear and nonlinear optimization problems. Finally, the simulation results show the efficiency and superiority of our algorithm over the random search and greedy strategy.

References

1. Flint Matthew, Fernández-Gaucherand E., Polycarpou M.: Cooperative Control for UAV's Searching Risky Environment for Targets. *IEEE Conference on Decision and Control*. (2003) 3567–3572
2. Beard Randak W., McLain Timothy W.: Multiple UAV Cooperative Search under Collision Avoidance and Limited Range Communication Constraints. *IEEE Conference on Decision and Control*. (2003) 25–30
3. Sujit P.B., Ghose D.: Search Using Multiple UAVs with Flight Time Constrains. *IEEE Transaction On Aerospace and Electronic Systems*. Vol.40, No.2. (2004) 491–509
4. Goldberg D.E.: Genetic algorithm in search optimization and machine learning, Addison Wesley, New York, 1988
5. Sjanic Z.: On-line mission planning based on Model Predictive Control. Master thesis, Linköping University, December 2001
6. Castañón David A., Wohletz Jerry M.: Model Predictive Control for Dynamic Unreliable Resource Allocation. *IEEE Conference on Decision and Control*. (2002) 3754–3759
7. Richards Arthur, How Jonathan P.: Model Predictive Control of Vehicle Maneuvers with Guaranteed Completion Time and Robust Feasibility. *American Control Conference*.(2003) 4034–4040
8. Richards Arthur, How Jonathan P.: Decentralized Model Predictive Control of Co-operating UAVs. *IEEE Conference on Decision and Control*.(2004)
9. Bellingham John, Richards Archur, How Jonathan P.: Receding Horizon Control of Autonomous Aerial Vehicles. *American Control Conference*. (2002) 3741–3746

Self-organization Evolution of Supply Networks: System Modeling and Simulation Based on Multi-agent

Gang Li^{1,2}, Linyan Sun¹, Ping Ji², and Haiquan Li³

¹ The School of Management, Xi'an Jiaotong University, Xi'an 710049, China

² Department of Industrial & Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

³ The School of Computer, Xi'an Shiyou University, Xi'an 710065, China

lee_rich@163.com, lysun@mail.xjtu.edu.cn,
mfpji@inet.polyu.edu.hk, hqli@xsyu.edu.cn

Abstract. This paper demonstrates the self-organization evolution of distributed Supply Networks (SNs) based on fitness landscape theory. The environment and the internal mechanism are the origin of SN evolution. The SN emerges from the local interaction of the firms to fulfill the stochastic demands. The collaboration among firms is path dependence. The evolution of a SN is self-reinforcement and sensitive to initial conditions, which may lead to multiple equilibrium state and chaos. The evolution result is non-deterministic and can not be predicted precisely. The long-term strategy is better than short-term strategy for a firm in SN collaboration to adapt to the environment.

1 Introduction

A Supply Network (SN) is a network of independent firms collaborating a series of interrelated business processes as the procurement, manufacturing and distribution to serve the final market. A firm collaborates with others in order to enhance its operational efficiency, reactivity, profitability and competence. Evidence in most industries reveals that a SN exhibits a wide variety. However, little is known how a SN is born, evolves and adapts to the environment. With the dynamics of the environment and the complexity of the SN itself, managerial efforts to control the evolution of the SN have often led to frustration. The unexpected SN emerges and evolves over time [1].

The answers to these questions are not known and limited research has been conducted on the SN evolution to understand and analyze them as a dynamic, emergent network with decentralized decision-making firms. Historically, most of the related research viewed a SN as an integrated and static network. In recently years, supply chain modeling, simulation based on distributed artificial intelligence, and complexity science have been used to study the adaptability of SNs [2]. The principal limitations of these studies are that they are unable to model structural and behavioral dynamics of a SN, which are essential for understanding the evolution observed in these networks. Due to the evolving nature of SNs, we need an approach which is rich enough to capture the

dynamic behavior, and sufficiently flexible to allow for the evolution of SNs [3]. The self-organization system modeling and simulation is such a way. Inspired with the ideas of self-organization of SNs proposed by Choi [1] and Kogut [5], we develop a self-organization evolution model based on fitness landscape theory and simulate the model based on multi-agent technology to show the evolution of a SN.

2 Modeling the Self-organization Evolution of SN

2.1 The Conception Model

The structure of a SN is dynamic. It can be modeled by a bi-directional graph. The nodes represent firms in a SN, while the edges between firms (nodes) represent the collaboration relationship of the nodes. In the edges of the graph, firms exchange materials, information and money. The weight of each edge represents the collaboration preference of the nodes. Firms reside in an environment and collaborate each others to fulfill the stochastic demand emerging from the environment. With the changes of the environment, the SN collaboration pattern is changing, namely, a SN evolves.

Fitness landscape theory is introduced to evaluate firms' adaptability to the dynamic environment. In management literature, McCarthy proposed: "the manufacturing fitness is the capability to survive by demonstrating adaptability and durability to the changing environment" [5]. The strategic management view of fitness is concerned as the balance between environmental expectations placed on the firm (costs, delivery, quality, innovation, customization, etc.) with the resources and capacity available in the firm. The different combination of the attributes (capacity, cost, delivery, etc.) of a firm represents different fitness. The goal of firm's collaboration is to improve their fitness to adapt to the environment. Due to market competition, the higher the performance of a firm is, the higher the probability of the firm to improve its fitness. The evolution of a firm's fitness is abstracted to the cost and the reward that the firm makes in the SN collaboration. It is shown in (1):

$$fitness_i(t+1) = fitness_i(t) + \Delta fitness_i(t) = fitness_i(t) + \Delta f(cost_i(t), reward_i(t)) - \Delta f_i(t) \quad (1)$$

where $cost_i(t)$ is the cost to fulfill the demand, and $reward_i(t)$ is the return that the firm can get in the demand cycle at time t . If a firm can not fulfill the orders, $reward_i(t)$ is negative. $\Delta f_i(t)$ is the degradation of fitness with time. The higher the uncertainty of the environment is, the higher $\Delta f_i(t)$ is. With the changes of the fitness, the probability for a firm to get success in the market competition also changes.

2.2 The Process Model

The process model discloses how a SN processes input and makes the output. There are two inputs of a SN: *the outside environment* and *firms' internal mechanisms*. The environment can be modeled as three aspects: 1) a fitness threshold which is the critical value for a firm to survive in the environment; 2) the micro operation condition which represents the demand, delivery, cost, price, flexibility, etc.; 3) the macro market conditions including industry structure, policy, culture, etc. Another input, firms' internal

mechanisms, defines a firm's behavior schema. The internal mechanism of a firm is controlled by: 1) Goal; 2) Strategy; 3) Operation rules; 4) Fitness. A firm's behavior is governed by its internal mechanism. The collaboration among firms is controlled by the environment and their internal mechanisms. During their collaboration, firms learn from each other, and adjust their internal mechanisms to adapt to the environment. The **output** of a SN is firms' dynamical behavior in the micro-level and the self-organization evolution of the SN in the macro-level.

2.3 The Self-organization Evolution Model

The firm's behavior model: A firm can be represented as a five-tuple which includes Goals (O), Collaboration strategy (S), Operation constraints (C) such as capacity, cost and resources, Operation rules (R_i) which are in accordance with the strategy, and the firm's fitness ($F, F \in [0,1]$). That is,

$$n_i = \langle O, C, S, R_i, F \rangle \quad (2)$$

The macro evolution model: A SN emerges from the dynamic interaction between firms and evolves with the environment. The macro evolution model of a SN is:

$$SN = f(\text{Environment}, \text{firm}_1, \dots, \text{firm}_m) = f_{so}(\text{Environment}, \text{firm}_1, \dots, \text{firm}_m) \quad (3)$$

$$\text{Environment} = \langle MOC, M, F_c, FEv \rangle, \text{Environment} \neq \text{Null} \quad (4)$$

where MOC is the micro operation condition such as price, lead-time, and demand. $F_c \in [0,1]$ is the critical fitness value. When a firm's fitness is lower than F_c , the environment kills it, namely, this is a natural selection. M is the market setting, represented as a multi-tuple, including the macro conditions of the market such as government policy, law, etc. The SN structure is a bi-directional graph, which is modeled as a three-tuple with $V(G)$ representing the nodes in the graph, $Edge(G)$ representing the edges between nodes, that is, the collaboration relationship between firms, and $Preference(G)$ representing the set of edges' weights, that is, the collaboration preference of firms. In other words,

$$G = \langle V(G), Edge(G), Preference(G) \rangle. \quad (5)$$

3 Simulation of the Evolution of SN

3.1 The Simulation Model

The simulation model is based on multi-agent. Each firm has three kinds of strategy: long-term, short-term and price-priority collaboration strategy. In each demand cycle, a stochastic demand is created. Firms collaborate with each other to fulfill the demand. There are two steps of competition. 1) The firm with the highest fitness gets the global demand. 2) Firms compete to get the subcontract demand. The operation rules of a firm are: 1) Objective: increase its fitness; 2) The firm manufactures the product to meet the demand subject to its capacity and subcontracts the surplus to other firms; 3) The

bidding price is stochastically decided by a firm’s manufacturing cost; 4) The selection of a subcontractor is decided by the tenderer’s strategy. At the end of each demand cycle, the performance of the firm is evaluated and its fitness is updated.

3.2 Simulation Experiments and Results

(1) The effect of the environment to the SN evolution: In experiment 1, there are 8 firms (numbered from 0 to 7) with cost-priority strategy. The environment is set as high demand volatility. The attributes of 8 firms (fitness, capacity and cost) are created randomly at the beginning of the simulation. Some characteristics of the self-organization system can be found from the simulation results.

1.1) Emergence: In Fig. 1, a SN with linear structure emerges from the interactions of 8 firms to fulfill the stochastic demand. Firms 5, 0, 1, and 2 form a relative stable win-win collaboration structure. Fig. 2 is the evolution of the 8 firms’ fitness. The SN changes with the changes of firms’ attributes (capacity, price, and cost) and the demand. When the capacities of the firms match the demand created by the environment, the firms adapt to it and there emerges a win-win SN structure.

1.2) Path dependence: In Fig. 3, we find that firms 1, 4, 5 and 6 form a relative stable collaboration structure in most of the demand cycles. While firms 2, 6, 8 also form a relative collaboration set. Firms’ fitness of the two structures is improved in their collaboration. The partners achieve a win-win result and get increasing total profit gradually. This phenomenon indicates that the SN evolution is path dependence.

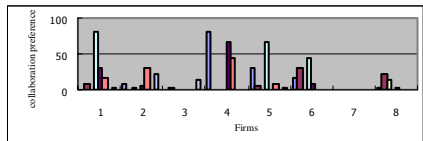
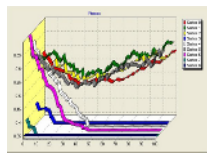
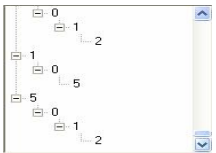


Fig. 1. Emergence

Fig. 2. Fitness Evolution

Fig. 3. Path Dependence

1.3) Chaos: We find four kinds of fitness evolution tracks as shown in Fig. 4, 5, 6, and 7, respectively. They are dramatically different. The 8 firms with slight difference evolve in dramatically different ways even in the same environment with the same internal mechanism. The evolution is highly sensitive to the initial conditions and difficult to be predicted. This indicates there is chaos in the SN evolution.

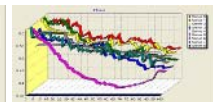
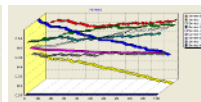
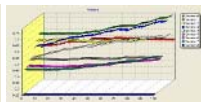
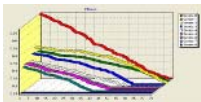


Fig. 4. Chaos (a)

Fig. 5. Chaos (b)

Fig. 6. Chaos (c)

Fig. 7. Chaos (d)

(2) The Effect of Firms’ Strategy to the SN Evolution: In experiment 2, we test the evolution of the 8 firms with the long-term strategy or short-term strategy under a highly uncertain demand environment. We find that once it emerges a superior

equilibrium state, the collaboration pattern of the firms will be fixed, regardless of the long-term or short-term collaboration strategy. Besides, the firms with the long-term collaboration strategy can improve their fitness greater than the firms with the short-term strategy in an environment with high uncertainty.

4 Conclusions

This article studies the self-organization evolution of SNs. The main findings are fourfold. Firstly, the environment and the internal mechanisms are the origin of the SN evolution. The SN emerges from firms' dynamic interaction and evolves in a self-organization way. Secondly, the collaboration of firms is path dependence. The evolution of the SN is self-reinforcement and it may emerge multiple equilibrium state. Some of them are in superior equilibrium and of system efficiency, and some are in inferior equilibrium and of system inefficiency. Thirdly, the evolution of a SN is highly sensitive to initial conditions. The result is non-deterministic and can not be predicted precisely. There is chaos in the evolution. And the fourth is that long-term strategy is better than short-term strategy in SN collaboration. However, the SN keeps in a relatively stable structure when there emerges a superior equilibrium, no matter the firms takes a long-term or a short-term strategy.

Acknowledgement

This paper was partially supported by the NSFC (Project No. 70433003) and by The Hong Kong Polytechnic University (Project No. A-PG64).

References

1. Y.T Choi, K.J Dooley, M Rungtusanatham: Supply Networks and Complex Adaptive Systems: Control Versus Emergence. *Journal of Operation Management*. 19 (2001) 351-366
2. Jayashankar M. Swaminathan: Modeling Supply Chain Dynamics: A Multi-agent Approach. *Decision Sciences*. 29 (1998) 607-632
3. Surya Dev Pathak, Gautam Biswas: A Multi-paradigm Simulator for Simulation Complex Adaptive Supply Chain Networks. *Proceedings of the 2003 Winter Simulation Conference*. IEEE. Piscataway, New Jersey. (2003) 809~816.
4. Kogut: The Network As Knowledge: Generative Rules and the Emergence of Structure. *Strategic Management Journal*. 21 (2000) 405~425
5. Ian P. McCarthy. Manufacturing Strategy: Understanding the Fitness Landscape. *International Journal of Operation & Production Management*. 24 (2004) 124~150

Modeling and Analysis of Multi-agent Systems Based on π -Calculus

Fenglei Liu¹, Zhenhua Yu^{2,3}, and Yuanli Cai³

¹ School of Electro-Mechanical Engineering,
Xidian University, Xi'an 710071, China

² Key Laboratory of Opto-Electronic Technology and Intelligent Control,
(Lanzhou Jiaotong University), Ministry of Education, China
zhenhua_yu@163.com

³ School of Electronic and Information Engineering,
Xi'an Jiaotong University, Xi'an 710049, China

Abstract. Dynamic architecture of multi-agent systems (MAS) is very important for the critical systems. As the existing formal specifications cannot describe the dynamic architecture of MAS, while π -calculus is specially suited for the description and analysis of concurrent systems with dynamic or evolving topology, a formal approach using π -calculus is presented to describe MAS. π -calculus can describe the interactions among agents and permit their analysis for some key properties, e.g. deadlock, bisimilarity. By constructing a MAS model in electronic commerce, the modeling process using π -calculus are illustrated.

1 Introduction

Multi-agent systems (MAS) have been recognized as a main aspect of the distributed artificial intelligence and predicted to be the future mainstream computing paradigm [1], [2].

MAS are adaptive and dynamic systems where agents may be added or deleted at run-time, and the agent behaviors and interactions among agents may vary dynamically [3], so MAS architecture can evolve during the execution of the system. For MAS applications, support for dynamism is important in the case of certain safety- and mission-critical systems, such as air traffic control, high availability public information systems. Shutting down and restarting such systems for upgrades may incur unacceptable delays, increased cost, and risk [11]. Consequently, there is a pressing need for a formal specification to support the design and implementation of MAS, and ensure the developed systems to be robust, reliable, verifiable, and efficient [5]. So far, there have existed several typical formal specifications for MAS, e.g. dMARS [6], DESIRE [4], Gaia [7], MaSE [8], Agent-based G-net [5], AUML [9], and Tropos [10]. Despite the important contribution of these formalisms to a solid underlying foundation for MAS, most formal specifications are not oriented for software engineering. Most importantly, the existing formal specifications cannot describe the dynamic architecture of MAS from the structure and behavior, which is a large challenge for formal specifications.

In this paper, to support the development of correct and robust dynamic MAS, a novel formal specification based on π -calculus [12] is proposed. The reason using π -calculus to model dynamic MAS is that π -calculus is a process algebra specially suited for the description and analysis of concurrent systems with dynamic or evolving topology, and furthermore, π -calculus supports formal analysis of MAS in a variety of well-established techniques. To illustrate the favorable representation capability of π -calculus, an example of dynamic multi-agent systems in electronic commerce is provided.

2 Modeling and Analyzing Multi-agent Systems Using π -Calculus

The π -calculus [12] is an extension of the process algebra CCS in order to allow dynamic reconfiguration of systems. The modeling entities are *names* and *processes*. Mobility is achieved in π -calculus by the transmission of channel names as arguments or objects of messages [13], which allows an easy and effective re-configuration of the system. The π -calculus can be varied in many ways. There are many useful subcalculi, e.g. the polyadic π -calculus [14]. In this paper, the polyadic π -calculus is adopted as the modeling tool.

2.1 Modeling Multi-agent Systems Using π -Calculus

According to the characteristics of π -calculus, π -calculus provides an appropriate context for specifying dynamic and concurrent MAS. π -calculus is used to describe the internal behaviors of the agents, and the agents respective interfaces with the other agents. The agent can be defined as a *process* or a concurrent *process* consisting of some subprocesses. The subprocess can represent the behavior of the agent, and there exists some channels for communication between agents.

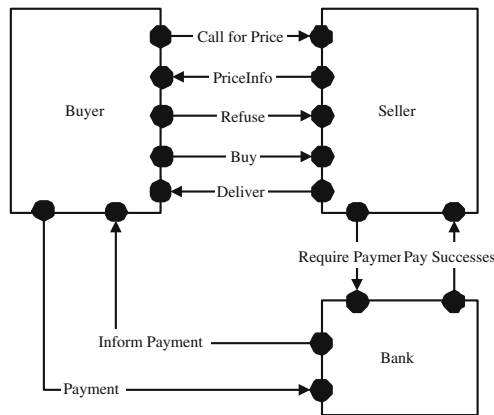


Fig. 1. The MAS architecture in electronic commerce

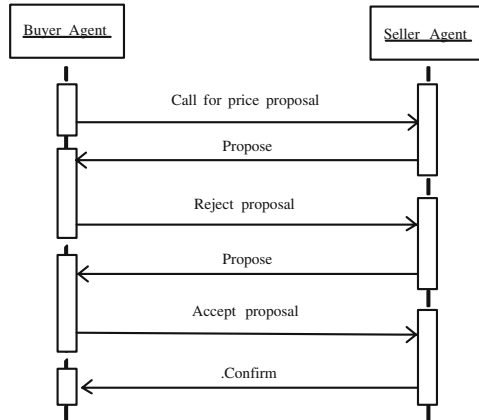


Fig. 2. An example of price-negotiation contract net protocol

In this paper, a multi-agent system in an electronic commerce is used to show how to model MAS by π -calculus. The buyer agents and seller agents negotiate price, and finally the buyer agents determine whether to buy or not. The MAS architecture is shown in Fig. 1, where some constraints are omitted. The MAS is composed of three functional agents (a buyer agents, a seller agent, and a bank agent) bargaining for goods. The interaction protocol among agents is contract net protocol. Fig. 2 depicts an example of price-negotiation contract net protocol expressed as an AUML sequence diagram between a buyer agent and a seller agent.

The buyer agent first sends a call for price (CFP) proposal to the seller agent, then the seller agent sends the price information after it receives the CFP. Upon the arrival of the price proposals, the buyer agent thinks whether the price proposals are acceptable or not. If the proposals are rejected, the buyer agent will negotiate again. If the proposals are accepted, the buyer agent will buy the goods and the seller agent send the pay request via the bank agent. If the buyer

Table 1. The messages among the buyer agent, seller agent, and bank agent

Message	Description
Call for Price (CFP)	The buyer agent sends call for price proposal.
PriceInfo	The seller agent returns the price information of the goods.
Refuse	The buyer agent refuses the price information.
Buy	The buyer agent sends the request of buy.
Deliver	The seller agent delivers the goods to the buyer agent.
Require Payment (ReqPay)	The seller agent inform the bank agent the buyer agent payment.
Pay Successes (PaySuc)	The bank agent inform the seller agent that the buyer agent has paid.
Inform Payment (InfPay)	The bank agent inform payment of the buyer agent .
Payment	The buyer agent pay the bill to the bank agent.

agent sends the payment, the seller agent will deliver goods to the buyer agent. The messages among these agents are shown in Table 1.

In Fig. 1, the buyer, seller, and bank agent can be treated as a process in π -calculus, and the three agents communicate each other via the channel u , v , w . Then the MAS architecture can be represented by π -calculus. By referring to [15], the buyer, seller, and bank agent are described using π -calculus as follows.

$$\begin{aligned} & BuyerAgent(u, w, CFP, PriceInfo, Refuse, Buy, Deliver, InfPay, \\ & Payment) = \bar{u}\langle CFP \rangle.u(msg).[msg = PriceInfo](\bar{u}\langle Refuse \rangle.BuyerAgent \\ & + \bar{u}\langle Buy \rangle.w(msg1).[msg1 = InfPay]\bar{w}\langle Payment \rangle.u(msg2). \\ & [msg2 = Deliver]BuyerAgent) \end{aligned}$$

$$\begin{aligned} & SellerAgent(u, v, CFP, PriceInfo, Refuse, Buy, Deliver, ReqPay, \\ & PaySuc) = u(msg).[msg = CFP]\bar{u}\langle PriceInfo \rangle.u(msg1)(\\ & [msg1 = Refuse]SellerAgent + [msg1 = Buy]\bar{v}\langle ReqPay \rangle.v(msg2). \\ & [msg2 = PaySuc]\bar{u}\langle Deliver \rangle.SellerAgent) \end{aligned}$$

$$\begin{aligned} & BankAgent(v, w, ReqPay, PaySuc, InfPay, Payment) = v(msg). \\ & [msg = ReqPay]\bar{w}\langle InfPay \rangle.w(msg1).[msg1 = Payment]\bar{v}\langle PaySuc \rangle. \\ & BankAgent) \end{aligned}$$

Finally, the whole MAS in electronic commerce is described as follows.

$$MAS = BuyerAgent(\dots)|SellerAgent(\dots)|BankAgent(\dots)$$

During the execution of the electronic commerce, its architecture may be dynamically changed, e.g. other buyer agents may want to join the system; there are some faults in the bank agent, so the backup bank agent is startup; an additional seller agent is needed to improve the computation speed. The dynamic architecture can be represented by π -calculus. When the system begins to run, a backup bank agent is added and backups data. The creation process of a backup bank agent is shown as follows.

$$BupBankAgent = \tau.Create$$

If the primary bank agent goes down, the backup bank agent is startup. By referring to [15], the dynamic configuration process is shown as follows.

$$\begin{aligned} & (\nu x)(\bar{v}\langle x \rangle.\bar{w}\langle x \rangle.BankAgent|BupBankAgent)|v(z1).SellerAgent| \\ & w(z2).BuyerAgent \\ & \xrightarrow{\tau} (\nu x)(\bar{w}\langle x \rangle.BankAgent|BupBankAgent|SellerAgent\{x/z1\})| \\ & w(z2).BuyerAgent \\ & \xrightarrow{\tau} (\nu x)(BankAgent|BupBankAgent|SellerAgent\{x/z1\}|BuyerAgent\{x/z2\}) \end{aligned}$$

where x is the private channel of between the bank agent and backup bank agent. The bank agent pass x along its link v , w to the seller agent and the buyer agent.

2.2 Analysis of Multi-agent Systems Using π -Calculus

A significant advantage provided by π -calculus is that the verification and validation of the model can be accomplished before implementation, and help ensure a correct design with respect to the original specification to enable software engineers to develop reliable and trustworthy MAS.

Based on the formal specification, the behavior properties of MAS can be analyzed. Safety and deadlock can be automatically verified using the supporting tool, namely Mobility Workbench [16]. The MWB is a tool for manipulating and analyzing mobile concurrent systems described in the π -calculus. In the current version, MWB can analyze the basic functionalities of model, such as deadlock, model checking. Deadlock analysis can help eliminate human errors in the design process, and verify some key behaviors for the MAS model to perform as expected, and increase confidence in the MAS design process. Furthermore, π -calculus can analyze the bisimilarity which determines the equivalence of behavior. Behavioral equivalence is a very important concept.

In this paper, we verify the deadlock properties of MAS by means of the MWB. It is important that how to handle deadlock situations for development of electronic commerce systems, where the communication plays a key role. Using the command *deadlocks* in MWB we can check if an agent can deadlock or not. By analyzing, MWB finds out the whole MAS in electronic commerce is *No deadlocks found*. Consequently, the MAS model help ensure a correct design to enable software engineers to develop reliable and trustworthy MAS.

3 Conclusions

Multi-agent systems are regarded as the most promising technology to develop complex software systems. In this paper, we use π -calculus to model and analyze MAS dedicated to electronic commerce. π -calculus can depict the overall and individual characteristics of MAS, represent the interactions among agents, and analyze the properties of MAS, such as deadlock and bisimilarity. Most importantly, π -calculus is suitable for modelling the dynamic or evolving MAS topology which is important for MAS.

Acknowledgements

This work is supported by the Opening Foundation of the Key Laboratory of Opto-Electronic Technology and Intelligent Control (Lanzhou Jiaotong University), Ministry of Education, China. Grant NO. K04110.

References

1. Wooldridge, M.J., Jennings, N.R.: Agent theories, Architectures, and Languages: a Survey. Lecture Notes in Artificial Intelligence, Vol. 890. Springer-Verlag, Berlin Heidelberg New York (1995) 1-32
2. Wooldridge, M.J., Jennings, N.R.: Intelligent Agents: Theory and Practice. Knowledge Engineering Review 10 (1995) 115-152

3. Jiao, W., Zhou, M., Wang, Q.: Formal Framework for Adaptive Multi-agent Systems. In Proceedings of the IEEE/WIC International Conference on Intelligent Agent Technology. Beijing (2003) 442-446
4. Brazier, F.M.T., Dunin-Keplicz, B.M., Jennings, N.R., Treur, J.: DESIRE: Modelling Multi-agent Systems in a Compositional Formal Framework. International Journal of Cooperative Information Systems 1 (1997) 67-94
5. Xu, H., Shatz, S.M.: A Framework for Model-based Design of Agent-oriented Software. IEEE Transactions on Software Engineering 29 (2003) 15-30
6. Luck, M., d'Inverno, M.: A Formal Framework for Agency and Autonomy. In Proceedings of the First International Conference on Multi-Agent Systems. San Francisco (1995) 254-260
7. Wooldridge, M., Jennings, N. R., Kinny, D.: The Gaia Methodology for Agent-oriented Analysis and Design. International Journal of Autonomous Agents and Multi-Agent Systems 3 (2000) 285-312
8. DeLoach, S.A.: Multiagent Systems Engineering. International Journal of Software Engineering and Knowledge Engineering 11 (2001) 231-258
9. Odell, J., Parunak, H.V.D., Bauer, B.: Representing Agent Interaction Protocols in UML. In Proceedings of the first International Workshop on Agent Oriented Software Engineering. Limerick (2001) 121-140
10. Bresciani, P., Perini, A.: Tropos: an Agent-oriented Software Development Methodology. Autonomous Agents and Multi-Agent Systems 8(2004) 203-236
11. Medvidovic, N., Taylor, R.N.: A Classification and Comparison Framework for Software Architecture Description Languages. IEEE Transactions on Software Engineering 26 (2000) 70-93
12. Sangiorgi, D., Walker, D.: The Pi-Calculus: a Theory of Mobile Processes. Cambridge University Press (2001)
13. Canal, C., Pimentel, E., Troya, J. M.: Compatibility and Inheritance in Software Architectures. Science of Computer Programming 41 (2001) 105-138
14. Milner, R.: The Polyadic π -calculus: a Tutorial. Springer-verlag (1993)
15. Liao, J., Tan, H., Liu, J.: Describing and Verifying Web Service Using Pi-Calculus. Chinese Journal of Computers 28 (2005) 635-643
16. The Mobility Workbench. <http://www.it.uu.se/research/group/mobility/mwb> (2005)

A Cooperation Mechanism in Agent-Based Autonomic Storage Systems

Jingli Zhou, Gang Liu, Shengsheng Yu, and Yong Su

College of Computer Science & Technology,
Huazhong University of Science & Technology,
Wuhan 430074, China
youyouwosi@163.com

Abstract. Cooperation between storage devices is an important aspect of autonomic storage system. By employing multiple distributed storage resources, storage system can greatly improve its performance. In this paper, agent-based methodology is introduced to build an autonomic storage system infrastructure. To select appropriate storage devices, queuing models are established to estimate the future storage device performance. A replica selection method and a data allocation algorithm are presented to gain an aggregate transfer rate according to the predicted performance. The results show that our models are useful for evaluating the mean response time of storage devices.

1 Introduction

In the last few years there exists a trend that storage systems are intelligent enough to automate tasks associated with storage management. Many approaches have been proposed to achieve the self-managing target of storage systems, and one of the most promising is autonomic storage system that introduces the concept of autonomic computing [1] into storage systems. A prominent advantage of autonomic storage systems is cooperation between different storage devices. That means autonomic storage system can automatically make use of multiple storage devices to satisfy the performance and capacity requirements of workloads.

In autonomic storage systems, the primary problems related with cooperation are as follows:

1. Performance prediction for a single storage device.
2. Communication mechanism between storage components.
3. Resource selection.

Agent-based methodology has been investigated to design and implement large-scale distributed intelligent system. It is incorporated into storage systems to solve the cooperation problems between storage components in this paper. Furthermore, we introduce performance models to predict the mean response time of storage devices. In the models we sufficiently consider the effect of cache; read and write requests are analyzed respectively. To choose proper storage resource for an I/O request, we present a replica selection method for read request and a data allocation algorithm for write request. We have built a real environment to validate our performance models, and the results indicate that our models are useful.

The remainder of the paper is organized as follows. In section 2 we introduce some related work in cooperation of storage systems. In section 3 we introduce the architecture of an agent-based autonomic storage system. In section 4 we present performance model to analyze the storage device with a cache. In section 5 we present a replica selection method and a data allocation algorithm to meet both performance and capacity requirements of a workload. In section 6 we validate our performance model by experiment results. Finally, we conclude the paper in section 7.

2 Related Work

Currently with development of distributed and heterogeneous storage systems, improving performance by self-adaptive cooperation becomes increasingly attractive. zFS [2] is a decentralized file system that distributes all aspects of file and storage management over a set of cooperating machines. RepStore [3] unites the self-organizing capability of P2P DHT and achieves the best cost-performance balance automatically. Multi-Storage I/O System [4] can not only effectively manage various distributed storage resources, but also enhance performance by utilizing many I/O optimizations. Self-* Storage System [5] composed of networked “intelligent” storage bricks designs several intelligent self-* components to organize an integrated storage system. FAB [6] constructs a highly scalable distributed array of intelligent bricks with self-management features. MAPFS [8] is a Multi-Agent Parallel File System for clusters and offers a file system interface.

All of these efforts have not considered the effect of cache in a current storage device. In this paper we introduce queuing models to analyze storage devices with cache, based on the models, we can develop valid schemes to transfer data for read and write requests.

3 Architecture of Agent-Based Autonomic Storage System

In a cooperation mechanism, it is necessary to consider how to partition a task and how to assign subtasks. To partition an I/O request task into some subtasks, the accessed file can be divided into a number of objects [7]. By distributing the objects across many storage devices, autonomic storage systems have the potential to provide high capacity, throughput, availability and scalability.

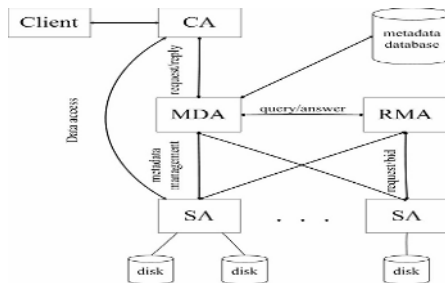


Fig. 1. architecture of agent-based autonomic storage system

Considering the agents with autonomy and social ability, it is very convenient for them to cooperate, manage and schedule distributed resources. The architecture of an agent-based autonomic storage system is illustrated in Figure 1.

Storage Agent (SA): Storage resources such as disk and tape are managed by SAs; SAs provide Object-Based Storage API.

Metadata Agent (MDA): MDA provide metadata API and preserve the metadata related with file management.

Client Agent (CA): CA resides at each client and presents the standard file system API. A CA can connect with multiple SAs. By making use of a big buffer, CA will accomplish the task of parallel I/O.

Resource Manager Agent (RMA): RMA is the manager of an autonomic storage system. RMA records all the storage resource within the storage system. Moreover, it also decides how data are partitioned among SAs, and how request are distributed.

4 Performance Prediction

A key issue of a self-managing storage system is storage device performance prediction. Many analytic models [9,10] have been utilized to analyze storage devices. This paper introduces queuing models that have advantage of analyzing SAs with cache. The follow describes the mean response time of read and write requests.

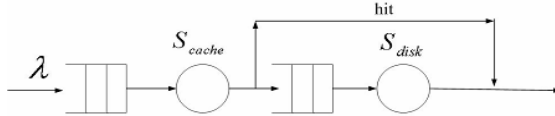


Fig. 2. Queuing model for read request

Figure 2 shows the queuing model for read request, where both the cache and the disk queuing model are M/M/1. The mean response time of a SA is:

$$T_{response} = \frac{1}{\mu_{cache} - \lambda_{cache}} + \frac{1}{\mu_{disk} - (1 - P_{cache_hit})\mu_{cache}} \tag{1}$$

Where λ_{cache} represents the arrival rate of cache; μ_{cache} represents the service rate of cache. μ_{disk} represents service rate of disk; P_{cache_hit} represents cache hit rate.

Due to write-behind policy, write requests are first written into cache, and cache signs the blocks as “dirty”. Then SA claims that the write request is complete. These “dirty” blocks will be written to disks finally. The queue of cache has a minimum and a maximum, called *min* and *max*. *min* indicates the number of “dirty” blocks in cache when cache begins to write data to disks, while *max* indicates the number of whole blocks in cache. The queuing model of cache is a Markov birth-death M/M/1/K(K=max-min) process; the queuing model of disk is still M/M/1.

Let λ_{cache} represent the arrival rate of cache, μ_{cache} represent the service rate of cache and ρ represent the cache utilization, $\rho = \lambda_{cache} / \mu_{cache}$. The probability distribution of dirty blocks in cache is

$$P_i = \begin{cases} \frac{\rho^{i-\min}}{\sum_{j=0}^K \rho^j}, & \min \leq i \leq \max \\ 0, & otherwise \end{cases} \quad (2)$$

The mean throughput of SA is

$$Throughput = \frac{\lambda_{cache} \times (1 - P_{\max})}{Num_{dirty_blocks}} \quad (3)$$

Where Num_{dirty_blocks} represents the number of dirty blocks to be written. When m tasks arrive at a SA, the mean response time is

$$T_{response} = \frac{m}{Throughput} \quad (4)$$

5 Replica Selection and Data Allocation

For read requests, as a file often has several replicas to improve availability and reduce response time of distributed system, the problem is to dynamically select the replicas that can satisfy a client’s timing requirement. to achieve a maximum throughput, all SAs must finish transmitting their data at the same time. Thus the portion of a file transferred by a SA should be proportional to its performance.

For write requests, if a write request is just to modify the existing objects, we need not to consider the issue of data allocation. If data are the first time to be written, data allocation is needed. Here we propose a performance-based data allocation (PDA) algorithm; the performance of SA is predicted based on our analytic model.

Suppose a CA partitions a file into N disjoint objects and store them into M SAs. Let n_i represent the number of objects that have been allocated to SA _{i} , the r_i represent the transmission rate for SA _{i} calculated according n_i , and c_i represent the remain capacity of SA. The minimum number of objects allocated to a SA is *min* in one step.

Algorithm PDA

Input: the number of objects N and the arrival rate λ of write requests

Output: the vector $\langle n_1, n_2, \dots, n_m \rangle$

1. Push all SAs to a list X
2. Calculate the arrival rate λ_i in SA _{i} according to the proportion of data having been allocated to SA _{i} . If no data has been allocated, an average value can be used.

3. Calculate r_i for SA_i in X according to λ_i and n_i . If no data has been allocated, suppose n_i is min
4. Choose the SA_j with the minimum transmission rate r_j . The data will be allocated to SA_j is $n_j + min$, and the data allocated to SA_i is $n_j + min r_i / r_j$
5. if $n_i > c_i$, then $n_i = c_i$, remove SA_i from X
6. If the objected has not been allocated completely, go to step 2; else stop

6 Experiment Results

We first evaluate the effect of performance models by comparing the estimated transfer rate against measured values of workloads running on a parallel environment. We utilize two PCs to act as SAs, and a PC to act as CA. To simplify calculation, every PC is assigned equal data.

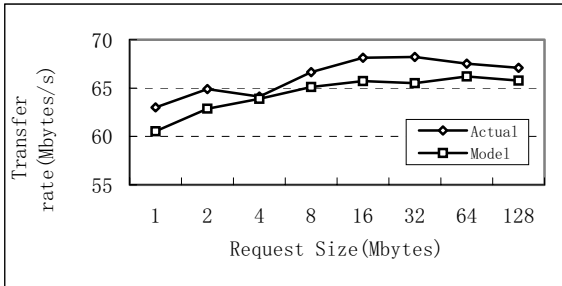


Fig. 3. Model vs. actual transfer rate for read request

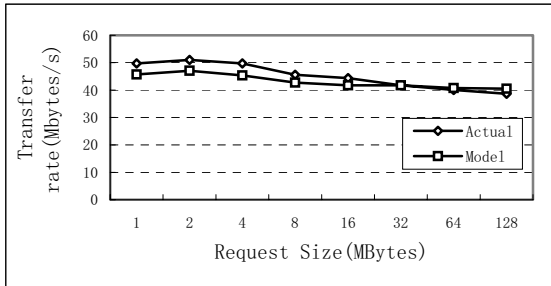


Fig. 4. Model vs. actual transfer rate for write request

Figure 3 and Figure 4 plot the model’s transfer rates and the measured transfer rates for read and write workloads for various request sizes. When we select appropriate cache hit rate and request arrival rate, the model’s performance predictions can match well with the experimental performance measures. Hence, we can conclude that our performance model can reflect the effect of cache in storage devices; it is also suitable for estimating whether a storage device can meet the performance requirement of a given workload.

7 Conclusion

In this paper, we have presented an agent-based system in heterogeneous autonomic storage systems. To achieve this target, we introduce performance models that can predict the mean response time of storage device with cache. We also propose an algorithm of data allocation. The evaluation results shown in the paper demonstrate that our performance model can reflect the behavior of SA.

Acknowledgement

This work was sponsored by National Science Foundation of China No.60373088 and National Key Laboratory Foundation No.51484040504JW0518.

References

1. Kephart, J.O., Chess, D.M.: The Vision of Autonomic Computing. IEEE Computer Society, Vol.36, no.1 (2003) 41-50
2. Ohad, R., Teperman, A.: zFS - A Scalable Distributed File System Using Object Disks. Proc. of 20th IEEE/11th NASA Goddard Conference on Mass Storage Systems and Technologies (2003)
3. Zhang, Z., et al.: RepStore: A Self-managing and Self-tuning Storage Backend with Smart Bricks. Proc. of the 1st International Conference on Autonomic Computing (2004)
4. Shen, X., Choudhary, A.: MS-I/O: A Distributed Multi-storage I/O System. Proc. of the 2nd IEEE/ACM International Symposium on Cluster Computing and the Grid (2002)
5. Mesnier, M., et al.: File Classification in Self-* Storage Systems. Proc. of the 1st International Conference on Autonomic Computing (2004)
6. Saito, Y., et al.: FAB: Building Distributed Enterprise Disk Arrays from Commodity Components. Proc. of the 11th international conference on Architectural support for programming languages and operating systems (2004)
7. Mesnier, M., Ganger, R., Riedel, E.: Object-Based Storage. IEEE Communications Magazine, Vol.41, no.8 (2003) 84-90
8. Perez, M.S., et al.: Cooperation Model of a Multiagent Parallel File System for Clusters. Proc. of IEEE International Symposium on Cluster Computing and the Grid (2004)
9. Anderson, B.: Mass Storage System Performance Prediction Using a Trace-Driven Simulator. Proc. of the 22nd IEEE/13th NASA Goddard Conference on Mass Storage Systems and Technologies (2005)
10. Varki, E., Merchant, A., Xu, J., Qiu, X.: Issues and Challenges in the Performance Analysis of Real Disk Arrays. IEEE Trans. on Parallel and Distributed Systems, Vol.15, no.6 (2004) 559-574

A Mobile Agent Based Spam Filter System

Xiaochun Cheng^{1,2}, Xiaoqi Ma¹, Long Wang², and Shaochun Zhong²

¹ Department of Computer Science, The University of Reading, Reading RG6 6AY, UK
{x.cheng, xiaoqi.ma}@reading.ac.uk

² Institute of Ideal Information Technology, North East Normal University,
Changchun 130024, China

Abstract. A new distributed spam filter system based on mobile agent is proposed in this paper. We introduce the application of mobile agent technology to the spam filter system. The system architecture, the work process, the pivotal technology of the distributed spam filter system based on mobile agent, and the Naive Bayesian filter method are described in detail. The experiment results indicate that the system can prevent spam emails effectively.

1 Introduction

Spam seems to be a growing problem. Since email becomes the most popular communication tool, recent increasing spam emails are wasting network resources and users' time [5][6]. Hence the research of spam filter technology is important.

With the increasing amount of emails, the requirement for the computing ability of the email server goes higher and higher, and a single server normally cannot meet all the needs. The clients and servers of most spam filter systems work independently instead of cooperatively. By applying the mobile agent technology [1][2][3][4][7] on spam filter system, different functions of email server are implemented on different servers. The clients and the servers of the systems can work collaboratively, and the information in the system can be shared.

Spam filter technology mainly includes filters based on blacklist, email header checking and email contents checking [5][6]. The servers and clients of spam filter systems accomplish multi-filtering by using the methods mentioned above. At present, the filter based on email contents checking is the mainstream technology used to filter spam. This technology includes rule-based method and probability-based method. In the probability-based method, the Bayesian filter method [5] is one key technology. This paper implements the filter based on email contents checking by using the Naive Bayesian filter method.

2 Mobile Agent Technology and Naive Bayesian Filter Method

2.1 Mobile Agent Technology

A mobile agent is an autonomous program, capable of traveling to a network host connected. If permitted, it can execute and communicate locally with other entities.

The mobile agent systems include two parts: mobile agent and mobile agent environment (MAE). Mobile agent environment provides the executing environment for the mobile agents, while mobile agents implement the traveling in the mobile agent environment [1][2][4][7].

One of the characteristics of mobile agents is mobility, which distinguishes mobile agents and ordinary agents. Another characteristic of mobile agents is independence. Mobile agents can execute continuously under the condition that others do not intervene, and can control their actions and their inner states.

The system established on mobile agents has many advantages, such as distribution, independence, and heterogeneity. These advantages can make up the drawbacks of the existing spam filter systems. It is feasible to apply the mobile agent technology on the spam filter system to have good performance.

2.2 Naive Bayesian Classification

The working principle of the Naive Bayesian Filter [5] is described as following. Firstly, the filter learns from the spam set and the non-spam set to establish the feature vectors of spam and non-spam. When an email is received, the filter extracts the features of the email contents and establishes the vector space of the email contents. Then the filter computes the probability of belonging to spam (P_1) and the probability of belonging to non-spam (P_2). If $P_1 > P_2$, then the email is a spam, otherwise the email is non-spam.

Suppose the email content L has n features (w_1, w_2, \dots, w_n), and the sample space has two sorts: spam C_1 and non-spam C_2 . Assume N_1 is the amount of the emails belonging to C_1 , and N_2 is the amount of the emails belonging to C_2 . The probability P_1 and P_2 are:

$$P_m = P(C_m | L) = P(C_m) \times P(L | C_m) \quad m = 1, 2$$

where $P(C_1)$ and $P(C_2)$ are:

$$P(C_m) = \frac{1 + N_m}{2 + N_1 + N_2} \quad m = 1, 2$$

The probabilities $P(L|C_1)$ and $P(L|C_2)$ are:

$$P(L | C_m) = P((w_1, w_2, \dots, w_n) | C_m) = \prod_{i=1}^n P(w_i | C_m) \quad m = 1, 2$$

$P(w_i|C_1)$ is the probability of w_i belonging to C_1 , and $P(w_i|C_2)$ is the probability of w_i belonging to C_2 . They can be computed as follows:

$$P(w_i | C_m) = \frac{1 + TF(w_i, C_m)}{|V| + \sum_{j=1}^{|V|} TF(w_j, C_m)} \quad m = 1, 2$$

where $|V|$ is the amount of features, and $TF(w_i, C_m)$ is the sum of times of w_i appearing in C_m . The weight is applied to increase the precision of the classification.

3 System Model and Work Process

3.1 System Architecture

According to the function requirements of the distributed Mobile Agent Spam Filter System (MASFS) as well as the organization relations, the cooperation relations and the business relations of the system, the different functions of the system are abstracted and implemented on different agents. The system architecture is shown in Fig. 1. The header check agent, the feature extraction agent, the classification agent, and the learning and training agent of the system are all mobile agents.

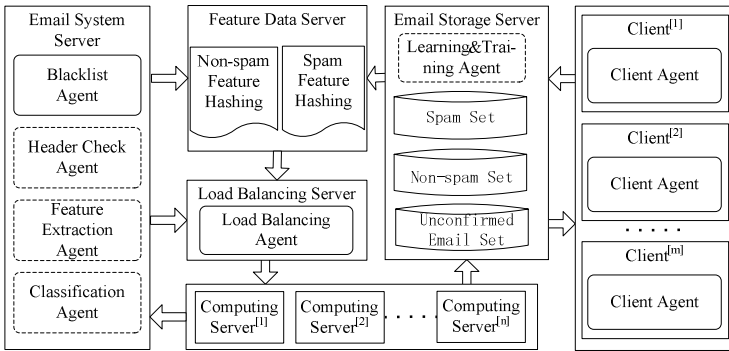


Fig. 1. The System Architecture of MASFS

3.2 System Components

The email system server is a multi-agent system. The blacklist agent extracts the address of the email sender. If the address is in the blacklist, the agent will refuse the connection immediately. The header check agent analyses the domain names of the email sender and receiver, then the agent will refuse to accept the email if the domain names are illegal. The feature extraction agent extracts the abstract of the contents of an email, then extracts the features of the email from the abstract. The classification agent classifies the email according to the features by using the Naive Bayesian filter method. The non-spam is marked *N*, while the spam is marked *S*. Then the marked emails are sent to the unconfirmed email set on the email storage server.

The email storage server contains a non-spam set, a spam set, an unconfirmed email set, and an email training agent. The email training agent is a mobile agent, which can obtain the information of the spam features and the non-spam features, and modify the spam feature hashing and the non-spam feature hashing in the feature data server using the spam set and the non-spam set. The frequencies of the features are saved in the feature hashing, where the frequency is calculated using the average times that the features of emails occur in the email sets.

A load balancing agent is included in the load balancing server, which is a non-mobile intelligent agent [7]. The load balancing agent distributes all kinds of

processes to computing servers by monitoring their computing processes. The system makes the best of the hardware equipments regarding requirements.

A client agent is included in the client, which is a non-mobile intelligent agent. The client agent receives new emails from the unconfirmed emails set on the email storage server, filters the emails, and then puts non-spam in the inbox, and puts spam in the recycler. After that it waits for the modification and checkup by the user. Then it sends the email messages to data server. The filter methods include tag filter and blacklist filter.

3.3 System Work Process

The work process of MASFS is shown in Fig 2. The working relations of the mobile agents are illustrated in the figure.

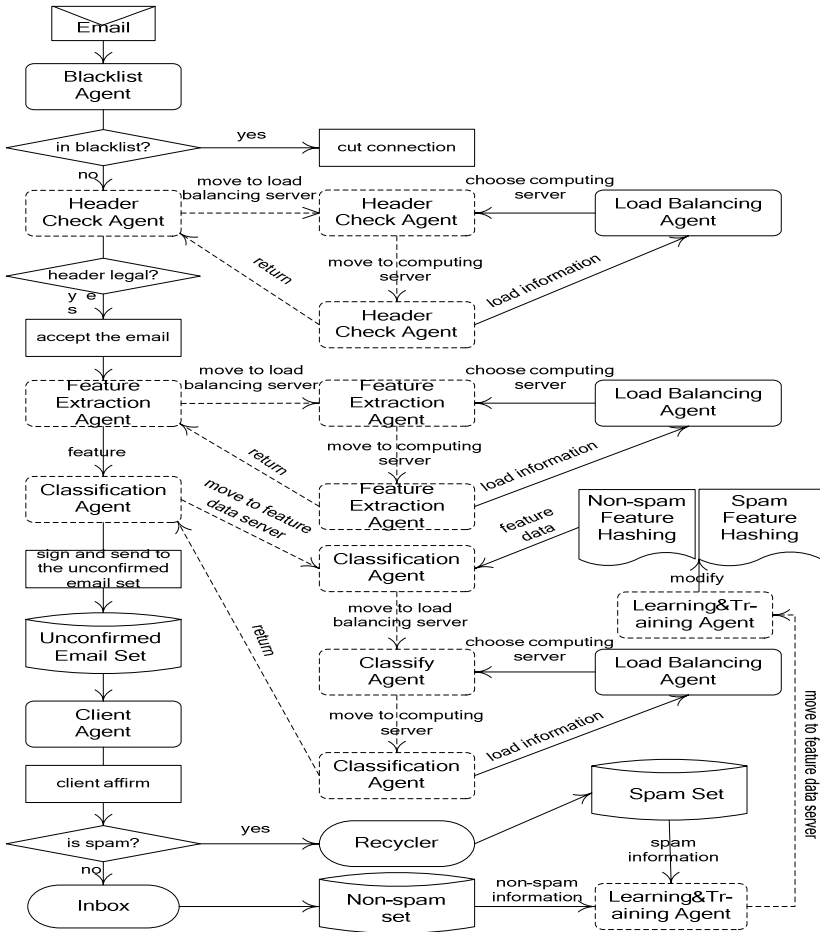


Fig. 2. The Work Process of MASFS

We can see that only small parts of the filter processes are on-line processes. So the system requires little network resources, and depends little on the stability of the network.

4 Experiment

Based on the above design, we set up the experiment. We collected 10000 emails, among which there are 5785 spam emails and 4215 non-spam emails. The experiments were conducted in cases when network traffic was normal and when network traffic was violently changed.

We extracted randomly 70% emails among spam set and non-spam set as learning set. The 50% of remaining emails were divided into five parts for testing. The remaining emails are used for testing too.

We filtered and tested each testing set by running the system. The testing sets that have more emails were tested later. The amount of the emails and the errors were recorded at the execution time. We computed the precision of the filtering at the execution time.

The result of the experiment is shown in Table 1.

According to the data listed, we can say that:

- (1) The system produced high precision in filtering emails;
- (2) The precision of filtering emails can be improved while the system is used;
- (3) The system can process with high speed;
- (4) The stability of the network has little influence on the system.

5 Conclusions

This paper presented the design and implementation of a distributed spam filter system based on mobile agent technology. The intelligent and mobile agents are distributed among the pivotal points of the system. Since the agents work cooperatively, the system has advantages, such as on intelligence, distribution, mobility and scalability. The system can be easily maintained and managed. The testing result of the experiment indicates that the system can effectively prevent the spam emails.

Table 1. Experiment Results

Test Set	Normal Network Bandwidth		Abnormal Network Bandwidth	
	Precision (%)	Time cost (sec.)	Precision (%)	Time cost (sec.)
1	91.23	175	92.13	193
2	92.35	167	92.23	201
3	93.45	181	92.31	187
4	93.50	179	93.36	191
5	93.54	171	93.40	184
6	93.55	623	93.42	701

Acknowledgement

Our research has been supported by EC, EPSRC, the National Natural Science Foundation of China, and Hong Kong K C Wang Education Foundation.

References

1. Zhong, S., Song, Q., Cheng, X., Zhang, Y.: A Safe Mobile Agent System for Distributed Intrusion Detection. Proceedings of the 2nd IEEE International Conference on Machine Learning and Cybernetics (ICMLC'03), Xi'an, China, 2009~2014, 2003.
2. Lyu, M.R., Chen, X.Y., Wong, T.Y.: Design and Evaluation of a Fault-Tolerant Mobile-Agent System. IEEE Intelligent Systems, 19(5): 32~38 (2004).
3. Cao, J.N., Fen, X.Y., Lu, J., Das, S.K.: Design of Adaptive and Reliable Mobile Agent Communication Protocols. Proceedings of the 22nd IEEE International Conference on Distributed Computing Systems (ICDCS'02), Viena, Austria, 471~427, 2002.
4. He, X., Cheng, X.: A Mobile Agent Model for Agent Based Intrusion Detection System, Proceedings of the IEEE SMC UK-RI Chapter Conference 2005 on Applied Cybernetics September 7-8, 2005, City University, London, United Kingdom, ISSN 1744-9189, 93-99.
5. Androusoopoulos, I., Koutsias, J., etc.: An Evaluation of Naive Bayesian Anti-Spam Filtering. Proceedings of the Workshop on Machine Learning in the New Information Age. Proceedings of the 11th European Conference on Machine Learning, Barcelona, Spain, 9~17, 2000.
6. Spam Statistics 2004, <http://spam-filter-review.toptenreviews.com/spam-statistics.html>.
7. Li, Y., Cheng, X., Formal Analysis for Dynamic Planning of Intelligent Agents, Proceedings of the IEEE SMC UK-RI 3rd Workshop on Intelligent Cybernetic Systems(ICS'04), pp 172-177. Londonderry, Northern Ireland, UK. Sept 2004.

Hexagon-Based Q-Learning to Find a Hidden Target Object*

Han-Ul Yoon and Kwee-Bo Sim

School of Electrical and Electronics Engineering, Chung-Ang University,
221 Heukseok-Dong, Dongjak-Gu, Seoul 156-756, Korea
huyoon@wm.cau.ac.kr, kbsim@cau.ac.kr

Abstract. This paper presents the hexagon-based Q-learning to find a hidden target object with multiple robots. We set up an experimental environment with three small mobile robots, obstacles, and a target object. The robots were out to search for a target object while navigating in a hallway where some obstacles were placed. In this experiment, we used two control algorithms: an area-based action making (ABAM) process to determine the next action of the robots and hexagon-based Q-learning to enhance the area-based action making process.

1 Introduction

Nowadays, robots are coping with tasks previously performed by men in dangerous field, such as rescue missions at fire-destroyed building or at gas contaminated sites; information retrieval from deep seas or from space; and weather analysis at extremely cold areas like Antarctica. Sometimes, multiple robots are especially needed to penetrate into hard-to-access areas, such as underground insect nests, to collect more reliable and solid data.

Multiple robot control has received much attention since it can offer a new way of controlling multiple agents more flexibly and robustly. For instance, Parker used the heuristics approach algorithm for multiple robots and applied it to cleaning tasks [1]. Ogasawara employed distributed autonomous robotic systems to control multiple robots that transporting a large object [2]. However, the more communication dependency does a system have, the more difficult a system hierarchy becomes; therefore, we propose an area-based action making (ABAM) process for an *implicit cooperation* such as ants world. This, in turn, becomes incorporated with hexagon-based Q-learning, which helps multiple robots to navigate, to avoid collision, and to search through their own trajectories.

Reinforcement learning through the explorations of its environment enables an agent to actively determine what the following action should be. During the exploration of an uncertain state space with reward, it can learn what to do by continuum of its state history and appropriately propagating rewards through the state space [3]. In our research, we focused on Q-learning as a reinforcement

* This research was supported by the Brain Neuroinformatics Research, Jul. 2004 to Mar. 2008, Program by Ministry of Commerce, Industry, and Energy in Korea.

learning technique. It is because Q-learning is a simple way to solve Markovian action problems with incomplete information. Also, it can map state-action pairs onto expected returns on the basis of the action-value function Q [4]. In addition to this simplicity, Q-learning can adopt to the real world situation. For example, the state space can be matched with the physical space of the real world. An action can also be regarded as physical robot maneuver. In this paper, we propose that the hexagon-based Q-learning can enhance the area-based action making process so that the learning process can adapt to real world situations better.

The organization of this paper is as follows. In section 2, the area-based action making process is introduced. In section 3, hexagon-based Q-learning adaptation is presented. In section 4, experimental results from the application of two different searching methods to find the object are presented. In section 5, conclusions are presented.

2 Area-Based Action Making Process

Area-based action making (ABAM) process is a process that determines the next action of a robot. The reason why this process is referred to as ABAM is because a robot recognizes surrounding not by distances, i.e., from itself to obstacle, but by areas around itself. The key idea of the ABAM process is to reduce the uncertainty of its surrounding. It is similar to the behavior-based direction change in regards to controlling the robots [5][6]. Under ABAM process robots recognize the shape of their surrounding and then take an action, i.e., turn and move toward where the widest space will be guaranteed. Consequently, each robot can avoid colliding into obstacles and other robots. Figure 1 depicts the different actions taken by distance-based action making (DBAM) and by ABAM in the same situation, respectively [7]. As it can be inferred by their name, the direction $d4$, the longest distance, will be selected under DBAM; otherwise, $a4$, the widest area, will be culled under ABAM.

Figure 2 illustrates the mechanism how the robot can avoid both obstacle and collision, and also proportionate its tracking area. In Fig. 2a, the robot is surrounded by 6-obstacles. Under ABAM, the robot will calculate the areas of its surrounding, and then it will recognize that an action toward the north will guarantee the widest space. Therefore, the robot will change its direction accordingly. In Fig. 2b, two robots are encountered by each other at one place. In

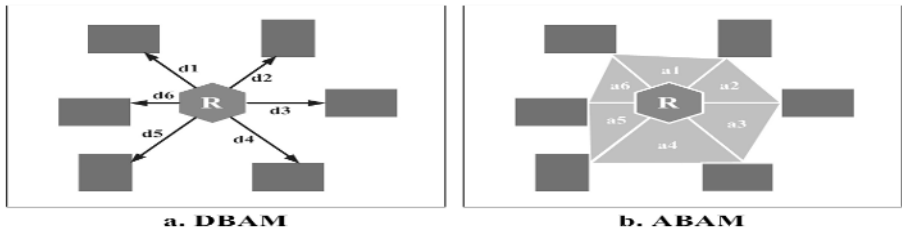


Fig. 1. Different actions taken under DBAM and ABAM

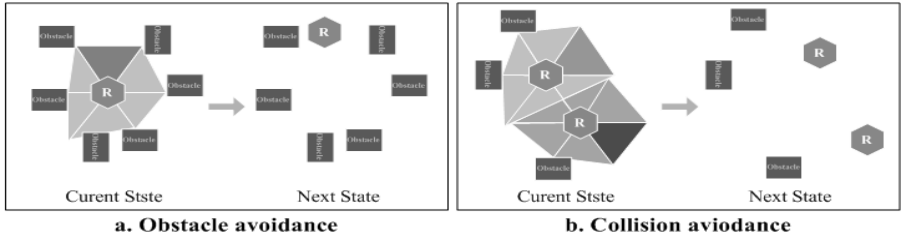


Fig. 2. Illustrative examples of the obstacle and collision avoidance of robot

this scenario, the infrared emitters of two robots will be faced side by side; thus, this encounter will inform each robot that other side has been already covered by its peer. Consequently, two robots will take its maneuver to somewhere not to violate peer’s search area. It is the reason why we refer ABAM as an *implicit cooperation* process.

3 Hexagon-Based Q-Learning

Q-learning is a well-known algorithm for reinforcement learning. It leads an agent to acquire optimal control strategies from delayed rewards, even when it has no prior knowledge of the effects of its actions on the environment [8][9].

The Q-learning for our robot system is adapted to enhance the ABAM process. The adaptation can be performed with a simple and easy modification, namely, hexagon-based Q-learning. Figure 3 illustrates example of hexagon-based Q-learning. Compared with normal square-based state space, the only thing that was changed is the shape of state space. The reason why we changed the shape of the space from a square to a hexagon was that the hexagon is a polygon which can be expanded infinitely by its combination. According to this adaptation, the robot can take an action in 6-direction and have 6-table entry \widehat{Q} value. In Fig. 3, the robot is in the initial state. Now, if the robot decides that $+60$ degree guarantee the widest space after calculating of its 6-areas of surrounding, the action of the robot would be a_{+60° . After the action is taken, if $Area6'$ is the widest area, the value of $\widehat{Q}(s_1, a_{+60^\circ})$ will be updated by the formula (1) as

$$\begin{aligned}
 \widehat{Q}(s_1, a_{+60^\circ}) &\leftarrow r + \gamma \max_{a'_\theta} \widehat{Q}(s_2, a'_\theta) \\
 &\leftarrow 0 + \gamma \max_{a'_\theta} \{Area1', Area2', \dots, Area6'\} \\
 &\leftarrow \gamma Area6' .
 \end{aligned}
 \tag{1}$$

where s is a possible state, a is a possible action, r indicates an immediate reward value, here is predetermined 0, and γ is the discount factor. After the movement from the initial state to the next state, immediate reward becomes

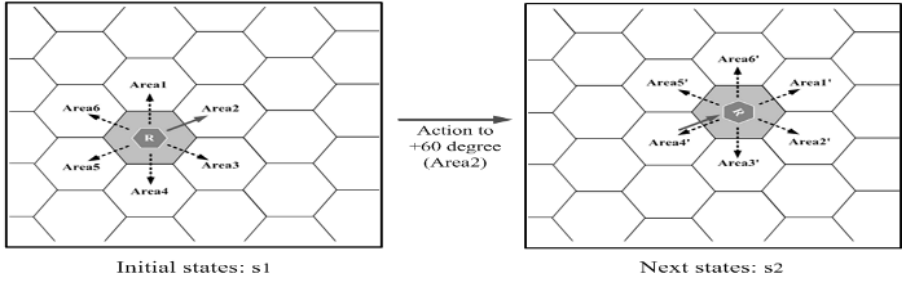


Fig. 3. An illustrative example of Hexagon-Based Q-learning

the difference between *the sum of total area after action is taken* and *the sum of total area before action is taken*. Thus, the reward is

$$r = \sum_{j=1}^6 Area_j - \sum_{i=1}^6 Area_i . \tag{2}$$

where $Area_i \in s$ and $Area_j \in s'$ respectively. The hexagon-based Q-learning algorithm is presented in Table 1.

Table 1. Hexagon-based Q-learning algorithm

For each s, a initialize the table entry $\widehat{Q}(s, a_\theta)$ zero
Calculate 6-areas at the current state s
Do until the task is completed
• Select the action a_θ to the widest area, and execute it
• Receive the immediate reward r
• Observe the new state s'
If $\widehat{Q}(s', a'_\theta)$ is greater or equal than $\widehat{Q}(s, a_\theta)$
• Update the table entry for $\widehat{Q}(s, a_\theta)$
• $s \leftarrow s'$
Else, if $\widehat{Q}(s', a'_\theta)$ is far less than $\widehat{Q}(s, a_\theta)$
• Move back to the previous state
• $s \leftarrow s$

4 Experiments with Two Different Control Methods

We performed experiments by using two control methods: ABAM and enhanced ABAM by hexagon-based Q-learning. The target was a stationary robot with the same shape as other robots and the color was green. It was located at a hidden place behind one obstacle such like a desk. Three robots, whose mission is to find the target, would recognize it by its color and shape. They would decide whether they finished the task by detecting the target after each action was taken.



Fig. 4. Three-robots are searching the object using ABAM

ABAM. We applied ABAM to the robots. They could sense their environment with 6-infrared sensors and calculate 6-areas. When the calculation was done, each robot tried to move toward the direction where the widest area would be guaranteed. Once the robots started to move, each robot spread out into the environment. Figure 4 shows that one robot located at the right side of the target succeeded to complete the task.

Enhanced ABAM by Hexagon-Based Q-Learning. We used the hexagon-based Q-learning to ABAM as a modified control method. This method allowed the robots to reduce the probability of wrong judgment and to compensate wrong judgment with reinforcement learning. It also learned the experimental environment, state by state, and canceled the state transition if the action caused critical reduction of \hat{Q} value. By using the hexagon-based Q-learning adaptation to ABAM, we could obtain more refined outcomes as more than two robots completed the task during ten trials and task completion time was faster. The search with hexagon-based Q-learning is presented in Fig. 5.



Fig. 5. Three-robots are searching the object using hexagon-based Q-learning

5 Conclusions

In this paper, we presented the area-based action making (ABAM) process and hexagon-based Q-learning. Three small mobile robots were used to search for the object hidden in the unknown space. The experimental results from the application of the two different control methods in the same situations were presented.

The area-based action making process and hexagon-based Q-learning can be a new way for robots to search for an object in the unknown space. This algorithm also enables the agents to avoid obstacles during their search.

For the future research, first, we need to clarify the problem of accessing the object. In other words, if multiple robots are to carry out a task such as object transporting or block stacking, they need to recognize the object first and then proceed to approach it. Second, our robot systems desire to be improved so that the main part and the subparts could adhere more strongly. In addition, stronger complex algorithms, such as Bayesian learning or TD(λ) method, need to be adapted. Third, a self-organizing Bluetooth communication network should be built so that robots can communicate with one another robustly even if one or more robots are lost. Finally, the total system needs to be refined to obtain better results.

References

1. Parker, L.: Adaptive Action Selection for Cooperative Agent Teams. Proc. of 2nd Int. Conf. on Simulation of Adaptive Behavior (1992) 442–450
2. Ogasawara, G., Omata, T., Sato, T.: Multiple Movers Using Distributed, Decision-Theoretic Control. Proc. of Japan-USA Symp. on Flexible Automation **1** (1992) 623–630
3. Ballard, D.: An Introduction to Natural Computation. The MIT Press Cambridge (1997)
4. Jang, J., Sun, C., Mizutani, E.: Neuro-Fuzzy and Soft Computing. Prentice-Hall New Jersey (1997)
5. Ashley, W., Balch, T.: Value-Based Observation with Robot Teams (VBORT) using Probabilistic Techniques. Proc. of Int. Conf. on Advanced Robotics (2003)
6. Ashley, W., Balch, T.: Value-Based Observation with Robot Teams (VBORT) for Dynamic Targets. Proc. of Int. Conf. on Intelligent Robots and Systems (2003)
7. Park, J.B., Lee, B.H., Kim, M.S.: Remote Control of a Mobile Robot Using Distance-Based Reflective Force. Proc. of IEEE Int. Conf. on Robotics and Automation **3** (2003) 3415–3420
8. Mitchell, T.: Machine Learning. McGraw-Hill Singapore (1997)
9. Clausen, C., Wechsler, H.: Quad-Q-Learning. IEEE Trans. on Neural Network **11** (2000) 279–294

A Naive Statistics Method for Electronic Program Guide Recommendation System

Jin An Xu and Kenji Araki

Graduate School of Information Science and Technology,
Hokkaido University, Kita 14 Nishi 9, Kita-ku,
Sapporo 060-0814, Japan
{xja, araki}@media.eng.hokudai.ac.jp

Abstract. In this paper, we propose a naive statistics method for constructing a personalized recommendation system for the Electronic Program Guide (EPG). The idea is based on a primitive approach of N-gram to acquire nouns and compound nouns as prediction features, and then to combine the *tf · idf* weighting to predict user favorite programs. Our approach unified feedback process, system can incrementally update the vector of extracted features and their scores. It was proved that our system has good accuracy and dynamically adaptive capability.

1 Introduction

Today, a number of technical developments, such as satellite, cable and digital TV technology have resulted in an increasing number of available TV channels, hundreds of channels broadcast thousands of TV programs everyday. The challenge, how to offer a convenient and intelligent user interface, has become a research point.

There have been several research projects around EPG recommendation system [1,2,3,4]. In our previous work, an approach using Inductive Learning with N-gram to predict user's habits and preferences, showed good dynamically adaptive capability in small data sets [5].

In this paper, we propose a naive statistics method for constructing a personalized recommendation system for EPG. The idea is based on a primitive approach of N-gram to acquire nouns and compound nouns as prediction features, and then to combine the *tf · idf* weighting to predict user favorite programs, and then to unified feedback process. The objective is to develop a good intelligent user interface between each TV fan and his/her digital television.

This paper includes three sections as follows: Presentation of our system architecture, evaluation of the performance of the present system and a summary of this work.

2 Outline

This section describes the procedure of our proposed method as shown in Figure 1, our system consists of term extraction process, prediction process, user feedback process and IEPG extraction process.

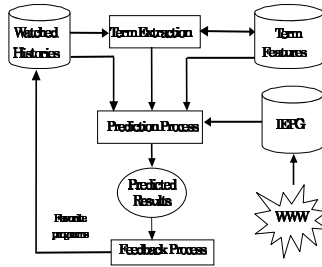


Fig. 1. Outline of the Procedure

2.1 Term Extraction

Our idea of term extraction is to use bigrams. It is based on the algorithm of Nakagawa [6] et al., which considered both the score using simple noun bigrams as components of compound nouns to calculate score of compound nouns, and the nouns independent frequency to calculate the score of each extracted term. These scoring methods content of four steps:

First, acquiring of simple noun bigrams including their frequencies shown in Figure 2 where [N M] means bigram of noun N and M.

In Figure 2, [LN_i N] (i=1,...,n) and [N RN_j] (j=1,...,m) are single-noun bigrams which constitute (parts of) compound nouns. #L_i and #R_j (i=1,...,n and j=1,...,m) mean the frequency of the bigram [LN_i N] and [N RN_j] respectively. Note that since we depict only bigrams, compound nouns like [LN_i N RN_j] which contains [LN_i N] and/or [N RN_j] as their parts might actually occur in a corpus. Again this noun trigram might be a part of longer compound nouns.

Second, inassigning the direct score of a noun, since a scoring function based on [LN_i N] or [N RN_j] could have an infinite number of variations, the following sample, yet is presentative, scoring functions are considered.

#LDN(N) and #RDN(N): The number of distinct single-nouns which directly precede or succeed N. There are exactly "n" and "m" in Figure 2.

LN(N,k) and RN(N,k): The general functions that take into account the number of occurrences of each noun bigram like [LN_i N] and [N RN_j] are defined as follows.

$$LN(N,k) = \sum_{i=1}^{LDN(N)} (\#L_i)^k \tag{1}$$

$$RN(N,k) = \sum_{j=1}^{RDN(N)} (\#R_j)^k \tag{2}$$

We can find various functions by varying the parameter **k** of (1) and (2). For instance, #LDN(N) and #RDN(N) can be defined as LN(N,0) and RN(N,0).

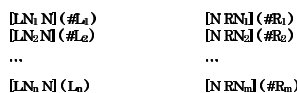


Fig. 2. Noun bigram and their Frequency

$LN(N,1)$ and $RN(N,1)$ are the frequencies of nouns that directly precede or succeed N .

Third, scoring the importance of compound nouns, it means to extend the scoring functions of a single-noun to the scoring functions of a compound noun. We adopt a very simple method, namely a geometric mean. Now think about a compound noun: $CN = N_1N_2 \dots N_L$. Then a geometric mean: GM of CN is defined as follows.

$$GM(CN,k) = \left(\prod_{i=1}^L (LN(N_i,k)+1)(RN(N_i,k)+1) \right)^{1/2L} \tag{3}$$

And then, combining the score of compound nouns and frequency of nouns, if a single-noun or a compound noun occurs independently, the score of the noun is multiplied by the number of its independent occurrences. Then GM(CN,k) of the formula (3) is revised. This new GM is called FGM(CN,k) and define it as follows.

if N occurs independently then

$$FGM(CN,k) = GM(CN,k) \times f(CN), \tag{4}$$

where $f(CN)$ means the number of independent occurrences of noun CN.

Finally, normalizing the credibility metric between 0 and 1 for each extracted term. These terms are extracted from programmes watched over time and from programmes available to the viewer, and a recommender system incrementally updates the scores of terms extracted from programmes viewed. In our system, we just focus on Japanese nouns and unknown words as extraction targets.

2.2 $tf \cdot idf$

$tf \cdot idf$ ranking is used for searches. It is a way of weighting the relevance of a term to a document. For a vector search in a set of documents for example, a document vector over all known terms is calculated for every document with $tf \cdot idf$ ranking. The correlations between this vectors and a query vector are then used to weight the documents according to a query [7].

The ranking takes two ideas into account for the weighting. The Term Frequency (tf) in the given document and the Inverse Document Frequency (idf) of the term in the whole documents. The tf in the given document shows how important the term is in this document. The document frequency of the term shows how generally important the term is. A high weight in a $tf \cdot idf$ ranking scheme is therefore reached by a high term frequency in the given document and a low document frequency of the term in the whole documents.

In this paper, we take each a program as a document. The $tf \cdot idf$ is defined as follows: Let d be a program, let w_i be the i th word in d . The tf of w_i , $tf(w_i, d)$, is the number of times w_i occurs in d . The program frequency of w_i , $df(w_i)$, is the number of program in which w_i occurs at least once. The inverse program frequency of w_i , $idf(w_i)$, is defined as follows:

$$idf(w_i) = \log \frac{|N|}{dfw_i} \tag{5}$$

```
'ドラマ=> Drama', '映画=> Movies', 'スポーツ => Spor
'演劇 => Drama', '音楽=>Music', 'バラエティ=>Talk
'ニュース・報道=> News', '趣味・実用=>Home/How-to'
'ドキュメンタリー・教養=>Documentary', 'キッズ=>Kids'
'アニメ・特撮=> Animation', '教育=>Educational'
'情報=> Home/How-to', 'その他 => Etc.'
```

Fig. 3. Categories of Japanese TV Programs

where $|N|$ is the total number of programs. Then, the $tf \cdot idf$ of w_i is defined as $tf(w_i, d) \cdot idf(w_i)$. In this paper, we use the normalize $tf \cdot idf$, shown as follows:

$$d_{ik} = \frac{tf_{ik} \times \log \frac{|N|}{df_{w_i}}}{\sqrt{\sum_{j=1}^N (tf_{ik} \times \log \frac{|N|}{df_{w_j}})^2}} \tag{6}$$

2.3 IEPG Extraction

The Internet is a globally distributed dynamic information repository that contains vast amount of digitized information, and more and more such information now available in multimedia forms.

We extract TV programs to XML format as our IEPG database from internet using XMLTV module [8], from <http://www.ontvjapan.com/program/>. All of the extracted IEPG data is classified automatically according to the Concurrent Versions System (CVS) of Japanese TV programs. The categories are shown in Figure 3.

IEPG data vector consists of start time, stop time, channel, title, description, Japanese category, English category, etc.. In our system, we changed some contents of IEPG, such as time, day of week and channel names and so on.

2.4 Prediction

The basic idea of prediction is to use extracted term features and $tf \cdot idf$ to evaluate each new TV program. The terms were extracted from user watched programs. These histories have the same format as the IEPG. They are classified according to its category, as shown in Figure 3. Using these categories, we can acquire classified text for each category, and then extract term features for each category. We assume acquired terms to be a vector of each category as follows:

$$F_i = (C_k, T_i, S_i, F_i), \tag{7}$$

where F_i is the i th term features vector, C_k is the k th category, T_i is the i th extracted terms, S_i is the score of the i th term, F_i means the frequency of the i th term.

The credibility of each new TV program for prediction process was calculated as follows:

- (1) Acquire all categories feedback tasks from watched programs to an array as a Categories Array (CA), re-ordering them according to the frequency of our feedback strategy (see next section).

(2) For each CA, acquire the category terms vector from term features to an array as a Terms Array (TA).

(3) For each CA, extract all new programs to another array as a Programs Array (PA).

(4) For the i th element PA_i of PA, extract the element TA_i if it matches PA_i , push TA_i to an array Q. let Q be the query of using $tf \cdot idf$, to calculate the total $tf \cdot idf$ score ($S_{PA_i}^{tfidf}$) of PA_i for query Q according to the Eq. 5 and Eq. 6, the N of the Eq. 5 is the number of PA.

(5) For the PA_i , calculate the credibility using the score of the query Q according to Eq.8, when the term of Q_k matches current evaluated PA program, reorder the evaluated PA according to the credibility.

$$S_{PA_i}^{bigram} = \sum_{k=1}^{max} Q_k^{S_k} \tag{8}$$

(6) The Score of the PA_i is calculate as follows.

$$S_{PA_i}^{Score} = (S_{PA_i}^{tfidf} \cdot S_{PA_i}^{bigram}) / (S_{PA_i}^{tfidf} + S_{PA_i}^{bigram}) \tag{9}$$

(7) Acquire all of the score of new programs to an array as a Result Array(RA).

(8) Sort RA according to the score, day of week etc..

(9) Output results to user, we can give some thresholds to control the number of recommendations.

2.5 Feedback

For realizing the dynamically adaptive capability, we have taken into account two factors that cause the dynamics of personalization.

The first step is to split the watched programs by time to generate two terms vectors. For prediction of the user’s favorite programs, a simple vector space modification model is used as a feedback method as follows:

$$H' = H_{new} + \gamma \cdot H_{old}, \tag{10}$$

where, H' is the extracted terms vector for prediction, H_{old} means the extracted terms vector from user watched programs before two weeks ago. H_{new} means the extracted terms vector from the latest user watched programs of two weeks. γ is coefficient. There are three ways to determine its value: by using results of existing research, or by determining them via experiments and/or by automatically learning them within the system.

The second step is to create some tasks for each category according to user watched programs, elements of tasks are start time (including day of week), channel, title, Japanese category, English category and the frequency of watched programs of certain category as shown in Figure 4.

These tasks are extracted from the watched programs, the frequency was acquired from user watched programs. Using these frequencies, to re-order the acquired categories for improving prediction precision.

Example of Japanese Feedback Tasks

```

<programme start="火曜日 19:00" channel="HTB">
<title lang="ja_JP">プロ野球</title>
<category lang="ja_JP">スポーツ</category>
<category lang="en">Sports</category>
<frequency>5</frequency>
</programme>

```

English Explanation

火曜日: Tuesday, プロ野球: Professional Baseball, スポーツ: Sports.

Fig. 4. Example of Category Feedback Tasks

3 Experiment

As mentioned above, the system based on our proposed approach was developed for experimentation to investigate its validity. In our experiments, we use open data to test the performance of our system. We adopted periodic training to our system. The training data is incremented on weekly basis.

Our experiment datasets were collected based on daily life of five graduate students of engineering over a period of about three months and total number of data was 2974. The data size of every week is shown in Table 1. The values of γ was given 0.5, the optimize value γ will be investigated in future experiments.

In order to keep the starting state constant for each user, the watched programs and the term features always started from an empty initial state.

Table 1. The Data Size of Each Week

Weeks	1	2	3	4	5	6	7	8	9	10	11	12
Data Num. (A)	48	47	46	48	52	52	51	51	50	45	48	48
Data Num. (B)	52	50	48	52	50	51	48	52	50	48	49	48
Data Num. (C)	48	52	50	50	48	52	50	52	52	50	48	48
Data Num. (D)	48	54	48	56	46	48	48	52	50	48	46	50
Data Num. (E)	50	52	48	52	48	48	50	50	50	48	48	52

Table 2. Comparison of Average Precisions

User	A	B	C	D	E
bigram(%)	65.0	64.2	58.3	64.5	62.7
bigram+tfidf(%)	67.1	65.2	63.1	64.9	64.8

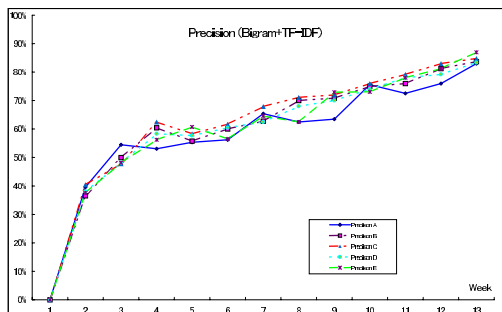


Fig. 5. Precision of bigram+ $tf \cdot idf$

We use open data to evaluate prediction performance, the correct recommendations are judged from rank 60, and we use the next week training data as the correct results. Precision is defined as:

$$\textit{Precision} = \frac{\textit{number of correct recommendations}}{\textit{number of given correct results}} * 100(\%) \quad (11)$$

Figure 5 shows weekly precision of our proposed method.

As a comparison, we performed other experiments, in case of just using the score of extracted features based on the algorithm of Nakagawa, as the baseline of our experiments. Table 2 shows the average precision results.

4 Results and Discussion

In Figure 5 we plot the performance of our proposed method. It is indicated that the average precisions are about 67.1% (User A), 65.2% (User B), 63.1% (User C), 64.9% (User D) and 62.9% (User E). Figure 5 also indicates that the highest precisions are about 83.0% (User A), 83.7% (User B), 84.8% (User C) and 83.3% (User D) and 86.9% (User E), all performance were improved than only using the score of extracted features based on the algorithm of Nakagawa. It is thought that the precision can be more improved, in case of enough learning datasets.

Moreover, the precision in the categories of "Movies" and "Sports" was found higher than the category of "News". It is thought that the description texts of category "News" are less descriptive than those of the categories of "Movies" and "Sports" in the IEPG data.

5 Conclusions and Future Work

In this paper, an idea for predicting users' favorite TV programs is described. The system has proven to have better performance according to our experiment results. We think that our system can provide simpler user interface with enhanced functions to TV watchers because they just need to select the recommendation programs rather than to consider any keywords for selection.

Our goal is to develop a personalized system of TV program recommendations with adaptive capability. We will try to test the performance and put our system to practical use for a computer or a digital television.

Experiments on a large scale including languages other than Japanese will be done in the near future to further verify the accuracy of the present system based on term extraction method. Other algorithms, such as reinforcement learning, SVM and neural networks will also be used for comparison.

References

1. Anna, B., John, Z., Kaushal, K.: Improving Ease-of-Use, Trust and Accuracy of a TV show Recommender. In Proc. of 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems: Workshop on Personalization in Future TV (AH), Universidad de Malaga, Spain (2002) 1-10.

2. Ardissono, L., Gena, C., Torasso, P., et al.: Personalized Recommendation of TV Programs. Lecture Notes in Artificial Intelligence n. 2829. AI*IA 2003: Advances in Artificial Intelligence, Pisa, Italy (2003) 474-486.
3. Johansson, P.: Natural language interaction in personalized EPGs. In Proc. of Workshop notes from the 3rd International Workshop on Personalization of Future TV, Johnstown, Pennsylvania, USA (2003) 27-31.
4. Setten, M. van, Veenstra, M., Nijholt, A.: Prediction strategies: Combining prediction techniques to optimize personalization. In Proc. of TV-02: 2nd Workshop on Personalisation in Future TV, Location Malaga, Spain (2002) 29-37.
5. Xu, J. A., Itoh, T., Araki K.: Action Prediction Method Using Recursive Different and Common Parts Extraction Method with N-gram. Journal of Human Interface Society, Vol.7, No.1, Japan (2005) 55-68.
6. Nakagawa, H., Mori, T.: A Simple but Powerful Automatic Term Extraction Method. In Proc. of 2nd International Workshop on Computational Terminology, COLING-2002 WORKSHOP, Taipei (2002) 29-35.
7. Joachims, T.: A Probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Technical Report, CMU-CS-96-118, Carnegie Mellon University, Pittsburgh, PA (1996).
8. <http://membled.com/work/apps/xmltv/>.

A Hybrid Text Classification System Using Sentential Frequent Itemsets

Shizhu Liu and Heping Hu

College of Computer Science, Huazhong University of Science and Technology,
Wuhan 430074, China
stoneboo@126.com

Abstract. Text classification techniques mostly rely on single term analysis of the document data set, while more concepts especially the specific ones are usually conveyed by set of terms. To achieve more accurate text classifier, more informative feature including frequent co-occurring words in the same sentence and their weights are particularly important in such scenarios. In this paper, we propose a novel approach using sentential frequent itemset, a concept comes from association rule mining, for text classification, which views a sentence rather than a document as a transaction, and uses a variable precision rough set based method to evaluate each sentential frequent itemset's contribution to the classification. Experiments over the Reuters corpus are carried out, which validate the practicability of the proposed system.

1 Introduction

In an effort to keep up with the tremendous growth of the World Wide Web, many research projects were targeted on how to organize such information in a way that will make it easier for the end users to find the information they want efficiently and accurately. Text classification, becoming a major subfield of the information systems discipline in the early '90s, is one such task, which is now being applied in many contexts, ranging from document filtering, to word sense disambiguation, population of hierarchical catalogue of Web resources, and in general any application or selective and adaptive document dispatching.

Recent studies in the data mining community proposed new methods for classification employing association rule mining[1], [2]. All these current associative classifier, to our best knowledge, exploit document-level co-occurring words, which are groups of words co-occurring frequently in the same document[3], [4]: training documents are modeled as transactions where items are words from the document. Frequent words (itemsets) are then mined from such transactions to catch document semantics.

However, assuming document is the unit representing an entire idea, the basic semantic unit in a document is actually the sentence in it. Words co-occurring in the same sentence have semantic association more or less, and convey more local information than the set of words scattering in several sentences of a document.

According to above observations, in this paper, we propose a system for text classification based on two key concepts. The first is the document DB model which treats sentence rather document as the transaction to mine the sentential frequent itemset

(SFI) as the feature of that document. The second concept is the variable precision rough set model based method which is to evaluate each SFI's contribution to the classification. The system consists of four components: 1. A document restructuring scheme that clean noisy info in the document and map the original document into a document DB in which sentence is the transaction where items are words in the sentence. 2. A SFIs generator using Apriori algorithm to mine sentential frequent itemsets, employed as the feature of the matrix document, in the training documents DB. 3. A topic template generator that prune the SFIs and using the remaining ones to construct topic templates. 4. A classifier that sore each SFI's weight in the test document and topic templates using our novel weighting scheme and measure the similarity between them.

The integration of these four components proved to be of superior performance to traditional text classification methods. The overall system design is illustrated in Fig.1.

The rest of this paper is organized as the follows: Section 2 introduces some related work. Section 3 presents the steps of data preparation. Section 4 introduces the document DB model and sentential frequent itemsets mining process. Section 5 introduces SFI pruning method. Section 6 presents our proposed SFI weighting scheme and SFI-based similarity measure. Section 7 discusses the experimental results. Finally, we conclude and discuss future work in the last section.

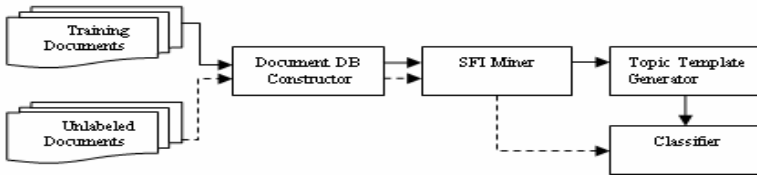


Fig. 1. Text classification system design

2 Related Work

2.1 Text Categorization

Most of researches in text categorization come from the machine learning and information retrieval communities. Rocchil's algorithm[9] is the classical method in information retrieval, being used in routing and filtering documents. Researchers tackled the text categorization in many ways. Classifier based on probabilistic methods have been proposed starting with the first presented in literature by Maron in 1961 and continuing with naïve-Bayes[10] that proved to perform well. ID3 and C4.5 are well-known packages whose cores are making use of decision tree to build automatic classifiers[11], [12]. K-nearest neighbor (k-NN) is another technique used in text categorization[14]. As reported in [15] the use of bigrams improved the text categorization accuracy as opposed to unigrams use. In addition, in the last decades neural networks and vector machines (SVM) were used in text categorization and they proved to be powerful tools[13], [16], [17].

2.2 Variable Precision Rough Set Model

Based on the majority inclusion relation in Pawlak’s rough set model, Ziarko [7] presented a generalized rough set model, named as variable precision rough set.

Given $X, Y \subseteq U$, we define the inclusion of X to Y , denoted as $C(X, Y)$, by:

$$C(X, Y) = \begin{cases} |X \cap Y| / |X|, & |X| > 0 \\ 0, & |X| = 0 \end{cases} \tag{1}$$

$IS = \langle U, A, V, f \rangle$ is the information system of disclosure, where A is the set of attributes, $A = \{a_1, a_2, \dots, a_k\}$. V is the domain of values of A . f is an information function $f : U \times A \rightarrow V$. In the text classification, U is the text collections and A is the feature set. V is domain of the weight values of features in A . R is indiscernibility relation defined on U and $U/R = \{X_1, X_2, \dots, X_N\}$ is a family of R -equivalence classes. $X \subseteq U$ is a subset of interest, and we define α -lower approximation and α -upper approximation by:

$$\begin{cases} \underline{R}_\alpha X = \cup \{X_i \in U / B \mid C(X_i, X) \geq \alpha\} \\ \overline{R}_\alpha X = \cup \{X_i \in U / B \mid C(X_i, X) > 1 - \alpha\} \end{cases} \tag{2}$$

Accordingly, X ’s α -boundary region is defined by:

$$BND_\alpha X = \cup \{X_i \in U / B \mid 1 - \alpha < C(X_i, X) < \alpha\} \tag{3}$$

Where $\alpha \in [0.5, 1]$. It is easy to show this model is equivalent to Pawlak’s model when $\alpha = 1$.

The variable precision rough set model defines the positive and negative regions as areas where the approximate classification with respect to target event with an error frequency less than some predefined level is possible. In other words, the positive region then becomes a region where the event occurred most of the time, and negative region is the region where the event occurred infrequently.

3 Data Preparation

In our approach, to convert text of document into our proposed document DB model which will be introduced in section 4.1, some data preprocessing measures are necessary to be taken to each document.

A sentence boundary detector algorithm was developed to locate sentence boundaries in the documents. The algorithm is based on a finite state machine lexical analyzer with heuristic rules for finding the boundaries. The resulting documents contain very accurate sentence separation, with almost negligible noise.

Finally, to weed out those words that contribute little to building the classifier and to reduce the high dimensionality of the data, a document cleaning step is performed to remove stop-words that have no significance, and to stem the words using the popular Porter Stemmer algorithm[5].

The subsequent phase consists of discovering SFI from each document DB.

4 Document Database and SFI Mining

4.1 Document DB Model

Sentence is a grammatical unit that is syntactically independent and has a subject that is expressed or understood. And the central meaning of a document is stated by organizing the basic idea of sentences. Focusing on mining the local context information in the sentences, we propose a document DB model. In document database model, a word is viewed as an item, a natural sentence is viewed as a transaction, and a document is viewed as a transaction database. The detailed work flow of constructing document DB is illustrated in Fig. 2.

The work presented here takes it a step further toward an efficient way of mining local context information.

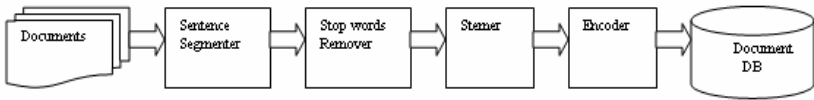


Fig. 2. Process of document DB construction

4.2 SFI Mining

After mapping each document as a transaction DB, we employ the apriori algorithm to extract frequent occurring sets of terms in sentences of each document and use them as that document's characteristic. Compared to documental frequent co-occurring words, sentential frequent words convey more local context information.

The sentential frequent itemsets discovered in this stage of the process are further processed to build the topic templates.

5 Variable Precision Rough Set Model Based SFI Evaluation Method

By merging SFIs of documents which belong to the same category, we get the features of that category spontaneously and we will use these frequent itemsets to construct each category's topic template. For the number of SFI concerning with a category could be very large, how to calculate each SFI's global weight, SFI's contribution to the classification, is the key problem.

We propose a weighting scheme based on variable precision rough set model to evaluate each SFI's global weight, on which we can select the SFIs for each topic template.

Let $F = \{X_1, X_2, \dots, X_{|C|}\}$ be a partition of D , which is the classification of training document according to a set of predefined categories $C = \{c_1, \dots, c_{|C|}\}$. $R \subseteq A$

is a SFI as the condition attributes subset here. According to the Pawlak’s rough set model , $\underline{R}F$ and $\overline{R}F$ are defined by:

$$\begin{cases} \underline{R}F = \{ \underline{R}X_1, \underline{R}X_2, \dots, \underline{R}X_{|C|} \} \\ \overline{R}F = \{ \overline{R}X_1, \overline{R}X_2, \dots, \overline{R}X_{|C|} \} \end{cases} \tag{4}$$

Correspondingly, in variable precision rough set model $\underline{R}_\alpha F$ and $\overline{R}_\alpha F$ are defined by:

$$\begin{cases} \underline{R}_\alpha F = \{ \underline{R}_\alpha X_1, \underline{R}_\alpha X_2, \dots, \underline{R}_\alpha X_n \} \\ \overline{R}_\alpha F = \{ \overline{R}_\alpha X_1, \overline{R}_\alpha X_2, \dots, \overline{R}_\alpha X_n \} \end{cases} \tag{5}$$

A measure was introduced to calculate the imprecision of this classification, which is named as approximate classification quality defined by:

$$\gamma_R(F) = \sum_{i=1}^{|C|} |\underline{R}_\alpha X_i| / |U| \tag{6}$$

Approximate quality denotes the ratio of objects that can be classified to the $F_equivalence$ classes with certainty by SFI R . In other words, $\gamma_R(F)$ measure the consistency degree between the classification by R and F , which may be interpreted as the contribution to classification that SFI makes.

If $\gamma_R(F)$ of all SFIs are calculated and ordered in ascending, we can obtain a concise representation of data by cutting the features whose classification quality value is lower than a threshold that users have predefined.

6 A SFI-Based Similarity Measure

As mentioned earlier, sentential frequent itemsets convey local context information, which is essential in ranking accurately a document’s appropriateness to categories. Towards this end, we devised a scheme to calculate weight of SFI in test document and topic templates and the cosine measure based on the weight is used to performed the classification.

This SFI weighting scheme is a function of three factors:

- the length of the SFI l ,
- the frequencies of the SFI in both document f , and
- the levels of significance (global weight) of the SFI γ , which is presented in section 5.

$$w_{ij} = f_{ij} \cdot l_{ij} \cdot \gamma_j \tag{7}$$

Frequency of SFI is an important factor in the measure. The more frequent the SFI appears in the document or the topic template, the more important the information conveyed by the SFI is. Similarly, The longer the SFI is, the more specific the information conveyed by the SFI is.

The similarity of the test document d_j and the topic c_i is calculated with cosine measure:

$$sim_{SFI}(c_i, d_j) = \frac{\sum_{k=1}^N w_{ik} \cdot w_{jk}}{\sqrt{\sum_{k=1}^N w_{ik}^2} \cdot \sqrt{\sum_{k=1}^N w_{jk}^2}} \quad (8)$$

However, if the similarity between document and topic is solely based on matching frequent itemsets, and no single-terms at the same time, related documents could be judged as nonsimilar if they do not share enough SFIs(a typical case.) Shared SFIs provide important local context matching, but sometimes similarity based on SFIs only is not sufficient. To alleviate this problem, and to produce high quality classification, we combine single-term similarity measure with our itemset-based similarity measure. We used the cosine correlation similarity measure[6], [7], with TF-IDF term weights, as the single-term measure. The cosine measure was chosen due to its wide use in the text classification literature.

The combination of the term-based and the SFI-based similarity measures is a weighted average of the two quantities, and is given by (9).

$$sim(c_i, d_j) = \alpha \cdot sim_{SFI}(c_i, d_j) + (1 - \alpha) \cdot sim_t(c_i, d_j) \quad (9)$$

where α is a value in the interval [0,1] which determines the weight of the SFIs similarity measure, or, as we call it, the Similarity Blend Factor. According to experimental results discussed in Section 7, we found that a value between 0.6 and 0.8 for α results in the maximum improvement in classification quality.

7 Experimental Results

Our set of evaluate experiment was conducted on the well-known Reuters-21578 collections, which are usually split into two parts: training set for building the classifier and testing set for evaluating the effectiveness of the system.

There are many splits of Reuters collection; we select the *ModApte* version. This splits leads to a corpus of 12,202 documents consisting of 9,603 training documents and 3,299 testing documents. All these documents belong to 135 topics. We select ten categories with largest number of corresponding training documents to test our system. There are 6488 training documents and 2545 testing documents in these ten retained categories.

In order to better understand the effect of the SFI-based similarity measure on classification quality, we carry out a set of experiments on the text corpora mention above and compare the experimental results with some of the most well-known methods.

Table 1(the results for the other classification systems are reported as given in [8]) shows a comparison between our classifier and the other well-known methods. The measure used here are precision/recall-breakeven point, micro-average and macro-average on the ten most populated Reuters categories.

Our system proves to outperform most of the conventional method, although Its performance is not every good for three categories, i.e., grain, money-fx, trade.

Table 1. Precision/Recall-breakeven point on ten most populated Reuters categories for SFI-BC and most known classifiers

BEP	SFI-BC	Bayes	Rocchio	C4.5	k-NN	Bigrams
acq	93.2	91.5	92.1	85.3	92.0	73.2
corn	83.2	47.3	62.2	87.7	77.9	60.1
crude	82.0	81.0	81.5	75.5	85.7	79.6
earn	96.8	95.9	96.1	96.1	97.3	83.7
grain	88.2	72.5	79.5	89.1	82.2	78.2
interest	77.3	58.0	72.5	49.1	74.0	69.6
money-fx	74.2	62.9	67.6	69.4	78.2	64.2
ship	85.7	78.7	83.1	80.9	79.2	69.2
trade	75.3	50.0	77.4	59.2	77.4	51.9
wheat	86.5	60.6	79.4	85.5	76.6	69.9
micro-avg	83.7	72.0	79.9	79.4	82.3	73.3
Macro-avg	84.24	65.21	79.14	77.78	82.05	67.07

8 Conclusion and Future Work

Text classification is a key text-mining problem, which is useful to a great number of text-based applications. We presented a system using apriori algorithm to mine the sentential frequent itemsets from document DB and use them as the feature of the corresponding document. The experimental results show that the SFI based classifier performs well and its effectiveness is comparable to most well-know text classifiers.

Although the work presented here is aimed at text classification, it could be easily adapted to Web document as well. However, it will have to take into account the semistructure of Web document. Our intention is to develop a Web document classification system with our approach.

References

1. W. Li and J. Pei. CMAR.: Accurate and efficient classification based on multiple class-association rules. In IEEE international Conference on Data Mining (ICDM '01), San Jose, California, November 29-December 2, (2001).
2. B. Liu, W. Hsu, and Y. Ma.: Integrating classification and association rule mining. In ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD '98), New York City, NY, August, (1998)80-86.
3. M. Antonie and O.R. Zaiane.: Text Document Categorization by Term Association. In Proc. of IEEE Intl. Conf. on Data Mining, (2002)19-26.
4. D. Meretakis, D. Fragoutidis, H. Lu and S. Likotheanasis.: Scalable Association-based Text Classification. In Proc. of ACM CIKM, (2000).
5. M.F. Poter.: An Algorithm for Suffix Stripping, Program, vol.14, no.3, July, (1980) 130-137.
6. G. Salton, A. Wong, and C. Yang.: A Vector Space Model for Automatic Indexing, Comm. ACM, vol. 18, no.11, Nov. 175, 613-620.

7. G. Salton.: Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by computer. Reading, Mas: Addison Wesley, (1989).
8. T. Joachims.: Text categorization with support vector machines: learning with many relevant features. In 10th European Conference on Machine Learning (ECML-98), (1998) 137-142.
9. D. A. Hull.: Improving text retrieval for the routing problem using latent semantic indexing. In 17th ACM international Conference on Machine learning (ECML-98), (1998) 137-142.
10. D.Lewis.: Naïve (bayes) at forty: The independence assumption in information retrieval. In 10th Conference on Machine Learning (ECML-98), (1998) 4-15.
11. W. Cohen and H. Hirsch.: Joins that generalize: text classification using whirl. In 4th International Conference on Knowledge Discovery and Data Mining (SigKDD '98), New York City, USA, (1998)169-173.
12. W. Cohen and Y. Singer.: Context-sensitive learning methods for text categorization. ACM transaction on Information systems, 17(2), (1999)146-173.
13. T. Joachims.: Text categorization with support vector machines: learning with many relevant feature. In 10th European Conference on Machine Learning(ECML-98), (1998) 137-142.
14. Y.Yang.: An evaluation of statistical approaches to text categorization. Technical Report CUM-CS-97-127, Carnegie mellon University, April, (1997).
15. C. M. Tan, Y. F. Wang, and C. D. Lee.: The use of bigrams to enhance text categorization. Journal of Information Processing and Management, (2002).
16. M. Ruiz and P. S.inivasan.: Neural networks for text categorization. In 22nd ACM SIGIR international Conference on Information Retrieval, Berkeley, CA, USA, August, (1999) 282-282.
17. Y. Yang and X. Liu.: A re-examination of text categorization methods. In 22ACM international Conference on Research and Development in Information Retrieval (SIGIR-99), Berkeley, US, (1999) 42-49.

An Approach of Information Extraction from Web Documents for Automatic Ontology Generation

Ki-Won Yeom and Ji-Hyung Park

CAD/CAM Research Center, Korea Institute of Science and Technology,
39-1, Hawolkog-dong, Seongbuk-gu, Seoul, Korea
{pragman, jhpark}@kist.re.kr

Abstract. We examine an automated mechanism, which allows users to access this information in a structured manner by segmenting unformatted text records into structured elements, annotating these documents using XML tags and using specific query processing techniques. This research is the first step to make an automatic ontology generation system. Therefore, we focus on the explanation how we can automatically extract structure when seeded with a small number of training examples. We propose an approach based on Hidden Markov Models to build a powerful probabilistic model that corroborates multiple sources of information including, the sequence of elements, their length distribution, distinguishing words from the vocabulary and an optional external data dictionary. We introduce two different HMM models for information extraction from different sources such as bibliography and Call for Papers documents as a training dataset. The proposed HMM learn to distinguish the fields, and then extract title, authors, conference / journal names, etc. from the text.

1 Introduction

Metadata of general documents such as plain text is useful for many kinds of document processing such as search, browsing, and filtering. Ideally, metadata is defined by the authors of documents and is then used by various systems. However, people seldom define document metadata by themselves, even when they have convenient metadata definition tools. Thus, how to automatically extract metadata from the bodies of documents turns out to be an important research issue.

Text by far is the most prevalent digital information resource. The current information explosion lies primarily on the unprecedented growth in the volume of on-line text information. Examples are postal addresses, bibliography records, and classified ads like CFP (Call for Papers). All these applications have the property that the data has an implicit schema consisting of a set of attributes for example, postal addresses comprise of elements like “street”, “city”, “country”, and “zip” and bibliography records comprise of elements like “author”, “title”, “publisher”, and “pages” and CFP documents consist of a little bit higher regulation such as “introduction”, “date”, “venue”, “contact information”, etc. However, the text string itself is generated by concatenating values of the different attributes without any explicit separator amongst them. The order of attributes is not fixed and not all attributes are present in all instances. These properties make the problem of extracting structure from such text

different from both the general problem of information extraction from natural language documents and the popular problem of generating wrappers to extract structure from HTML documents.

In this paper we present a methodology for information extraction from the Web to build automatically taxonomy of terms and web resources for a given domain. We have developed a tool to automatically segment such text data with small seed set of example segmented records based on Hidden Markov Model (HMM).

2 Anatomy of Bibliography Records and CFP Documents

The bibliography and CFP documents structure that is major motivation are presented in this section.

2.1 Drawn Properties from Observation of Bibliography Records

A prime motivation is arranging bibliographic records for the construction of citation indices like DBLP and Citeseer. This requires extracting the list of references from the end of each research paper and matching each reference to a database of entries. References appear in different formats in different documents. As an example, we show in the Fig.1 below the various forms in which the classical "Information Retrieval" paper is referred.

1. Abasolo JM, Gmez M, "MELISA: An ontologybased agent for information retrieval in medicine", Proceedings of the First International Workshop on the Semantic Web, Lisbon, Portugal, 2000, pp. 73-82.
2. Arens, Y., Chee, C., Hsu, C., and Knoblock, C. Retrieving and Integrating Data from Multiple Information Sources. International Journal of Intelligent and Cooperative Information Systems. Vol. 2, No. 2. Pp. 127-158, 1993.
3. Ricardo Baeza-Yates, Berthier Ribeiro-Neto, Modern Information Retrieval. Addison Wesley, 1999.
4. Wu Chenggang, Jiao Wenpin, Tian Qijia et al, "An information retrieval server based on ontology and multiagent", Journal of computer research & development, 2001, 38(6), pp. 641-647.
5. M. P. Papazoglou, N. Russell, and D. Edmond. A translation protocol achieving consensus of semantics between cooperating heterogeneous database systems. In Conference on Cooperative Information Systems, pages 78-89, 1996.

Fig. 1. An example of bibliography data. This shows that there are many format in describing bibliography in research papers.

They indeed appear very different overall. However, a closer examination reveals that in all references the "title" and "year" fields are the same and in most at least one author is common. Hence, a field level match on title, year and author instead of an approximate record level match would be more accurate. The field extraction problem is non-trivial in this case because of the high variance in the structure of the record. Author is not the first field in all cases. The year field appears within parenthesis for some, at the end for others. Comma separates some fields but comma is also used to separate last names from first names. A dot usually appears at the end of title but also appears after abbreviations. Several fields like location, month and page-numbers are optional.

2.2 Structure of Call for Papers Documents

A second motivating example is CFP documents that can be easily seen in real life. The overall format is presented in Fig.2. As mentioned in previous section, the CFP documents also have quite unstructured properties. However, we can generalize the CFP form with some rule as shown Table 1.

Table 1. The example of CFP structure that we can see

Field Name	Basic Rule	Examples
Conference Name	{NAME_SB + year}, {NAME_SB + "" + year}, {NAME_SB + " " + year}, etc. NAME_SB:= Capital character year := 2-digit 4-digit	CISRA2000 SIGGRAPH '01 INFORMS 2000 CCIS 2000
Conference Date	{day + month + "," + year}, {day + "," + month + "," + year}, {day + DATE_SB + day + "," + month + year}, etc DATE_SB:= "-" "to"	01 November, 2000 14-17, June, 2002 May 05-10, 2001
Avenue	{n_place + <NL> + city + "," + country}, {n_place + "," + city + "," + country}, etc. NL := new line character	Sheraton Resort Hotel Florida, Orlando USA
Due Dates	{D_Day_SB + <NL> + date}, {D_Day_SB + ":" + date}, {D_Day_SB + date}, etc. D_Day_SB:= "Important Dates" "Deadline" "Submission Deadline" "Paper Submission"	Important Dates December 21, 2003: Submission Deadline: April 19, 2001
Contact Information	{CON_SB1 + ":" + number}, {CON_SB2 + ":" + e-mail}, etc. CON_SB1 := "Phone" "Tel" "Fax" CON_SB2 := "E-MAIL"	Phone: 820-628-3901 FAX: 820-747-8517 E-MAIL: cfp@cfp.org

3 Hidden Markov Model

A Hidden Markov Model (HMM) is a probabilistic finite state automaton [8, 10] comprising of a set of states, a finite dictionary of discrete output symbols, and edges denoting transitions from one state to another. Each edge is associated with a transition probability value.

3.1 HMMs for Information Extraction

The HMM model consists of: a set of n states, a dictionary of m output symbols, a $n \times n$ transition matrix A where the i_j^{th} element a_{ij} is the probability of making a transition from state i to state j , and an $n \times m$ emission matrix B where entry b_{jk} denotes the probability of emitting the k^{th} output symbol in state j .

This basic HMM model needs to be augmented for segmenting free text into the constituent elements. Let E be the total number of elements into which the text has to be segmented. Each state of the HMM is marked with exactly one of these E

elements, although more than one state could be marked with the same element. The training data consists of a sequence of element-symbol pairs. This imposes the restriction that for each pair $\langle e, s \rangle$ the symbol s can only be emitted from a state marked with element e . In Fig. 2 we show a typical HMM for address segmentation for information extraction of bibliography. The number of states n is 10 and the edge labels depict the state transition probabilities (A Matrix). For example, the probability of an address beginning with Author Name is 0.92 and that of seeing a City after Book Name is 0.22. The dictionary and the emission probabilities are not shown for compactness. Fig. 3 presents another HMM for handle data extraction for CFP documents. The number of states n is 14 and the edge labels depict the state transition probabilities like Fig. 3.

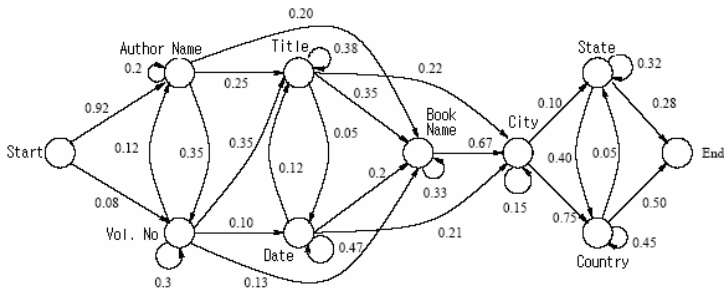


Fig. 2. Structure of a naïve HMM for Bibliography

3.2 Learning Transition and Emission Probabilities

The goal of the training process is to learn matrices A and B such that the probability of the HMM generating these training sequences is maximized. Each training sequence consists of a series of element-symbol pairs. The structure of the HMM is fixed and each state is marked with one of the E elements. This restricts the states to which the symbols of a training sequence can be mapped. Consider two cases. In the first case, there is exactly one path from the start to the end state for all training sequences. The second case is where there is more than one valid path. All HMM structures we discuss in the paper satisfy the first condition of having a unique path. Hence we do not discuss this case further. In the first case, the transition probabilities can be calculated using the Maximum Likelihood approach on all training sequences.

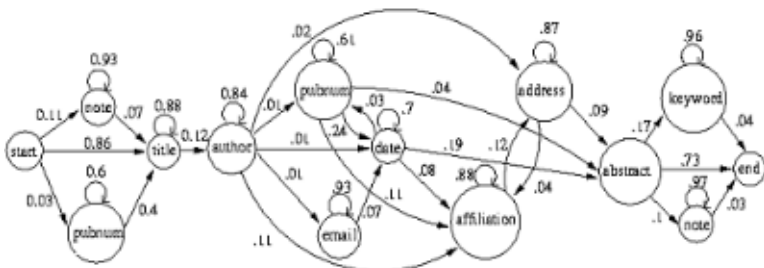


Fig. 3. Another HMM example for CFP data extraction

Accordingly, the probability of making a transition from state i to state j is the ratio of the number of transitions made from state i to state j in the training data to the total number of transitions made from state i . This can be written as:

$$a_{ij} = \frac{\text{Number of transitions from state } i \text{ to state } j}{\text{Total number of transitions out of state } i} \tag{1}$$

The emission probabilities are computed similarly. The probability of emitting symbol k in state j is the ratio of the number of times symbol k was emitted in state j to the total number of symbols emitted in the state. This can be written as:

$$b_{ij} = \frac{\text{Number of times the } k^{\text{th}} \text{ symbol emitted at state } j}{\text{Total number of symbols emitted at state } j} \tag{2}$$

Computationally, training the A and B matrix involves making a single pass over all input training sequences, mapping each sequence to its unique path in the HMM and adding up the counts for each transition that it makes and output symbol it generates.

Nested HMM: In this model, we have a nested structure of the HMM. Each element has its own inner HMM which captures its internal structure. An outer HMM captures the sequencing relationship amongst elements treating each inner HMM as a single state. The HMM is learnt in a hierarchical manner in two stages. In the first stage we learn the outer HMM. In this stage, the training data is treated as a sequence of elements ignoring all details of the length of each element and the words it contains. These sequences are used to train the outer HMM. In the second stage we learn the structure of the inner HMMs. The training data for each element is the sequence of all distinct tokens (word, delimiter, and digit) in the element. An element typically has a variable number of tokens. For example, city-names most frequently have one token but sometimes have two or more tokens. We handle such variability automatically by choosing a parallel path structure (Figure 4) for each inner HMMs. In the Figure, the start and end states are dummy nodes to mark the two end points of an element. All

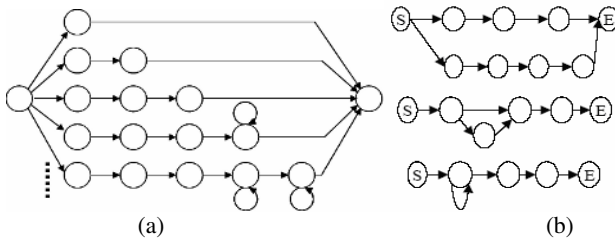


Fig. 4. (a) An example of several length parallel path structure; the proposed model creates as many paths as the number of distinct token counts. This might leave some paths with insufficient training examples resulting in a poorly trained dictionary. (b) An example of merging process; such paths should be merged to its neighboring path as follows: Starting with the longest path, we merge each path to the next smaller path as long as it improves the objective function described in the next paragraph. Then, we choose the one that gives the best value of the objective function. At the end we select the largest path and replace any parallel path within it with a self-loop.

records of length one will pass through the first path, length two will go through the second path and so on. The last path captures all records with four or more tokens.

Feature Selection: We use that the features be arranged in a hierarchical graph. An example taxonomy is shown in Figure 5 where at the top-most level there is no distinction amongst symbols; at the next level they are divided into “Numbers”, “Words” and “Delimiters”; “Numbers” are divided based on their length as “3-digit”, “5-digit” or any other length numbers; and so on. Such taxonomy is part of a one-time static information input to the proposed HMM. The training data is used to automatically choose the optimum frontier of the taxonomy. Higher levels loose distinctiveness and lower levels are hard to generalize and require more training data.

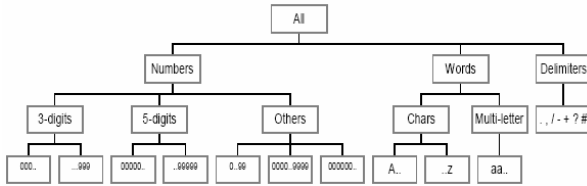


Fig. 5. An example of taxonomy on symbols. One-third of the total data is set aside for validation. Firstly, the dictionary is trained by all symbols at their original detailed level. Next the validation dataset is used to choose the right level of the taxonomy. Starting from the bottom the tree is pruned at various levels in turn and checked the accuracy on the validation data. Finally the highest accuracy sub-tree is chosen. This process does not require training data to be scanned again since the higher level dictionaries can be formed from lower level symbols.

4 Experimental Results

We measure the efficacy of the proposed techniques on bibliography databases and CFP web documents. We compare our results with prior work on information extraction using rule-learning methods. We quantify the benefits of the nested HMM structure and the hierarchical feature selection steps and measure the sensitivity of our results to the number of training instances.

4.1 Experiments on Bibliography Data

The bibliography entries were obtained from a collection of PDF files. The training set had 100 references and the test set had 200 references and the number of elements was 10. The results on our approach and TextMiner are shown in Table 2. Our approach yields an accuracy of about 89% whereas TextMiner although provides high precision suffers on the Post.

4.2 Experiments on CFP Data

For this experiment we visited CFP site and gathered the web documents. This was very tedious time consuming works but since we had to collect the data for training

Table 2. Accuracy comparison of our approach and Rapiere on bibtex format data

Fields	No. of Tokens	Our Approach		TextMiner	
		Prec. (%)	Post (%)	Prec. (%)	Post (%)
Author	1274	88.43	87.86	0.00	0.00
Title	1561	92.27	94.58	87.15	84.39
Conference	1329	83.04	86.40	84.27	78.88
Vol. No.	277	94.64	93.06	0.00	0.00
Editor/Publisher	164	66.02	70.35	64.24	60.30
Page	1158	95.88	97.30	0.00	0.00
Year	1426	100.00	100.00	68.25	65.46
Month	724	100.00	100.00	100.00	100.00
Overall	7913	90.04	91.20	50.50	48.62

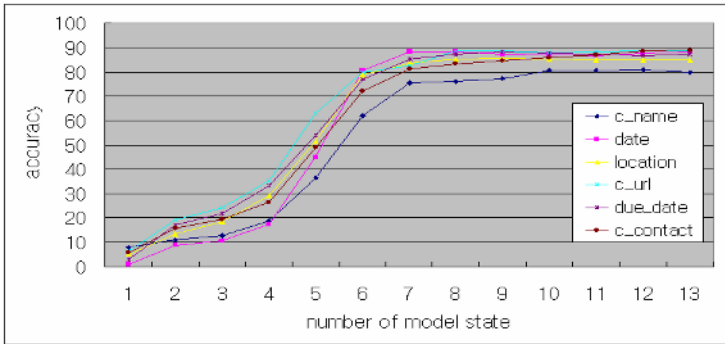


Fig. 6. The experimental results of CFP documents

HMM, we got sufficient data from web. The environment is same as described in previous section. The result shows that as the number of model state increases the accuracy also become high.

5 Conclusion and Future Work

Recently there has been a lot of interest in extracting structure from HTML documents. The text segmentation problem is different and more challenging in that separators between elements are rarely present and data is highly irregular because most of it is human generated - often by different people at different times. In this paper we presented an automated approach for segmenting unformatted text into a set of structured elements.

We proposed a practical approach based on the powerful Hidden Markov Modeling technique. The basic HMM method had to be extended in various ways to solve practical information extraction tasks. We also presented a nested two-level model for learning the structure of the HMM, and a concept hierarchy on the input features for

robust smoothing and automatic feature selection. This approach is a departure from existing rule-based systems for information extraction that rely on heuristics to control the order in which rules are read and extract each element in isolation exploiting only a subset of the information that an HMM can exploit. Experiments on bibliography datasets and CFP documents yield accuracies ranging from 88% to 92%.

Future work in the area include correcting and allowing for spelling mistakes in the data, automatically supplying missing fields for some records and exploiting active learning methods to reduce the amount of training data that needs to be manually tagged.

References

1. Faure, D., Poibeau, T.: First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX. In: the proceedings of the 14th European Conference on Artificial Intelligence, ECAI'2000, Berlin (2000)
2. Sebastiani, F.: Machine Learning in Automated Text Categorization. Vol.34(1). ACM Computing Surveys (2002)
3. Larocca Neto, J.; Santos, A. D.; Kaestner, C.A.; Freitas, A.: Document clustering and text summarization. Proc. of 4th Int. Conf. Practical Applications of Knowledge Discovery and Data Mining (PADD-2000) London: The Practical Application Company (2000) 41-55
4. Mitra, M.; Singhal, A.; Buckley, C. Automatic text summarization by paragraph extraction. In: Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization. Madrid (1997)
5. Rabiner, L.: A tutorial on Hidden Markov Models and selected applications in speech recognition. Vol.77(2). In: Proceedings of the IEEE (1999)
6. Yaari, Y. Segmentation of Expository Texts by Hierarchical Agglomerative Clustering. Technical Report, Bar-Ilan University Israel (1997)
7. Crespo, A., Jannink, J., Neuhold, E., Rys, M., Studer, R.: A survey of semi-automatic extraction and transformation. <http://www-db.stanford.edu/crespo/publications/>
8. Freitag, D., McCallum, A.: Information extraction using HMMs and shrinkage. AAAI-99 Workshop on Machine Learning for Information Extraction (1999) 31-36
9. Liu, L., Pu, C., Han, W.: Xwrap - An xml-enabled wrapper construction system for web information sources. International Conference on Data Engineering (2000) 611-621
10. Stanley B., Andrew M.: Machine learning of event segmentation for news on demand. Vol.43(2). Communications of the ACM (2000)

Improving Text Categorization Using the Importance of Words in Different Categories

Zhihong Deng^{1,2} and Ming Zhang²

¹ National Laboratory on Machine Perception, Peking University, Beijing 100871, China

² School of Electronics Engineering and Computer Science,

Peking University, Beijing 100871, China

zh deng@cis.pku.edu.cn, m zhang@net.pku.edu.cn

Abstract. Automatic text categorization is the task of assigning natural language text documents to predefined categories based on their context. In order to classify text documents, we must evaluate the values of words in documents. In previous research, the value of a word is commonly represented by the product of the term frequency and the inverted document frequency of the word, which is called $TF*IDF$ for short. Since there is a different role for a word in different category documents, we should measure the value of the word according to various categories. In this paper, we propose a new method used to measure the importance of words in categories and a new framework for text categorization. To verify the efficiency of our new method, we conduct experiments using three text collections. The k -NN is used as the classifier in our experiments. Experimental results show that our new method makes a significant improvement in all these text collections.

1 Introduction

In recent years we have seen an exponential growth in the volume of text documents available on the Internet. While more and more textual information is available online, effective retrieval is difficult without organization and summarization of document content. Text categorization is one of the hopeful solutions to this problem. Text categorization is an active research area in information retrieval and machine learning. A growing number of statistical classification methods and pattern recognition techniques have been applied to this area in recent years, including k nearest neighbor classification (k -NN) [1], Naïve Bayes [2], decision trees [3], neural networks [4], and Support Vector Machines [5].

A text categorization task consists of two phases: the training phase and the classification phase. The former includes feature extraction process and text representation. Words are the most common representation form of features in text categorization. In this paper, we adopt this representation method. The vector space model (or VSM for short) has been used as a conventional method for text representation [6]. In this model, each document is considered to be a vector in the word space. Given a document d , VSM represents it by vector $V_d = (v_{d1}, v_{d2}, \dots, v_{dn})$, where v_{di} stands for the value (or weight) of i th word in document d . Let f_i be the i th word, v_{di} is the product of f_i 's term frequent (TF) and f_i 's inverted document frequency (IDF). This model simply counts

the *IDF* of a word by dividing the number of all documents by the number of documents containing the word without considering what type of documents that the word occurs. The motivation behind *IDF* is that words appearing frequently in many documents have limited discrimination power. A major drawback of *IDF* is that it ignores the fact that words play different roles in different document categories. This case often comes out in special domains such as news.

In this paper, we use a new and intuitional method for measuring the relevance of words and categories. Based on the method, we propose a new system (framework) for text categorization. To test our proposed method, we used three text collections and implemented two classifiers. Experiment results indicate that our proposal method show a better performance than the basis system in all text collections.

The remainder of this paper is organized as follows. Section 2 explains the proposed text categorization system in detail. In section 3, we describe the *k*-NN classifiers used in our experiments. Section 4 experimentally compares the performance of our proposed system with that of the basis system on three text collections. The final section summarizes works in the paper and points out future research.

2 The Proposed Text Categorization System

The proposed system consists of two modules as show in Fig. 1: one module for training phrase and the other module for classification phase. The each process of Fig. 1 is explained in the following sections in detail.

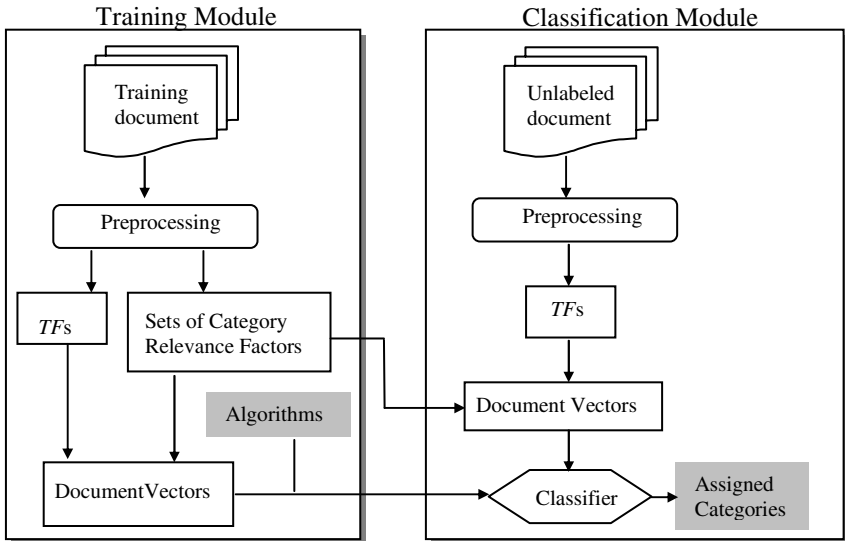


Fig. 1. Overview of the proposal system

Document preprocessing is a procedure that can be divided mainly into four text operations. These operations are lexical analysis of the text, elimination of stopwords,

stemming, and feature (index word) selection. In our system, we adopt other three operations expect for feature selection.

In vector space model, term frequency is one of two factors that consist of value of words in documents. *TF* provides one measure of how well that a word describes document contents. Let $F = \{f_1, \dots, f_n\}$ be the set of total words in the system, and $freq_{ij}$ be the number of times word f_i is mentioned in the text of document d_j . Then, the *TF* of word f_i in document d_j is given by

$$TF (f_i , d_j) = \frac{freq_{ij}}{\max_k freq_{kj}} \tag{1}$$

where the maximum is computed over all words that are mentioned in the text of the document d_j . For the sake of sententiousness, $TF (f_i, d_j)$ is also written as TF_{ij} .

2.1 Category Relevance Factors

In vector space model, inverse document frequency (or *IDF* for short) is the other factor for measuring the value of words in documents. *IDF* provides one measure of how well that a word distinguish relevant documents for non-relevant ones. As mentioned in session 1, *IDF* may be not good enough. By experiment analyses, we find that a word usually occurs in more documents of category c than in those documents that do not belong to c if the word is a keyword (or topic word) of c . Based on above finding, we propose a new method called category relevance factor (or *CRF* for short) for measuring the discriminating power of a word to a category.

Let $C = \{c_1, \dots, c_m\}$ be the set of predefined categories, $F = \{f_1, \dots, f_n\}$ be word set, and $DOC = \cup D_i$ be the set of training documents with labeled categories, where D_i is the set of documents are labeled category c_i . The category relevance factor *CRF* of f_i to c_j is given by

$$CRF (f_i , c_j) = \log \frac{X / Y}{U / V} \tag{2}$$

where X is the number of documents that contain word f_i and belong to D_j , Y is the number of documents that belongs to D_j , U is the number of documents that contain word f_i and don't belong to D_j , V is the number of documents that don't belongs to D_j . For convenience, $CRF(f_i, c_j)$ is also written as CRF_{ij} .

The bigger is the value of *CRF* of a word to a category, the stronger is the ability of the word to discriminate documents of the category from documents of other categories. If word f_i only occurs in documents labeled c_j , we deem that f_i possesses the highest ability for discriminating documents of c_j from non- c_j documents. This is the case that U is equal to zero. In this case, $CRF_{ij} (U = 0)$ should be assigned the biggest values. On the contrary, if word f_i doesn't occur in any training documents labeled c_j , we may consider that f_i plays no or negative role in discriminating documents of c_j from non- c_j documents. This is the case that X is equal to zero. In this case, $CRF_{ij} (X = 0)$ should be assigned the smallest values. It is obvious that the definition of *CRF* is rational and corresponds to the theory of statistics.

For each category c_j , the set of CRF_{ij} (for all $j, 1 \leq j \leq n$) is called the word-relevance set of c_j .

2.2 Document Vectors

In our proposal system, we compute the weight vectors for each document using the *TF* and *CRF*. The value of a word f_i in a document d ($\in D_j$) is calculated as follows:

$$v(f_i, d) = tf(f_i, d) \times CRF(f_i, c_j) \quad (3)$$

Based on formula (3), the vector for document d is represented by $V_d = (v_1, v_2, \dots, v_n)$, where v_i is equal to $v(f_i, d)$. This method for constructing document vectors is called *TF-CRF* scheme. The basic difference between *TF-CRF* scheme and the conventional *TF-IDF* scheme [6] is that the former adopt *CRF* for measuring the discriminating power of words while the latter adopt *IDF*.

3 k -NN Classifiers in Our Experiments

To verify our proposed system, we built two k -NN classifiers. One is based on *TF-IDF* scheme. The other is based on *TF-CRF* scheme. We use the conventional vector space model in text categorization, which represents each document as a vector of word weights and the similarity between two documents is measured by using the cosine value of the angle between the corresponding vectors. [1] has the details of k -NN based on *TF-IDF* scheme. More details about k -NN based on *TF-CRF* scheme is described as follows.

If we construct a classifier with k -NN algorithm and *TF-CRF* scheme, two problems must be solved. One is how to predict the category of an unlabeled document d . The other is how to measure the similarity of d and each training document.

The first problem results from that there is no method for creating the document vector of d in *TF-CRF* scheme. Formula (3) used by *TF-CRF* scheme, concerns the training documents, of which categories we have known in advance. Therefore, it cannot be applied to compute the document vectors of unlabeled documents directly. To solve the first problem, we adopt a technique called supposition-test. Given $C = \{c_1, \dots, c_m\}$ and an unlabeled document d , supposition-test method first assumes that d belongs to each c_i ($c_i \in C$) and computes the document vector V_i of d according to formula (3). Then, it computes the similarity score of d with each training document labeled c_i . Finally, we can get n similarity scores. Based on these scores, k -NN algorithm can be used to predict the category of d .

The second problem is caused by the negative values contained in document vector computed by *TF-CRF* scheme. Because the product of two negatives is positive, we may get incorrect result if we adopt cosine of two document vectors for computing similarity. Therefore, we propose similarity function as follows.

Let $V_1 = (v_{11}, v_{12}, \dots, v_{1n})$ be the document vector of d_1 , $V_2 = (v_{21}, v_{22}, \dots, v_{2n})$ is the document vector of d_2 , and d_1 and d_2 belong to category c . The similarity of d_1 and d_2 is defined as:

$$s_score(d_1, d_2) = \frac{\sum_{i=1}^n \text{sign}(CRF(f_i, c)) \times v_{1i} \times v_{2i}}{\sqrt{\sum_{i=1}^n v_{1i}^2} \sqrt{\sum_{i=1}^n v_{2i}^2}} \tag{4}$$

Where $\text{sign}(x)$ is equal to 1 for $x > 0$, equal to 0 for $x = 0$, and equal to -1 for $x < 0$.

Based on above discussion, we design a k -NN classifier based on *TF-CRF* scheme, which is called *CRFB_k-NN*. Fig. 2 has the details.

Training Phase:
Input: training documents set $D = \cup D_i, 1 \leq i \leq m, D_i = \{\text{document } d \mid d \text{ is labeled category } c_i\}$.
Output: word set $F = \{f_1, f_2, \dots, f_n\}$; $CRF = \{CRF_1, CRF_2, \dots, CRF_m\}$, where CRF_i is the word-relevance set of c_i ; $SVD = \{SVD_1, SVD_2, \dots, SVD_m\}$, where SVD_i is the set of vectors of documents labeled c_i .

Step1. $F = \emptyset, CRF = \emptyset, SVD = \emptyset$.
 Step2. Scan the training documents set D once. Collect words and their term frequencies (*TF*) in D . Insert all these words into F . Let $F = \{f_1, f_2, \dots, f_n\}$.
 Step3. For $i = 1$ to m do:
 For each f_j in F do:
 Compute $CRF(f_j, c_i)$ according to formula (2) and insert $CRF(f_j, c_i)$ into CRF_i ;
 $CRF = CRF \cup \{CRF_i\}$;
 For each document $d \in D_i$ do:
 Compute the document vector of d according to formula (3) and insert its vector into SVD_i ;
 $SVD = SVD \cup \{SVD_i\}$;

Classifying Phase:
Input: $F; CRF; SVD$, an unlabelled document d_{new} , and a nearest neighbor threshold k .
Output: the category of d_{new} .

Step1. $Sim_Scores = \emptyset$; Scan d_{new} once. Collect the term frequency of each word $f_i (\in F)$ in d_{new} .
 Step2. For $i = 1$ to m , do:
 Suppose that d_{new} is a document belonging to c_i . Compute the vector of d according to formula (3);
 For each d in D_i do:
 Compute the similarity score of d_{new} and d according to formula (4) and insert the similarity score into Sim_Scores ;
 Step3. Sort Sim_Scores in score descending order as OSS . The training documents corresponding to top k scores in OSS compose the set of the nearest neighbors. We can find the category c_x , which is most similar to d_{new} , and outputted it as the category of d_{new} .

Fig. 2. CRFB_k-NN

4 Experiments

To test our proposed system, we used three text collections, which are News-groups-18828, Ohscal, and Reuters-10. Each document in these collections has only one category. Reuters-10 consists of documents from top 10 categories with highest frequency of Reuters-21578. In terms of the performance measures, we followed the micro-averaging F_1 measure [7]. Limited by space, we only list the best performances. Table 1 has the details.

Table 1. The comparison of best performance on the three text collections

	Newsgroups-18828	Ohscal	Reuters-10
CRFB_ k -NN	0.856	0.743	0.943
k -NN with $TF*IDF$	0.831	0.69	0.909

5 Conclusions

In this paper, we have presented a new word weighting method called TF - CRF for text categorization. TF - CRF scheme consider the fact that words play different roles in different types of documents and define a suitable function for measuring the importance of words to categories. Based on the scheme, we proposed a text classification framework and implemented a classification system. Comparison experiments on three text collections show the effect of our method. As a future work, we will study better methods for measuring the category relevance factor.

Acknowledgement. This research is supported by the National Natural Science Foundation of China under grant No. 90412010 and ChinaGrid Project. We are also grateful to anonymous reviewers for their comments.

References

1. Yang, Y.: Expert network: Effective and efficient learning from human decisions in text categorization and retrieval. In Proceedings of 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (1994) 13-22
2. McCallum, A., Nigam, K.: A comparison of event models for naïve bayes text classification. In Proceedings of AAAI-98 Workshop on Learning for Text Categorization (1998)
3. Apte, C., Damerau, F., Weiss, S.: Text mining with decision rules and decision trees. In proceedings of Conference on Automated Learning and Discovery, Workshop 6: Learning from Text and the Web (1998) 487-499
4. Ng, H.T., Goh, W.B., Low, K.L.: Feature selection, perceptron learning, and a usability case study for text categorization. In Proceedings of 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (1997) 67-73
5. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In Proceedings of the 1998 European of conference on Machine Learning (1998) 137-142
6. Salton, G., Buckley, C.: Term weighting approaches in automatic text retrieval. Information Processing and Management 24(5) (1988) 513-523
7. Yang, Y., Liu, X.: A re-examination of text categorization methods. In Proceedings of 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (1999) 42-49

Image Copy Detection with Rotating Tolerance

Mingni Wu², Chiachen Lin³, and Chinchun Chang^{1,2}

¹ Department of Information Engineering and Computer Science,
Feng Chia University, Taichung, Taiwan 40724, China
ccc@cs.ccu.edu.tw

² Department of Computer Science and Information Engineering,
National Chung Cheng University, Chiayi, Taiwan 621, China
mnwu@cs.ccu.edu.tw

³ Department of Computer Science and Information Management,
Providence University, Taichung, Taiwan 433, China
mhl in3@pu.edu.tw

Abstract. In 2003, Kim applied DCT technique to propose a content-based image copy detection method. He successfully detected the copies with or without modifications, and his method is the first that can detect the copies with water coloring and twirling modifications. However, Kim's method can only detect copies modified with a 180 degree rotation. When copies are rotated by 90 or 270 degrees, Kim's method fails to discover them. Also, his method can not deal with the copies with minor rotations like 1 degree or 5 degree rotation, and so on. To conquer this weakness, we apply ellipse track division strategy to extract the features and propose our methods. The experimental results confirm that our proposed method can successfully capture block features of an image even if it is rotated to any degree.

1 Introduction

Strictly speaking, content-based image copy detection is a special category of image retrieval methods that can detect the true copies of one image rather than just similar images. In most traditional image retrieval schemes, they retrieve some images which are judged to be similar to the query image according to some characteristics, such as color distribution [1], object features [5], and so on. However, only some of the retrieved results are manipulated copies of the original image, and the rest are different images. In the content-based copy detection scheme, the query objective is to find the copies with some kind of manipulation; such as, contrast adjustment, noise addition, scaling, compression, etc., rather than similar images.

Some researches about copy detection method had been proposed [2, 3, 4]. Kim applied DCT technique to propose a content-based image copy detection method [4]. It successfully detected some manipulated copies but failed to discover the rotation version. To solve the weakness, we propose a novel method which extracts the image features by adopting ellipse track division strategy. Even if the image is rotated, the signatures captured by our methods are almost the same as those of the original. Algorithm's details and further discussions are described in the following sections.

The second section of this paper briefly states a content-based image copy detection method proposed by Kim. Section 3 describes our proposed method. Four

experiments are conducted to compare our performance with Kim's, and experimental results are presented in Section 4. The conclusions are given in the last section.

2 Kim's Content-Based Image Copy Detection Method

Kim used the ordinal measures of AC coefficients to generate image signatures and proposed a content-based image copy detection method [4]. In Kim's method, each image is first divided into an 8×8 equal-sized sub-image, which makes his method independent of input image sizes. Then, these average intensities of the sub-images are derived, and the discrete cosine transform (DCT) is used to produce the 63 AC coefficients. The 63 AC coefficients are ranked by the ordinal measure to form the signature of an image. Kim's matching algorithm is summarized as follows.

1. A gray-level input image is divided into 8×8 sub-images with equal size and their average intensities are generated.
2. The derived average intensities are transformed into a series of coefficients by performing an 8×8 two-dimensional DCT.
3. For ordinal measure of AC coefficients, a 1×63 rank matrix is generated to be the image signature of an input image, which contains the ranks of 63 AC magnitudes.
4. Let the signature of the query image Q , be $q = [q_1, q_2, \dots, q_{63}]$, and that of a test image T , be $t = [t_1, t_2, \dots, t_{63}]$. Then the ordinal measure between two images

$$D(Q, T) \text{ is } D(Q, T) = \sum_{i=1}^{63} |q_i - t_i|.$$

3 Our Proposed Methods

To diminish the amount of source data and eliminate local damage caused by image manipulation, Kim's first step divides an image into 8×8 equal-sized blocks and then derives the average intensity for each block. However, the rotation manipulation could make some border pixels shift to another block. In this case, the average intensities can not accurately represent the relative blocks after rotation manipulation. Figure 1 shows an example of this pixel shifting phenomenon. Figure 1(a) is the original image and its block boundary. Figure 1(b) is the rotated image and its block boundary. It is obvious that some bordering pixels have been shifted to different blocks. This phenomenon causes the follow-up detection process to fail.

In our proposed method, we divide the image by ellipse track instead of square as Kim's method did. Figure 2 shows the same images as Figure 1, and their ellipse track boundaries. Each corresponding ellipse track block covers the same region, so it can keep accurate representation of each block's mean value, even when the image is rotated.

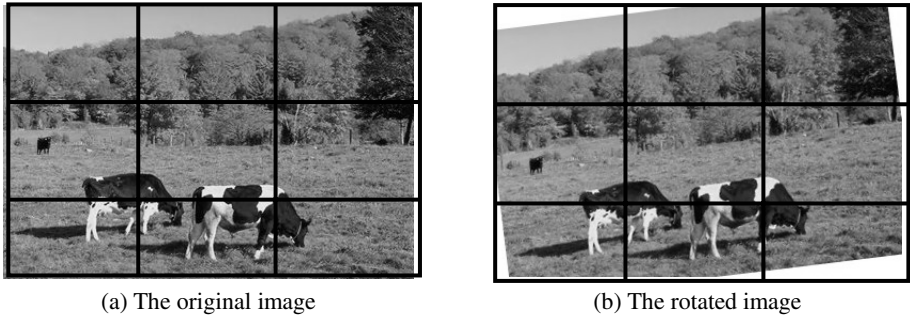


Fig. 1. The pixel shifting phenomenon

Based on the ellipse track division strategy, our proposed methods can be broken down into two phases: feature extraction and comparison phases. In the feature extraction phase, the image features of all images saved in the image database are extracted using our proposed feature extraction algorithm and then stored in the signature database for future use. In the comparison phase, for a suspected image, its feature will first be extracted using the same algorithm as in the feature extraction phase. Then, the extracted feature will be compared with those signatures stored in the signature database using our proposed detection algorithms. Finally, from the comparison, we can determine whether the suspected image is a copy or not.

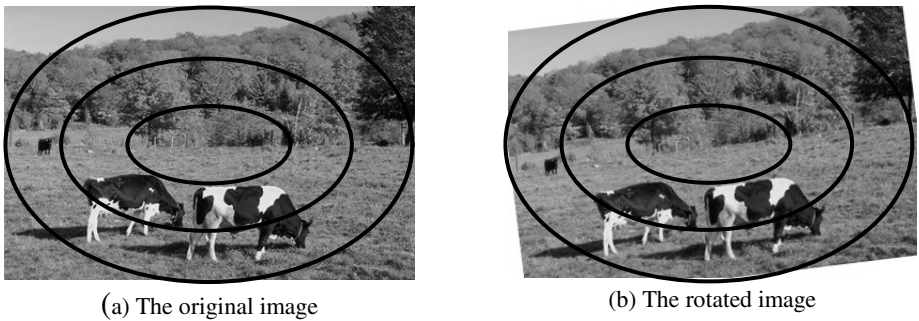


Fig. 2. The elliptical track division strategy vs. the pixel shifting phenomenon

To perform feature extraction, images stored in the image database are converted to YUV color model in advance, and then only Y component is used in the copy detection process. Next, we divide the image into N elliptical track blocks. For each block, we derive the track mean value first. Next, we divide the track mean value by the image mean value to generate its relative mean value, also called signature value. Finally, we can get a feature set which contains N signature values. In our feature extraction phase, once pixel locations are out of the elliptical region they will be discarded, because these regions are almost unimportant in an image.

In order to decide which pixels belong to which division, we propose Equations (1) and (2) as follows. They can determine the track number for each pixel in an image.

In these equations, a pixel p 's coordinate is (x,y) , the coordinate of the image's central is (x_0,y_0) , N is the amount of tracks, w is the image width and h is the image height. If p locates at track n , it will satisfy Equation (1) and not satisfy Equation (2) for $n>1$. If p locates at track 1, it will satisfy Equation (1) only.

$$\frac{(x-x_0)^2}{\left(\frac{wn}{2N}\right)^2} + \frac{(y-y_0)^2}{\left(\frac{hn}{2N}\right)^2} \leq 1 \quad (1)$$

$$\frac{(x-x_0)^2}{\left(\frac{w(n-1)}{2N}\right)^2} + \frac{(y-y_0)^2}{\left(\frac{h(n-1)}{2N}\right)^2} \leq 1 \quad (2)$$

The distance between query image and test image is expressed by Equation (3).

$$D(Q,T) = \frac{\sum_{k=1}^N |Eig_{Qk} - Eig_{Tk}|}{N} \quad (3)$$

Here, Eig_{Qk} is the k th mean signature extracted by Algorithm 1 of the query image, Eig_{Tk} is the k th mean signature extracted by Algorithm 1 of the test image, $D(Q,T)$ is the distance between the query image and the test image. The smaller D value indicates the greater similarity between two images.

4 Experimental Results

In our experiments, we used an image database (IDB), which is downloaded from <http://wang.ist.psu.edu/docs/related/> [6, 7]. Total of 20,000 images in sizes of 384×256 and 256×384 were chosen from the library which are saved in jpeg format. 10 images are randomly chosen from the test database. Each chosen image has gone through various image processes and thus produced 25 processed results to form a manipulated set of images. In total, there are 250 manipulated images for our experiments that were produced by 10 chosen original images.

Four experiments have been conducted to evaluate Kim's method and the proposed methods. The first experiment is to measure the detection ability for manipulated images. 25 processed images derived from Figure 4 (a) are inserted in the IDB and there are 20,025 test images in the IDB. The original image shows as Figure 4 (a) serves as the query image in the detection process. The query process extracts the most similar images with the smallest distance value between the query image and the test images. Table 1 ranks the top 30 images as this result of Kim's method, and our method. From Table 1, it is noted that all rotated images are successfully detected by our method. Conversely, some rotated images, such as 3° rotation, 5° rotation, and so on, can not be detected by Kim's method.

In the second experiment, 250 manipulated images are served as query images and 20,000 images in the IDB are served as test images. We gather statistics which present the frequency of the original image being successfully detected. If the original

image is ranked among the top 100 images for each query test result, then it is treated as a successful detection. The frequency of successful detection with our method is 235 and Kim’s method is 174. We can see the success ratio is up to 94.0% using our method, which is significantly higher than 69.6% provided by Kim’s method.

Table 1. The rank of detected order

Rank	Our method	Kim’s method
1	original image	original image
2	1° rotation	180° rotation
3	compression	horizontal flip
4	Resizing	vertical flip
5	Twirling	Resizing
6	Mosaic	Compression
7	motion blurring	Mosaic
8	Gaussian noising	Gaussian noising
9	90° rotation	motion blurring
10	horizontal flip	saturation increased
11	270° rotation	contrast enhancement
12	180° rotation	hue change
13	saturation increased	Twirling
14	5° rotation	inserting text
15	hue change	Sponge
16	Sponge	mosaic tile
17	vertical flip	other image
18	3° rotation	other image
19	7° rotation	other image
20	9° rotation	other image
21	mosaic tile	other image
22	contrast enhancement	other image
23	inserting text	other image
24	11° rotation	other image
25	other image	other image
26	13° rotation	other image
27	other image	other image
28	other image	water coloring
29	other image	other image
30	other image	1° rotating

Table 2. The successful detection frequencies for different image manipulations

Manipulation type	Our method	Kim’s method
1° rotation	10	7
3° rotation	10	4
5° rotation	10	4
7° rotation	10	1
9° rotation	10	0
11° rotation	10	0
13° rotation	10	0
90° rotation	10	0
180° rotation	10	10
270° rotation	10	0
Compression	10	10
contrast enhancement	8	10
Gaussian noising	10	10
horizontal flip	10	10
hue change	10	10
inserting text	9	10
Mosaic	9	10
mosaic tile	8	10
motion blurring	10	10
Resizing	10	10
saturation increased	10	10
Sponge	7	10
Twirling	9	10
vertical flip	10	10
water coloring	5	8
Total	235	174

The number of tracks plays a crucial role in our proposed methods. A larger track amount can produce more features and it is beneficial for judging the similarity between two images. It also raises the detection time and storage space. More tracks are unfavorable for eliminating local damage caused by image manipulation. In our last experiment, we try to explore the effect of different track amounts. The relative

frequencies of successful detection are 95, 185, 235, 224, and 196 for different track amounts 4, 8, 16, 32, and 64, respective. We can see our proposed method provides better detection performance when the track amount is set to be 16. Therefore, we set the track amount as 16 in the above experiments.

In the signature matching stage, the proposed method gets top 100 images as the matched images. If the target image is among these images, it is presume that the detection is successful. But, can we get a D value as a threshold to differentiate the detection status? The last experiment is to search adaptively for the best threshold value for different N value. The average D values are 7.8620, 8.2924, 8.7931, 9.4253, and 0.1636 for the original images are detected successfully in the top 100 rank when different block number 4×4 , 8×8 , 16×16 , 32×32 , and 64×64 are used. In general, the threshold will vary with the query image, but we can use these values shown in Table 3 as reference values. For instance, 10.7931 could be a suggested threshold for 16×16 blocks. (Add a slight value 2 to the average D value 8.7931)

5 Conclusions

In this paper, we propose a new method which adopts ellipse track division strategy to extract image feature for image copy detection. The method divides an image into several ellipse track blocks and calculates the mean value and variation value of each block to form the characteristics for an image. Experimental results further confirm that our method perform the same tasks as Kim's general variation in assessing noise, resizing, hue change, etc., but also provide outstanding detection capability beyond Kim's method in dealing with variations of rotations. Such superior achievement can successfully enhance the detection capability of existing copy detection mechanisms.

References

1. Y. K. Chan and C. C. Chang.: A Color Image Retrieval Method Based on Color Moment and Color Variance of Adjacent Pixels, International Journal on Pattern Recognition and Artificial Intelligence, Vol. 16, No. 1, (2002)113-125.
2. E. Y. Chang, J. Z. Wang, C. Li, G. Wiederhold.: RIME: A Replicated Image Detector for The World-Wide-Web, Proceedings of the SPIE Multimedia Storage and Archiving Systems, San Jose, CA, Vol. III, November (1998).
3. W. Hsu, T.S. Chua, H.K. Pung.: An Integrated Color-Spatial Approach to Content-Based Image Retrieval, Proceedings of the ACM Multimedia, San Francisco, CA, (1995)305-323.
4. C. Kim.: Content-based Image Copy Detection, Signal Processing Image Communication, Vol. 18, (2003)169-184.
5. C. R. Kim and C. W. Chung.: A Multi-Step Approach for Partial Similarity Search in Large Image Data Using Histogram Intersection, Information and Software Technology, Vol. 45, (2003)203-215.
6. J. Li, J. Z. Wang. Automatic Linguistic Indexing of Pictures by a Statistical Modeling Approach, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, (2003)20.
7. J. Z. Wang, J. Li, G. Wiederhold.: SIMPLicity: Semantic-sensitive Integrated Matching for Picture Libraries, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No.9, 947-963.

Interactive and Adaptive Search Context for the User with the Exploration of Personalized View Reformulation

Supratip Ghose¹ and Geun-Sik Jo²

¹ Intelligent E-Commerce Systems Laboratory,
School of Computer and Information Engineering, Inha University,
Yonghyun-Dong, Nam-Gu, Incheon, 402-751, Korea
SGresearch@gmail.com

² School of Computer and Information Engineering, Inha University
gsjo@inha.ac.kr

Abstract. The explosive growth of information on the web demands effective intelligent search and filtering methods. Consequently, techniques have been developed that extract conceptual information to form a personalized view of the search context. In a similar vein, this system ventures to extract conceptual information as a weighted term category automatically monitoring the user's browsing habits. This concept hierarchy can be served as a thematic search context to disambiguate the words in the user's query to form an effective search query. Experimental results carried out with this framework suggests that implicit measurements of user interests, combined with the semantic knowledge embedded in concept hierarchy can be used effectively to infer the user context and to improve the results of information retrieval.

1 Introduction

Search engine, of late, needs to augment the relevant result for users with improved precision tailored to user individual needs. There are many attempts to classify the web content automatically into taxonomy [2][3], centered with personalized view. PVA [1] is the automatic enhancement of personal view by learning user interests and adapt the personal view to changes in the user interest. This system refined personalized view reformulation of user with periodic updates on user interest changes. In another research with the user profile reformulation, Guach and Trajkova et al. [2] [3] showed the top down hierarchical user profile and profile reformulation with the re-ranked result set as the similarity of the summarized concept and highly weighted concept on the user profile. Although, we maintain the formation of same weighted term concept node, our system conjectures to formulate the weight for query expansion for the modified term concept hierarchy after the user interaction with the system. The "importance" of concepts and relations with respect to the search process can be explicitly stored to the ontology. With that kind of task, ARCH [4], an adaptive agent has been formulated with the aggregated representation of user interests captured by cluster representation and user interaction with concept hierarchy there, helps to summarize the actual search intent. Takings that notion, user interaction to form queries in the ontology hierarchy may decide to form the "central" concept in

which query reformulation can be benefited. In our framework, we view query enhancement as a filtering task in the concept category and therefore delve with the discriminativeness of words [8] to formulate of improvement of search result by the user. The contribution of this research is to provide a dynamic user view reformulation with the modified weight and consequently to find an accurate description of user search intent based on the above discussions. Our research, thereby, attempts to deliver the summarization of the work in the following sections.

2 Derivation of Concept Hierarchy from User Information Space

The search system, in which we apply ontology, relies on a sophisticated indexing process extracting taxonomy of related concepts from the raw documents. In order to do that, we take ODP [Open Directory Project] [9] concept hierarchy [and associated manually classified web pages] as our reference ontology. The webpages for a particular concept in the reference ontology are merged to create a collection of super-documents sd_j and trained to form the index. The collection is obtained by preprocessing with stopword removal and stemming [4]. Essentially, each concept is represented by a vector from the training set and weighted using a variation of vector space term weighting formula [4][6]. The weight of term in concept j

$$wt_{ij} = tf_{ij} * idf_i * cdf_j \tag{1}$$

where tf_{ij} =the total frequency of term i in all training documents for concept j
 idf_i = the inverse concept frequency; cdf_{ij} is the concept document frequency [2]. As shown in the Figure 1, the search system in our architecture uses a query module to

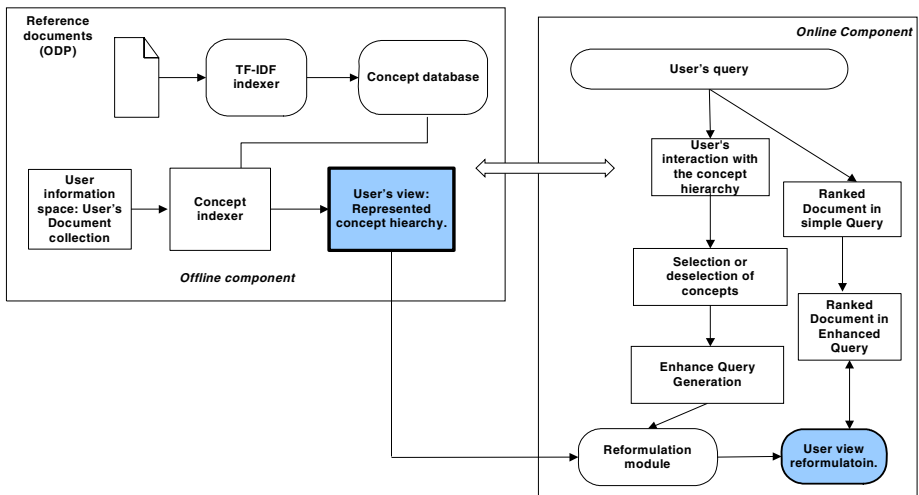


Fig. 1. System Architecture

allow the user to generate queries, an offline module to extract and learn the concept hierarchy representing user’s view from the user information space, and an online module. The online module involves displaying the concept hierarchy to the user, allowing the user to interact over the indexed concept hierarchy as described in section 3. In order to initiate the query generation process, the user enters a keyword query. The query is represented as the same manner as in [6]. Based on the user’s interaction with the system, the system responds by displaying the appropriate portions of the hierarchy. The system matches the term vectors representing each node in the concept hierarchy with the list of keywords typed by user. These nodes, which exceed a similarity threshold, are displayed to the user, along with other adjacent nodes. These nodes are the matching nodes of the concept hierarchy. Thereafter, user interface allows the user to provide explicit feedback by user selection and deselection of nodes and it provides an accurate view formulation of the user considering the context of the desired information need. We employ a variant of Word contribution method [8] for relevance feedback to generate the enhanced query. The pre-computed term vectors associated with each node in the hierarchy are used to enhance the original query as follows:

$$\text{inf}(t, q, n_i) = \text{Sim}(q, n_i(t_{\text{select}})) - \text{Sim}(q'(t), n_i(t_{\text{deselect}})) \tag{2}$$

where, $\text{inf}(t, q, n_i)$ is the influence of the term t for the similarity between query q and the document d_j for the selected (expanded) node n_i under consideration, $\text{Sim}(q, n_i(t_{\text{select}}))$ is the similarity among the query q and document d that is considered fitting by the user upon selection of the category node, and $\text{Sim}(q'(t), d'_{\text{noise}}(t))$ is the similarity among the query q and the terms in the noise documents d for the node n_i that is deselected by the user, excluding the term t of the query q and the document d , both in the fitting and noise document collection. We can define the word contribution as being the difference between the similarities of the query q and the document d , excluding the word w in the querying and in the document. As mentioned by Hoasi et al. [8], the terms with positive contribution, or some of them, are used by the reformulation module for the view actualization. Initially the word contribution of all the terms belonging to the query and the document are calculated using the WC presented previously. The terms t with small $\text{inf}(t, q, n_i)$ values, will be used for the calculation of the *Score* through the equation below:

$$\text{Score}(t) = \text{wgt} \times \sum_{d_j \in \text{fit}(q)} \text{inf}(t, q, n_i) \tag{3}$$

where, wgt is a parameter with a negative value (since the contribution of the extracted word is also negative), and $\text{inf}(t, q, n_i)$ is the influence of term vector to the similarity of the profile p and terms extracted from the selected document d for the node n_i under consideration. On this procedure, the calculated score is regarded as the TF (term frequency) element of the word. Finally, all extracted words and their weights are added to the profile. The value of wgt depends on the characteristics of each collection, and $D_{\text{fit}(q)}$ are the fitting (relevant) documents sent by the user.

As of the formula 3, the information from the noise documents in the deselected category node is considered for the aggregation of the user view hierarchy. When the fitting document is under consideration, the weight of term t is added to the element of the profile vector, which expresses terms t . When the noise document is considered, the weight is subtracted from the element of the profile vector. Separate parameters (wgt) are used for the calculation of $Score(t)$ described in formula 3, depending on the relevance for the collection of documents for the selected node under consideration.

3 Experiments with User View

The goal of this framework is to analyze the users' information need in the weighted concept hierarchy in order to formulate the context of his/her search. Consequently, for experiments, we include top three levels of the ODP concept hierarchy.

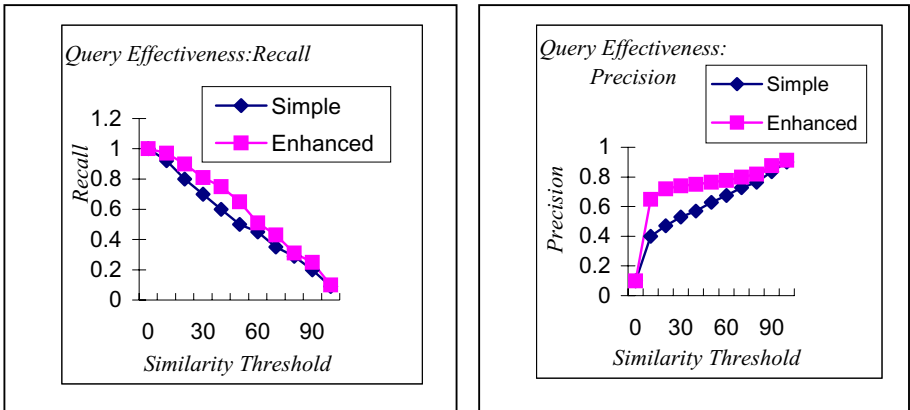


Fig. 2. Average recall and precision for enhanced query on user view versus simple query search using two keywords

User information space is formed by document collections during the browsing and heuristics used by the systems and counts on several factors including the frequency of visits to a page or a site; the amount of time spent on the page (or related pages within a site). In this phase, the URL's, time when visited, and webpages sizes are stored in log file by a proxy server. The webpages pertaining to the user interest collection are preprocessed and vectored similarly as in section 2 and highest weighted 30 words are used to represent the content. Consequently, the documents in the user's information space are classified by comparing the document vector to the representative vector for each concept [3]. The top-similarity values, thus calculated, are then stored in the concept-based index.

The user view is represented by the total weight and number of pages associated with each concept in the ontology. These sets of concepts are further restricted to

include only those concepts that had sufficient training data (20 pages). The view was done on a daily basis for one month for total of six subjects and which three sets of users participated throughout the entire study. For this purpose, we created views consisting of level 1 concepts; views based on both levels 1 and 2, and views based on all three levels.

As an example, for the keyword query a total of 30 documents are collected whose 10 documents related to the Nuclear fusion sense of the term fusion, 10 documents related to the computer equipment sense of fusion and the rest 10 documents discuss aspect of fusion as music related fusion. Thereafter, we compiled a list of keyword queries. The first set of keyword queries contained only one term, such as, fusion and costumes from the ODP ontology. The second set of queries contained two terms, such as, cold fusion, cloths and costumes. We collected a number of “fitting” and a number of “noise” documents to construct our document collection.

For example, if the intent for the search were to find documents about a Nuclear Fusion, the “fitting” documents would contain the Atomic fusion sense of the word fusion, while “noise” documents would contain the computers fusion, or Music related Jazz fusion as shown in Figure 3.

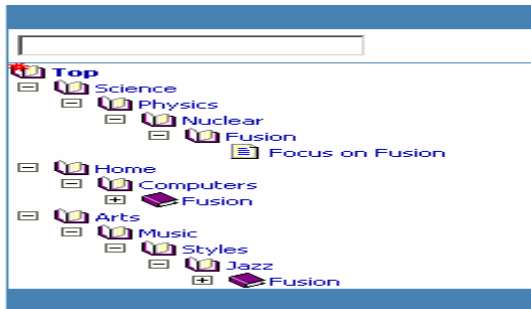


Fig. 3. Representation of the user view hierarchy interface

Using the expanded and deselected nodes, the system generates an enhanced query using the relevance feedback formula, which is described before. Based on the user’s interaction with the concept hierarchy, the enhanced query that is generated accurately represents the user’s intent for the initial query. In the case of enhanced query, we used the query, that is generated after the user interaction over the concept hierarchy based on his/her relevance judgment depicted in Figure 3. The search results were retrieved from the “fitting” and “noise” document collection by using a cosine similarity for matching. We have seen two types of improvements in the search results using the enhanced query. From the user’s perspective, precision is improved since ambiguous query terms are disambiguated by the enhanced query. In addition, we have better recall in the search results since additional query terms retrieve documents that would not be retrieved by using only the original keyword query.

4 Conclusion

We have presented the implementation of query enhancement in this framework based on the user's interaction with a concept hierarchy to offer users a more personalized search experience. First, by browsing, user view that represents user interest by concept. As the user's view is more improved over the concept hierarchy, we can let the user select the concepts to find the context of the query. We have shown the user preference with top-down concept hierarchy in multiple level and we try to add those views in our future work to reformulate the query so as to minimize the gap between search and query formulation. Furthermore, a number of different statistical tests will be conducted to measure the strict validity of the standard performance measures on our framework.

Acknowledgement

This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

References

1. Chen C., Chen M., Sun Y.: PVA: A self -Adaptive Personal View Agent. *Journal of intelligent Information Systems*, (2002)173-194.
2. Ravindran, D. and Gauch S.: Exploiting hierarchical relationships in conceptual search. In *Proceedings of the Thirteenth International conference on Information and knowledge management (CIKM 2004)*, Washinton, DC, Nov. 8-13, 238-239.
3. Trajkova, J. Improving.: *Ontology-Based User Profiles*, M.S. Thesis, EECS, University of Kansas, August, (2003).
4. S. Parent, B. Mobasher, and S. Lytinen.: An adaptive Agent for Web Exploration based of Concept hierarchies, In *Proceedings of the 9th International Conference on Human Computer Interaction*, New Orleans, LA, (2001).
5. G. Salton and M.J. McGill.: *Introduction to Modern Information Retrieval*. McGraw-Hill, NewYork, NY, (1983).
6. J. Rocchio.: Relevance feedback in information retrieval. In G. Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processings*, Prentice Hall, (1971) 313-323.
7. Yan, T. W. and H. Garcia-Molina.: Index structures for information filtering under the vector-space model, *Proceedings of International Conference on Data Engineering*, (1994) 337-347.
8. K Hoashi, K.M.N.I., and Hashimoto, K. 2000b, Query expansion method on word contribution. Berkley CA-USA: ACM SIGIR
9. Open Directory Project. <http://dmoz.org>.

Integrating Collaborate and Content-Based Filtering for Personalized Information Recommendation

Zhiyun Xin^{1,2}, Jizhong Zhao³, Ming Gu², and Jianguang Sun²

¹Department of Computer Science and Technology, Tsinghua University,
Beijing 100084, China

² School of Software, Tsinghua University, Beijing 100084, China
xinzy@csrc.gov.cn

³ Department of Computer Science and Technology, Xi'an Jiaotong University,
Xi'an 710049, China
zjz@mail.xjtu.edu.cn

Abstract. To achieve high quality of push-based information service, in this paper, collaborative filtering and content-based adaptability approaches are surveyed for user-centered personalized information, then based on the above method, we proposed a mixed two-phased recommendation algorithm for high-quality information recommendation, upon which performance evaluations showed that the mixed algorithm is more efficient than pure content-based or collaborative filtering methods, for pure of either approaches is not so efficient for the lack of enough information need information. And moreover we found with large amount registered users, it is necessary and important for the system to serve users in a group mode, which involved merged retrieval issues.

1 Introduction

With fast development of computing technologies and widespread of internet, it become more and more easy to achieve information through network. Search engine provides a valid information search way for users to find what he(he) needs. However with the explosive growth of the Internet and World Wide Web, locating relevant information is time consuming and expensive, push technology promises a proper way to relieve users from the drudgery of information searching. Some current commerce software or prototype systems such as PointCast Network, CNN Newswatch, SmartPush and ConCall serve users in a personalized way[1-6], while adaptability system[7-9] such as GroupLens, MovieLens, Alexa, Amazon.com, CDNow.com and Levis.com are used in many Internet commerce fields. Although kinds of personalized adaptability systems were developed, still many things left unresolved, these problems result in deficiency and low quality of information service as the systems declared. One of most important reasons of which is that single adaptability mechanism such as content-based or collaborative adaptability is difficult to serve kinds of users for their various information needs [4, 7-9, 12-16].

In this paper, Collaborate Filtering and Content-based adaptability approaches are surveyed for user-centered personalized information. Both the two methods are evaluated for information needs for different goals of in adaptability services. As one of the results of the work in the paper, mixed adaptability method can be adopted to

improve the quality services of adaptability services, for pure of either approach is not so efficient for the lack of enough information need information. And moreover we found with large amount registered users, it is necessary and important for the system to serve users in a group mode, which involved merged retrieval issues.

2 Push-Based Information Service

Usually traditional information retrieval system [1] is composed of Retrieval-Enabled Information Source (IS), Indexing Engine (XE), Information Model (IM), Retrieval Engine (RE), Graphic User Interface (GUI), entrance for users to retrieve for information, which includes the order option parameters.

An Adaptive and Active Computing Paradigm (AACP) for personalized information service[3,5,8-10] in heterogeneous environment is user-centered, push-based high quality information service in a proper way, the motivation of which is generalized as R4 Service: the Right information at the Right time in the Right way to the Right person, that is:

$$R^4 : =R \times R \times R \times R \quad (1)$$

The R4 Service serves users in an active and adaptive mode for high quality information timely and correctly, which can adjust the information content according to user's preference dynamically and adaptive to users interest in some degree, automatically retrieve and delivery related information to users with interaction. The R4 service can be described for adaptive and active information retrieval.

In fact, some adaptability methods such as collaborative filtering, content-based filtering can be adopted for the improvement of the adaptive and active information retrieval.

In fact, personalized information service over heterogeneous information source is a matching map R (where R stand for Retrieval) from VIS to VUser with the Condition C and trigger T during the period P, that is:

$$Q=R(V_{User}, V_{IS, C}, T, P; M_{Info}) \quad (2)$$

where Minfo is the information Model for the Retrieval Map, which may be Boolean, Probable, or VSM modal. To serve users adaptively and actively, the system must know users' needs, usually user profile is the basic infrastructure, which decides the quality of adaptive and active service. Except for adaptability and activity, the adaptive and active information retrieval paradigm also builds on the infrastructure[1] of information indexing, information retrieval, information filtering. Moreover, automatically monitoring for retrieval and delivery is another key technology.

3 Basic Approaches of Information Adaptability

In adaptive and active information retrieval system, the most important issue is to improve the adaption performance for the rightness of the R4 system goal.

Information Filtering [2-5,8-10] is the key problem to take into account in adaptive and active information retrieval system, which includes collaborative filtering and content-based filtering.

Pure collaborative filtering systems are not proper when little is known about a user, or he/she has different preferences, while content-based filtering system can't account for community endorsements. Combining collaborative and content-based filtering together can improve the performance for more precise and efficient information adaptability.

3.1 Content-Based Independent Adaptability

Content-based adaptability approach[4-7] is the method that the system adaptively and actively retrieve information for a given and specific user a deep retrieval by the user registered profile, in which profile may be keywords, phrases or even the question his asked, among which GroupLens, MovieLens are the most popular ones. Thus during the information cycle, the system serves users according to the profile and deduces what the user may want more. Content-based filtering make adaptability by matching a user's query especially by user profile, that is:

$$\text{RecDoc}=\text{ContFilter}(\text{Prof}(u),T_0) \quad (3)$$

where Prof(u) is the user profile that defines the user's information interests and preferences, and T₀ is the threshold in what degree the similarity is measured for the content-based independent recommendation.

Content-based adaptability is effective for recommending textual documents since it is able to recommend only those items that are 'understandable' to computers. Content-based adaptability systems analyze the textual information about preferred items and recommend new items by finding items with similar information. Since the content-based method is appropriate when there is rich content information, it has been applied to recommend news articles or web pages. The advantage of content-based adaptability is it can recommend users precise content just according to the information in the user profile. The information retrieval model maybe vector space model (VSM), Boolean Model, or the Probability Model [1]. However when there exist large amount users have similar information needs which may be different expression with different semantic, the demerits of the content-based approach is obvious, if in a pure content-based adaptability mode, the system much retrieval for each users a time which may have the same or similar information need, and moreover each user just got the information according to him own profile itself, and does not share other's information need.

3.2 Collaborative Adaptability

Collaborative adaptability [8-10] is similar with collaborative filtering methods, which gather and retrieve information for a user not just according to the user's profile alone, instead, it retrieves information for a specific user by the group information interest. That is, it builds up the services based on group information preferences, a most popular case is in a n profile members system, n-1 users would like to retrieve the same content, such as a most basic, basic and popular research paper, then we may regard the specific research paper is very important for the group, and then recommend the resource to the left 1 person as a collaborative recommendation. In the collaborative recommendation is based on the assumption that if two users rate certain items similarly, they are likely to have similar tastes and

will rate other items in a similar manner, which can employ k-means, k-medoid, k-modes, k-prototypes, k-NN algorithms to recommend information to a target customer u based on the preferences of neighbors. Collaborative filtering methods utilize explicit or implicit rating from many users to recommend items to a given user, which can be formally defined as:

$$\text{RecDoc} = \text{CollFilter}(U_1, U_2, \dots, U_n, T_0) \quad (4)$$

where U_i ($i=1, \dots, n$) is the user or user profile, and T_0 is the threshold in what degree the similarity is measured for the collaborative recommendation.

Collaborative adaptability is based on the assumption that if two users rate certain items similarly, they are likely to have similar tastes and will rate other items in a similar manner, the advantage of this approach is it may provide wide view of public and popular information according to neighbors users profiles. The obvious disadvantage of the approach is that the system recommend fairly good information relies on enough registered user profiles, another problem is it may recommend “garbage” information to a user profile that he only needs what he subscribed rather than what the system recommended.

4 Trade-Off: Mixed Adaptability

Although the system serves users with high quality information service in a single style, however it is obvious that many users have similar interests in a proper threshold, less or more than the threshold may belongs to a specific categorization. Moreover, when the system disseminates the information to the user one by one, this may lead to great cost to accomplish the work. The group user manage method can reduce the cost to a certain degree.

Usually clustering algorithms such as k-means, k-medoid, k-modes, k-prototypes are the common methods to cluster user profiles in a proper way, which may make the results inaccurate, but in fact, the adaptability system itself just provides service in an acceptable degree, clustering is also suitable to delivery the results in a group mode which can reusing network bandwidth.

To achieve more efficient adaptability results, classifying user profiles into categories for collaborating filtering is a promising approach, upon which a method that combined content-based filtering and collaborating filtering methods together is efficient for high quality information recommendation], which improved the recall and precision in a high degree. In fact, user profile aggregation, clustering and categorization are the most important ways to improve the adaptability and personalization of systems.

For more precise information recommendation for personalized adaptability, in this paper we proposed a mix recommendation algorithm, which in the first stage, it recommends personalized information according to collaborative intelligence, and then in the second stage it recommends by the user’s current interests, thus during the 2-phase, the algorithm considers not only group intelligence but takes a person’s real information needs into account. The mixed recommendation algorithm can be described as following:

```

Status MixRecAlgorithm (UID (i0), &R S )
{ //In Stage I, the system finishes classification of user information need UID as
  the initial recommendation.
Stage I: IN Classification
  for all UID (i)(1< i< N )
    for (1< j< M )
      { extract its indpedent PIS(i).DIS(j)
        normalizing all DIS(i).PIS(j)
        NormVector(j): = Normaliz e(D IS(i). P IS(j)) ;
        Input all NormVector of DIS(I)into SVM-Classifer,
        SVM calculate the Class of D IS(i);
        output Class Ck (1< k< Ck0);
        return the Ck ClassSet CS to PIS(i0).
      }
//In Stage II, the system recommends information nearly according to user's
  actual information need to provide more precise content.
Stage II: Adaptive Recommendation
  for each D IS(j)(1< j< M )
    for each D IS(j). P IS(k)(1< k< K0)
      for each D i ∈ CS (1< i< S0)
        {
          { calculate similarity s of P IS (k)and D i, supposing its
            corresponding vecto r is v and d:
            s: = cos ( v, d ) ;
            if ( s> s+0 )
              R+ (j): = R+ (j) ∪ Di;
            else if (s.0 ≤ s< s+0 )
              R(j): = R(j) ∪ Di;
            otherwise
              discard Di;
          }
          RS+ (j): = R+ (j);
          RS(j): = R(j);
          RS(j): = RS+ (j) ∪ RS(j);
          Return RS(j) to U ID(i0) according to its weight in < Ii,wi> ;
        }
  }
}

```

5 Group-Oriented Results Merge

When retrieve and delivery the results to users one by one, if the network is free enough, the performance is acceptable, otherwise when a large amount of information waiting for delivery, the network will collapse because of congestion. This case occurs frequently especially when the number of users is large enough and the retrieval period is dense. To reduce the cost of delivery and transportation it is necessary to merge similar results into groups by a certain threshold, simply to say, if Query Q₁ and Q₂ have some similarity in respective dimension, such as for Q₁, x ∈ [a, b], y ∈ [c, d], while for Q₂, x ∈ [a - δ , b + δ'], y ∈ [c + η , d - η'], thus for a proper threshold set T = [α, α' ; β, β'], for x, y, there exists a merge M between Q₁ and Q₂.that is:

$$Q_1 = \sigma_{((a \leq x \leq b) \wedge (c \leq y \leq d))}$$

$$Q_2 = \sigma_{((a + \delta \leq x \leq b - \delta') \wedge (c - \eta \leq y \leq d + \eta'))}$$

$$M(Q_1, Q_2) = ((a - \alpha \leq x \leq b + \alpha') \wedge (c + \beta \leq y \leq d - \beta'))$$

In fact, the vector of user profile is not just a 2-dimension space, which is usually high dimension space. It has been proved that the hardness of merge in n-dimension is Co-NP problem, which has no polynomial solution, thus proper approximate solution such as heuristic solution is candidate one.

6 Performance Evaluation

For performance evaluation we based on our implemented platform, an adaptive information retrieval system(AIRS) to evaluate the algorithm we proposed in this paper. The AIRS system is based on J2EE, XML Web Service, and WebSphere (as Application Server) for Self-Adaptive and Active Computing platform aim at scientific researchers to gain useful and fresh information timely and efficiently. We have evaluated the performance in efficiency of system service for user(s), where pure content-based filtering and collaborative information recommendation methods are evaluated. For reasonable evaluations we adopted Reuters21578 dataset <http://www.research.att.com/~lewisreuters21578.html>) to evaluated the efficiency. Table 1 showed the recall and precise of the 3 methods for personalized information recommendation.

Table 1. Precision Evaluation of the MixRecAlgorithm

User Amount	Collaborative (%)	content-based (%)	Mixed(%)
50	43.5	50.3	62.4
100	40.8	48.9	54.3
150	36.2	42.7	48.2
200	35.0	39.2	43.4
250	33.6	35.6	41.1

Table 1 showed that when using mixed approach, the precision is improved obviously than pure collaborative or pure content-based. However as a inverse ratio between precision and recall. the recall results of the mixed recommendation approach is not so good when precision is improved, this is the normal phenomenon in information retrieval system.

7 Conclusion

Push-based high quality information service has become a new paradigm in information field. To achieve high quality of push-based information service, in this paper, Collaborate Filtering and Content-based adaptability approaches are surveyed for user-centered personalized information. Both the two methods are evaluated for information needs for different goals of in adaptability services. As one of the results

of the work in the paper, mixed adaptability method can be adopted to improve the quality services of adaptability services, for either approaches is not so efficient for the lack of enough information need information. And moreover we found with large amount registered users, it is necessary and important for the system to serve users in a group mode, which involved merged retrieval issues.

Acknowledgement. The work was supported by the National High-Tech Research and Development Plan of China under Grant No.2003AA143010, No.2003AA414031, 2004AA413010.

References

1. G.Salton and M.J.McGill.: Introduction to Modern Information Retrieval, McGraw-Hill, New York, (1983).
2. Min, Sung-Hwan, Han, Ingoo.: Detection of the customer time-variant pattern for improving recommender systems, *Expert Systems with Applications*, 28(2), (2005) 189-199.
3. Basu, C., Hirsh, H., & Cohen, W.: Recommendation as classification: using social and content-based information in recommendation, *Proc of the 1998 workshop on recommender systems*, Menlo Park, CA: AAAI Press, (1998) 11–15.
4. Lalmas and Mounia.: Logical models in information retrieval: introduction and overview, *International Journal of Information Processing and Management*, 34(2), (1998) 19-33.
5. Underwood, George M., Maglio, Paul P., Barrett, Rob.: User-centered push for timely information delivery, *Computer Networks and ISDN Systems*, 45(3), (1998) 33-41.
6. Cho Yoon Ho, Kim Jae Kyeong, and Kim Soung Hie.: A personalized recommender system based on web usage mining and decision tree induction, *Expert Systems with Applications*, 35(12), (2002) 329-342.
7. G. Linden, B. Smith, and J. York.: Amazon.com recommendations: item-to-item collaborative filtering, *Internet Computing IEEE*, 7 (1), (2003) 76–80.
8. H.C. Chen and A.L.P. Chen.: A music recommendation system based on music data grouping and user interests, *Proc of the ACM International Conference on Information and Knowledge Management*, Atlanta, GA, November, (2001) 231–238.
9. Billsus, D., & Pazzani, M. J.: Learning collaborative information filters, *Machine learning proc of the fifteenth international conference (ICML'98)*, (1998).
10. Breese, J. S., Heckerman, D., & Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering, *Proc of the 14th conference on uncertainty in artificial intelligence (UAI-98)*, (1998) 43–52.
11. Cheung, K., Kwok, J. T., Law, M. H., & Tsui, K.: Mining customer product ratings for personalized marketing, *Decision Support Systems*, 35, (2003) 231–243.
12. Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted collaborative filtering for improved recommendations, *Proc of the 18th national conference on artificial intelligence (AAAI-2002)*, Edmonton, Canada, July, (2002)187–192.
13. Schafer, J. B., Konstan, J. A., & Riedl, J.: Electronic commerce recommender applications”, *Data Mining and Knowledge Discovery*, 5(1/2), (2001) 115–153.
14. Song, H. S., Kim, J. K., & Kim, S. H.: Mining the change of customer behavior in an internet shopping mall, *Expert Systems with Applications*, 21, (2001) 157–168.
15. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J.: Analysis of recommendation algorithms for e-commerce, *Proc of ACM E-Commerce 2000 conference*, 158–167.

Interest Region-Based Image Retrieval System Based on Graph-Cut Segmentation and Feature Vectors

Dongfeng Han, Wenhui Li, Xiaomo Wang, and Yanjie She

College of Computer Science and Technology,
Key Laboratory of Symbol Computation and Knowledge Engineering of
the Ministry of Education, Jilin University, Changchun 130012, China
jlu_hdf@126.com

Abstract. In this paper, an interest region-based image retrieval system (IRBIR) that combines similarity contributions from interest regions specified by user in images to form a single value for measuring similarity between images is proposed. The interest region-based framework utilizes the segmentation result to capture the higher-level concept of images. A novel image segmentation based on Graph-Cut is proposed for the final result. The segmentation method in this paper is fast and accurate enough for the real-time image retrieval demand than previous region-based methods. Experimental and comparison results, which are performed using a general purpose database containing 2,000 images, are encouraging.

1 Introduction

Currently the researchers have already designed many content based image retrieval systems. However, most of these researches are focus on global features. While these researches establish the basis of content-based image retrieval (CBIR), the retrieval performance is still far from users' expectations. Users accessing a CBIR system often look for images containing particular objects, possibly arranged in a specific spatial organization.

Region-based image retrieval systems aim to improving the performance of content-based search by segmentation each image into a set of "homogeneous" regions. Thus, similarity between images is assessed by computing similarity between pairs of regions and then combining the results at the image level. Region-based image retrieval systems attempt to overcome the drawback of global features by representing images at object-level, which is intended to be close to the perception of human visual system. There have been many region-based image retrieval systems, such as Netra^[1], Blobworld^[2], Simplicity^[3], Walrus^[4] and Windsurf^[5] etc. For considering the difference of the important degree in the region of the image, in recent years some researchers put forward according to many image retrieval methods based on regions of interest. This method not only makes use of the local characteristic of the image, but also removes the disturbance of unimportant information, generalizing a more accurate result.

Inspired by Blobworld, this paper presents an interest region-based image retrieval system. There are two main contributions of our system. The first contribution is an

efficient image segmentation algorithm. The second contribution is the interest region-based image retrieval framework.

2 Overview

The system can be divided into two parts: one is Client and the other is Server. The whole procedure follows

Step 1: A user should specify an interest region which means an image segmentation procedure. In this step, an efficient and accurate segmentation method based on Graph-Cut is proposed.

Step 2: The interest region specified by the user is send to Server by Internet.

Step 3: Features are extracted based on color, shape and position.

Step 4: The similarity of the feature vectors is computed, and find the top10 images as the searching results.

Step 5: A user feedback is used to improve the results.

In the following of this paper, we will concern on the new image segmentation method, features extraction method and the query method respectively.

3 Graph-Cut Based Image Segmentation

The interest regions are very important to the final results. An efficient segmentation method is introduced and the segmentation result can be controlled by users themselves.

3.1 Basic Segmentation

The segmentation task can be viewed as labeling problems with label 1 representing Object and 0 otherwise. Finding the most likely labeling translates to optimizing an energy function. Because a pixel always has the similar value with its neighbors, we can model the optimization problem as a MRF like in [6]. A graph is constructed and the Potts Energy Model (1) is used as the minimization target.

$$E(G) = E_{data} (v_i) + E_{smooth} (v_i, v_j) \quad (1)$$

$i \in V$ $\{v_i, v_j\} \in N$

Our goal is to minimize the energy function (1) and make it adapt to human vision system. In [7] authors give the construction of the graph in detail. A different way is used to construct the graph in this paper because its construction way is so complex that in practice it will be inconvenient and it will influence the speed.

First a k-Means clustering method is applied on the seed region including “O” and “B”. We use the algorithm described in [8] which performs well in practice. Then, for each node v_i , the minimum distance from its color $color(i)$ to foreground and background clusters are computed as

$$d_i^O = Min_{j \in O} \|color(i) - kMeans_j\| \quad (2)$$

$$d_i^B = \underset{j \in B}{\text{Min}} \|color(i) - kMeans_j\| \quad (3)$$

So the energy of the first term is

$$\left. \begin{aligned} E_{data}(v_i = 1) &= \left\{ \begin{array}{ll} 0 & \text{if } v_i \in B \\ CONST & \text{if } v_i \in O \\ \frac{d_i^O}{d_i^B + d_i^O + \varepsilon} & \text{if } v_i \neq 1 \text{ or } 0 \end{array} \right\} \\ E_{data}(v_i = 0) &= \left\{ \begin{array}{ll} CONST & \text{if } v_i \in B \\ 0 & \text{if } v_i \in O \\ \frac{d_i^B}{d_i^B + d_i^O + \varepsilon} & \text{if } v_i \neq 1 \text{ or } 0 \end{array} \right\} \end{aligned} \right\} \quad (4)$$

Where $v_i = 1$ or 0 means that this node belongs to Object or Background.

The second term $E_{smooth}(v_i, v_j)$, which is define as follow

$$E_{smooth}(v_i, v_j) = e^{-\frac{\|I(i)-I(j)\|^2}{\sigma I}} \times e^{-\frac{\|L(i)-L(j)\|^2}{\sigma L}} \quad (5)$$

Where $I(i)$ is the color of the node v_i , $L(i)$ is the location of the v_i . So once a user specifies the seeds of Object and Background then a graph can be constructed as described above. Then using the Max-Flow algorithm (a more efficient implementation is described in [9]), we can efficient minimize the energy. The output is divided into two parts: one is Object the other is Background.

4 Extraction of Features

After segmentation interest regions are obtained. For each interest region, we compute its color, position and shape for this region's features, which form the feature vectors of this region. Below the details of the calculation features are given.

4.1 Color Features

We use HSV space which has a better fit to the human vision system. We treat the hue-saturation-value (HSV) color space as a cone: for a given point (h; s; v), h and s are the angular and radial coordinates of the point on a disk of radius v at height v. All coordinates range from 0 to 1. Points with small v are black, regardless of their h and s values. The Cartesian coordinates of points in the cone ($c1=s \times \cos(2\pi h)$; $c2=s \times \sin(2\pi h)$; $c3=v$) can now be used to find color differences. For each region, the average color describes its color features.

4.2 Position and Shape Features

For each region, the center of gravity is used to describe the region’s position features. The position features used are described below.

$$X_c = \frac{\sum_x \sum_y f(x, y)x}{\sum_x \sum_y f(x, y)} \quad Y_c = \frac{\sum_x \sum_y f(x, y)y}{\sum_x \sum_y f(x, y)} \tag{6}$$

Where (x, y) are pixel coordinates and f (x, y) is set to 1 for points within the boundary and set to 0 elsewhere. Normalization (6) and get a normalized position feature as (7).

$$P_{norm} = \left(\frac{X_c}{Width}, \frac{Y_c}{Height} \right) \tag{7}$$

The shape features used are enumerated below:

$$\rho = (Number\ of\ pixels\ in\ the\ region / Number\ of\ pixels\ in\ the\ image) \tag{8}$$

Where ρ is the area of the region, which reflects the size of the region.

$$\sigma = \frac{\sqrt{\sum_{(x,y \in Region)} (x - X_c)^2 + (y - Y_c)^2} / Number\ of\ pixels\ in\ the\ region}{Max(Width, Height)} \tag{9}$$

σ is the dispersion degree, which reflects the degree of dispersion of the region relative to the gravity center.

$$\begin{aligned} u_x &= \sum_{x \in Region} (x - X_c)^2 \\ u_y &= \sum_{y \in Region} (y - Y_c)^2 \\ u_{x,y} &= \sum_{x,y \in Region} (x - X_c)(y - Y_c) \end{aligned} \tag{10}$$

(10) is used to calculate the eccentricity in (11).

$$e = \frac{u_x + u_y - \sqrt{(u_x - u_y)^2 + 4u_{x,y}^2}}{u_x + u_y + \sqrt{(u_x - u_y)^2 + 4u_{x,y}^2}} \in [0, 1], \tag{11}$$

The ratio of the major to the minor axis is called eccentricity of the region.

5 Similarity Measure

Gaussian function is used to measure the similarity between the two regions. By adjusting each feature vector variance, we can normalize different feature vectors and get the similarity S between 0 and 1. Finally the query results can be obtained by order S.

The similarity between two regions can be calculated easily using (12).

$$S_{region}(i, j) = \sum_{r=1}^4 W_i S_i, \sum_{r=1}^4 W_i = 1 \quad (12)$$

Where S_i denotes the similarity of color, position and shape respectively. W_i can be used during the initial and the feedback procedure by the user.

In the system, the user composes a query by submitting an image to the segmentation/features extraction algorithm in order to see its representation, selecting the region to match, and finally specifying the relative importance of the region features. Once a query is specified, we score the database image based on how closely it satisfies the query.

6 Experimental Results

First the segmentation results are shown in Fig. 1. One is the input images and a user should specify the Object and Background using red line and green line. The other is the segmentation results. The average time for segmenting an RGB color image of size 481×321 is less than 2 seconds.

The image database contains 2000 images including people, scene, building, car, dinosaur, elephant, flower, house, food and mountain respectively. The retrieval results are shown in Fig. 2.

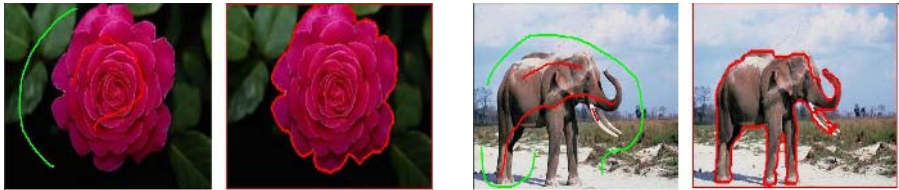


Fig. 1. Segmentation results for flower and elephant



Fig. 2. Retrieval results for flower and elephant

7 Conclusions

In this paper, an interest region-based image retrieval system (IRBIR) is proposed. The main contribution of this paper is a powerful image segmentation method and

image retrieval framework which includes: interest regions selection, features extraction and the similarity measure. The results of our system are promising.

References

1. Ma Weiyang, Manjunath B. S. Netra.: A Toolbox for Navigating Large Image Databases. *Multimedia Systems*, Vol. 7, No. 3, (1999) 184-198
2. Chad Carson, Serge Belongie, Hayit Greenspan, Jitendra Malik.: Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, (Aug 2002) 1026-1038
3. James Ze Wang, Jia Li, Gio Wiederhold.: SIMPLiCITY: Semantics-Sensitive Integrated Matching for Picture Libraries. *IEEE Trans. Pattern Anal. Mach. Intell.* Vol 23, no 9 (2001) 947-963
4. Apostol Natsev, Rajeev Rastogi, Kyuseok Shim.: WALRUS: A Similarity Retrieval Algorithm for Image Databases. *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 3, (March 2004) 301-316
5. S. Ardizzoni, I. Bartolini, and M. Patella.: Windsurf: Region-Based Image Retrieval Using Wavelets. *Proc. DEXA Workshop*, (1999) 167-173
6. Y. Boykov, O Veksler and R Zabih.: Faster approximate energy minimization via graph cuts. In *IEEE transactions on Pattern Analysis and Machine Intelligence (PAMI)*, Vol 23, no. 11. (Nov 2001) 1222-1239
7. Boykov Y. and Jollymap.: Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. In *Proceedings of ICCV* (2001)
8. Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D Piatko, Ruth Silverman, Angela Y Wu. : A Local Search Approximation Algorithm for k-Means Clustering *Proceedings of the 18th Annual ACM Symposium on Computational Geometry* (2003)
9. Y. Boykov and V. Kolmogorov.: An Experimental Comparison of Min-Cut/Max-Flow Algorithm for Energy Minimization in Vision. In *3rd Intl. Workshop on Energy minimization Methods in Computer Vision and Patten Recognition(EMMCVPR)*. Springer-Verlag. (Sep 2001)

A Method for Automating the Extraction of Specialized Information from the Web

Ling Lin¹, Antonio Liotta¹, and Andrew Hippisley²

¹Department of Electronic Systems Engineering,
University of Essex, Colchester, CO4 3SQ, UK
{llini, aliotta}@essex.ac.uk

²Department of Computing, University of Surrey, UK
a.hippisley@surrey.ac.uk

Abstract. The World Wide Web can be viewed as a gigantic distributed database including millions of interconnected hosts some of which publish information via web servers or peer-to-peer systems. We present here a novel method for the extraction of semantically rich information from the web in a fully automated fashion. We illustrate our approach via a proof-of-concept application which scrutinizes millions of web pages looking for clues as to the trend of the Chinese stock market. We present the outcomes of a 210-day long study which indicates a strong correlation between the information retrieved by our prototype and the actual market behavior.

1 Introduction

The Web has now become the major source of information, a gigantic database in constant growth which disseminates news and documents at an incredible speed. On the other hand, people can gain access to the information they are interested in thanks to sophisticated search engines providing keyword matching and thematic filters.

Despite its success, the web looks increasingly more like a 'black hole' from where the information is difficult to retrieve. The user is more often overwhelmed by documents that are difficult to digest in 'real-time' by the 'human' user of the Internet. This is because the tools currently available are not suited to the extraction of cognitive knowledge. With such a powerful, dynamic and large-scale database at everybody's fingertips, tools allowing automatic, intelligent scans of the web are needed. One would expect to extract knowledge, which is aggregate information, from the web rather than raw documents requiring human interpretation.

Armed by the objective of building an intelligent information retrieval system, we have developed a novel methodology for extracting specialized information from the Web. We present such methodology by means of a case study, where we extract 'sentiments' of the Chinese Stock Market resulting from a daily scan of thousands of Web pages that are then interpreted by our intelligent agent system. We combine the use of self-learning techniques with simple statistical processing to establish whether the web page under scrutiny gives positive or negative information about the stock market (we actually use a range of 10 marks). We then aggregate the information extracted from all web pages scrutinized in one day and come up with a market

sentiment, which is an indication of the status of the market. This high-level information is then offered to the user – as opposed to the thousands of web pages that the user couldn't possibly digest in a single day. Finally, in order to verify that the information we are providing is accurate, we compare our 'sentiments' with the actual index of the stock exchange.

We propose a new information retrieval [1] technique which extracts relevant information from documents. In our work we make use of a Chinese language processing algorithms to address their inherent word segmentation problem not present in Western languages, adopting a dictionary-based maximum forward match approach proposed [2]. We develop a case study focused on the extraction of financial knowledge out of Chinese web sites that report news about the Chinese market. Cross-validation is, hence, performed with the Shanghai Stock Exchange. The method is, however, more general and could be used for the extraction of any other specialized information in any other language. To apply our method to other contexts one should just select its three dictionaries according to the language, specialized terminology and specific semantics to be used to produce the aggregate information.

2 Methodology

2.1 Overview

In order to evaluate our methodology, we developed an intelligent agent system case study, the Chinese Language Market Sentiment System, which extracts from the web semantically-rich information about the Chinese stock market. The idea lies on quantifying market sentiments which are expressed in Chinese financial news. Market sentiments are the perception of traders as to how good or bad the stock market is. They can be used to indicate the demand or the lack of demand for financial instruments. If the feeling of stock market can be expressed numerically, the market sentiments expressed in financial news can be treated as daily series and compared with the daily series of actual stock market values [3]. Therefore, the strategy goes as following: first, extract market sentiments from web pages in the Internet. Then, correlate these market sentiments into daily series and compare them with daily series of Shanghai Stock Exchange Composite to examine if there is a potential correlation between them. In our case the correlation between sentiments and stock market trend is used to validate our methodology.

Fig 1 shows the high-level process including three intelligent agents, web spider agents, html parser agents and language processing agents. Web spider agents are responsible for extracting the volumes of financial news HTML documents available in the Internet. Starting from a 'seed' link, the yahoo stock news web site in this case, web spider agents follow breadth-first searches through the links in web pages across the Internet. Web spider agents have two main tasks. One is to feed HTML source documents to HTML parser agents; the other is to detect URLs and fetch them back to crawl through the Internet. In the mean while, visited URLs should be filtered before feeding back to URL buffer queue.

The Html parser agents convert information implicitly stored as an HTML structure into information explicitly stored as an XML structure for further

processing. For each HTML source document, a title, content, a written date and an author are extracted to construct a content xml, which is stored according to the written day to generate the daily series of market sentiment. Both the title and the content are analyzed by language processing agents.

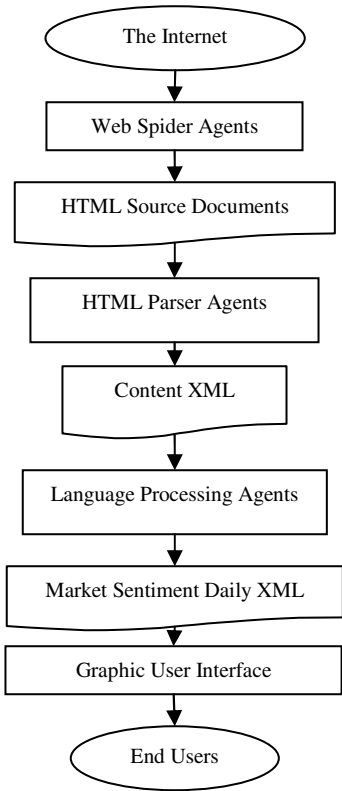


Fig. 1. High Level Process

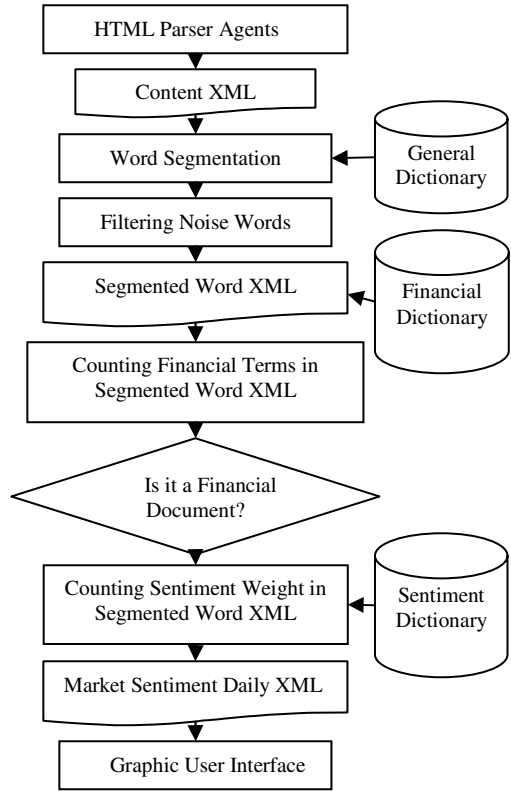


Fig. 2. Language Processing Agent

The language processing agents are dictionary-based and auto-training, which are in charge of identifying market sentiments from content xml into market sentiment daily series. As demonstrated in fig 2, this is done through three steps. First, segment content xml into segmented word xml and filter ‘noise’ words, such as ‘is’, ‘am’, which can be found in virtually every sentence and are, therefore, useless for language processing purposes. Second, look up on the financial dictionary to determine whether the web page article is finance related. Then filter all of content xml which are not finance related. Last, collect market sentiments in segmented word xml, such as ‘increase’, ‘rise’, ‘decrease’, etc, by searching the semantic dictionary, and sum document sentiment weights according to the written day. If there is ‘not’ or ‘no’ in a sentence, we skip all sentiment words in the sentence. Finally, compare a daily series of market sentiments with that of the Shanghai Stock Exchange Composite. This is achieved through three dictionaries, general dictionary, financial dictionary, and semantic dictionary.

2.2 Natural Dictionary for Word Segmentation

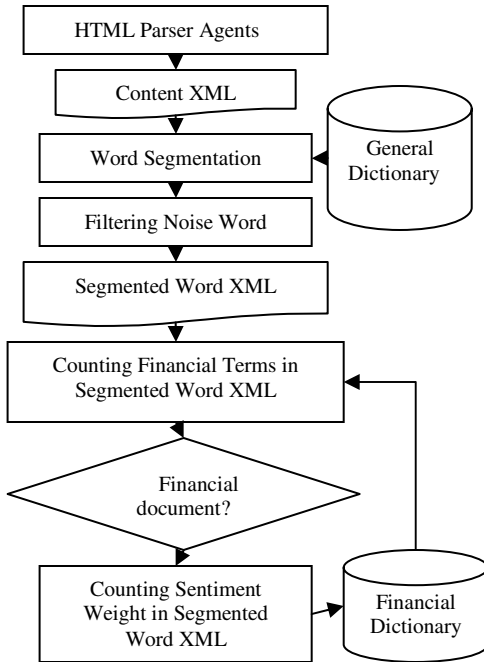


Fig. 3. Auto-training Specialized Dictionary

Word Segmentation is required in most East Asia natural language processing System since they do not have built-in delimiters to mark the boundaries of multi-character terms of phrases.

The first phase of Language Processing is dictionary-based word segmentation whose aim is to segment the titles and contents in content xml by looking up a general Chinese Dictionary. Content xml is scanned sequentially and the maximum matching word from the general Chinese dictionary is taken at each successive location. A maximum matching algorithm is a greedy search routine that walks through a sentence trying to find the longest string of words starting from a given point in the sentence that matches a word entry in the general Chinese dictionary. The string having the highest match is then taken as indexing tokens and shorter tokens within the longest matched strings are discarded.

2.3 Auto-training to Generate Specialized Dictionary

This phase is to judge whether the content xml is financial related or not by summing financial term’s weights (Fig.3). First, we look up the financial term dictionary. We then sum the financial weight of every word occurring in the segmented XML word, dividing by the total number of segmented words.

$$\text{FinancialRelated} = \frac{\sum \text{FinancialTermWeight}}{\text{TheNumberOfTermsOccurringInTheArticle}}$$

If the FinancialRelated value is higher than a certain threshold, it is concluded that the article is financial related. Otherwise, the article is discarded.

Our system depends to a large extent upon the financial dictionary. There are two main approaches to building dictionaries. One is the knowledge engineering approach; the other is the auto-training approach. In the former, words for the dictionary are constructed by hand. The development process can be very laborious and requires high maintenance costs. For the auto-training approach, someone with sufficient knowledge of the domain annotates a set of training documents. Once a

training corpus has been annotated, a training algorithm is run to train the system for analyzing texts. This approach requires a sufficient volume of training data.

It is very almost impossible to find any free financial dictionary available. Furthermore, the number of Chinese financial term exceeds thousands. Therefore, human involvement will be costly and time-consuming.

We even have no financial news training documents to train our financial term dictionary. Fortunately, the Yahoo stock news available in the Internet is a good training corpus. The idea lies in creating automatic and unsupervised dictionary construction agents which can read Yahoo stock news in the Yahoo web site and count the word frequency occurring in financial news. We assume that the words which occur more in stock news are more related to finance, which will be included in the financial term dictionary. Another advantage of this approach is that it is updating constantly to reflect the current status of language uses in the Internet, especially for continually changing nature of the Internet.

2.4 Semantic Dictionary (Financial Term Trend Dictionary)

The expression of optimism or pessimism w.r.t. the behavior of the stock market relies on a choice of words which is generally understood. However, the words expressing the market sentiment haven't been standardized in a way that science and technology are standardised. Instead, there is a general consensus on how to express optimism or pessimism about an instrument.

Sentiment terms are limited and each term has its own inherent market sentiment. So, it is possible to construct the sentiment dictionary by hand. To our understanding, we have identified 138 Chinese sentiment terms, each of which conveys 'good' or 'bad' market sentiment. We allocated them into 21 groups. Each group has an associated sentiment weight, whose weight is from 10 to -10 consecutively.

$$\text{FinancialTrend} = \frac{\sum \text{MarketSentiment}}{\text{TheNumberOfTermsOccurringInTheArticle}}$$

3 Method Validation

According to the assumption that the words which occur more in stock news are more related to finance, the financial dictionary has been trained starting from the Yahoo stock news web site. It was sufficient to train the system for seven days from (12 AM, 16 July, 2004 to 12 AM, 23 July, 2004) to achieve satisfactory results.

Having put together the three dictionaries, we then run our information retrieval system for another one week, (between 12Am 26 July, 2004 and 12AM, 2 Aug, 2004), obtaining the Market Sentiment diagram of Fig. 5, which includes also the actual values of the shanghai stock exchange composite index.

The most apparent result is that the aggregate information built by our system is in strong agreement with the actual market trends. It may be worth noting that the most evident horizontal bits usually correspond to week ends. On the other hand, in small portions of the diagrams there is discrepancy between the two lines. Clearly the

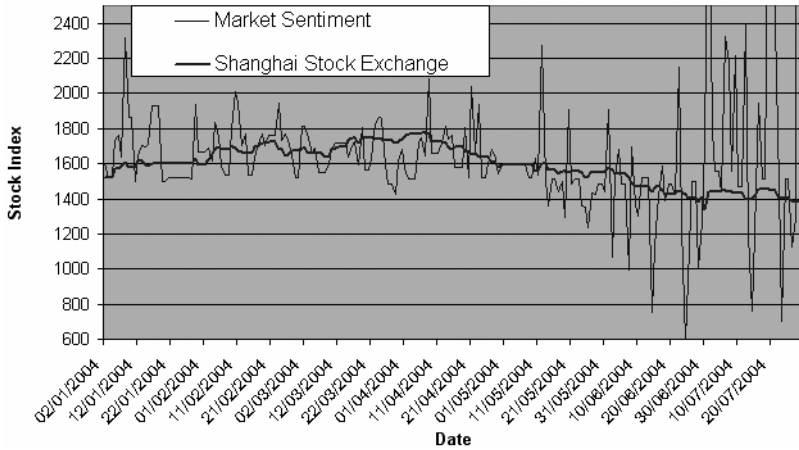


Fig. 4. Comparing Shanghai Stock Exchange Composite and the Market Sentiment

system is not perfect and may not be used to forecast the future but that was not the purpose of our experiments. The strong overall correlation proves, nevertheless, that the system provides a good level of extraction of semantically-rich information.

4 Concluding Remarks

In this paper we have presented a novel approach to extracting semantically-rich information from the Web, illustrating its efficacy by way of a practical case study. To develop our experimental prototype we have used a combination of technologies, ranging from intelligent agents to language segmentation and self-training knowledge systems. We haven't yet optimized the system and there is still space for improvement but the results achieved are extremely encouraging. Our work so far represents a significant first step towards automating semantic data extraction from the web.

References

1. Gaizauskas, R. *et al* (1998), Information Extraction: Beyond Document Retrieval, Computational Linguistics and Chinese Language Processing 3(2):17-60.
2. Cheng, K. S., *et al* (1999), a Study on Word-Based and Integral-Bit Chinese Text Compression Algorithms, J. American Society for Information Science 50(3):218-228
3. Gillam, L., *et al* (2002), Economic News and Stock Market Corr.: A study of the UK Market

An Incremental Updating Method for Clustering-Based High-Dimensional Data Indexing

Ben Wang and John Q. Gan

Department of Computer Science,
University of Essex,
Colchester CO4 3SQ, United Kingdom
{bwangm, jqgan}@essex.ac.uk

Abstract. Content-based information retrieval (CBIR) of multimedia data is an active research topic in intelligent information retrieval field. To support CBIR, high-dimensional data indexing and query is a challenging problem due to the inherent high dimension of multimedia data. Clustering-based indexing structures have been proved to be efficient for high-dimensional data indexing. However, most clustering-based indexing structures are static, in which new data cannot be inserted by just modifying the existing clusters or indexing structures. To resolve this problem, a two-level indexing method, called IASDS plus IPAT method, is developed in this paper. At the IASDS level, clusters and the corresponding subspaces can be incrementally updated, while the indexing structures within the clusters can be incrementally updated at the IPAT level. Furthermore, the proposed IASDS plus IPAT method is able to balance indexing efficiency and query accuracy by choosing an appropriate number of children nodes. The experimental results show that the IASDS plus IPAT method is very efficient for updating clusters and indexing structures with newly inserted data, and that its query accuracy is only slightly degraded while its query time is almost the same in comparison with the similar indexing structure built by non-incremental method.

1 Introduction

There is urgent requirement to support content-based information retrieval (CBIR) of multimedia data from large datasets in an intelligent information retrieval (IR) system. Instead of searching keywords, CBIR searches the content or character of multimedia data, e.g. color, shape, or texture in an image. The feature vectors that characterize multimedia data are often high-dimensional [4][5][7]. As a result, the efficiency of an CBIR system on multimedia data very much depends on high-dimensional data indexing structures. Recently, McNames [13] proposed a principal axis tree (PAT) method that outperforms or matches sixteen leading k -NN algorithms on benchmark data sets and can be used as an efficient indexing structure for k -NN query. The PAT method firstly creates a root node which is then partitioned into children nodes along principal axes generated by singular value decomposition (SVD) [5]. Let n_p be the number of data points in the root node, n_c the number of children nodes from a parent node, and n_y the number of points assigned to a node. The partitioning is repeated for each child node until $n_y < n_c$. However, using the PAT

method on large data sets would be inefficient if the data is high-dimensional. Clustering-based data indexing has been proved to be efficient for indexing high-dimensional data [3][6]. For instance, the high-dimensional data points are firstly clustered and stored in separate files in Clindex [6], and a hash table data structure is then utilized to search the cluster where the query point exists. Nevertheless, clustering high-dimensional data is also a challenging problem [8][9][12]. Subspace clustering is one of the effective solutions to high-dimensional data clustering. Aggarwal et al. [1][2] developed projected clustering algorithms, PROCLUS and ORCLUS, which construct clusters in arbitrarily aligned subspaces with low dimensions, but it is quite hard to define a proper value for the number of clusters and the dimensions of subspaces extended by the clusters in ORCLUS. In our previous work [16], we developed a two-level clustering-based indexing structure that combines ASDS (Automatic Subspace Dimension Selection) and PAT indexing structure, in which a data set is partitioned into clusters by the ASDS algorithm as the first level, and the PAT method is then used to create a tree indexing structure on each cluster at the second level.

Similar to most clustering-based indexing structures [3][6], the ASDS plus PAT indexing structure is static, that is, new object cannot be inserted by just updating the existing clusters and the tree indexing structure. Instead, the whole indexing structure has to be rebuilt. This is inefficient and even impossible for real-time systems, in which time is a crucial element. Various methods for incremental updating of clusters have been developed [10]. Recently, iteration-free approach resolves the cluster updating problem as a constrained optimization problem [17]; subspace tracker approach updates the singular value decomposition (SVD) [3] to track cluster change by inserting objects [11], or SVD can also be updated by approximate algorithms [15]. These methods are too time-consuming to be applied in real-time systems. Besides updating the clusters generated by ASDS, the tree indexing structure on each cluster created by PAT should also be updated. This paper proposes IASDS plus IPAT, which is able to incrementally update the existing clusters and inner-indexing structures very efficiently without sacrificing the query accuracy considerably.

In section 2 and section 3, the methods for incrementally updating of IASDS and IPAT are proposed respectively. Experimental results on five data sets are analyzed in section 4, and a face recognition demo system is also presented at the end of section 4. Finally, a conclusion is drawn in section 5.

2 Incremental Updating of IASDS Clusters and Subspaces

Let C_i represent the i -th cluster, s_i be its centroid. Subspaces of C_i are extended by \mathcal{E}_i , where \mathcal{E}_i represents a set of coordinate eigenvectors $\{e_{ij}\}$ in the subspace of cluster C_i . The \mathcal{E}_i is by determined by ranking the average projected distances $ArePrDist_{ij}$ on all the eigenvector directions and choosing $\{e_{ij}\}$ with large $ArePrDist_{ij}$, which is defined by formula (1). Here, the “.” represents dot product operator, and the “|” represents absolute value operator. $NumPInC_i$ is the number of data points in C_i , p_{ik} is the k -th data point in C_i , M is the number of clusters, and L is the total number of eigenvectors in clusters C_i .

$$AvePrDist_{ij} = \sum_{k=0}^{NumPInC_i} |p_{ik} \cdot e_{ij} - s_i \cdot e_{ij}| / NumPInC_i, \quad (i \in [1, M], j \in [1, L], k \in [1, NumPInC_i]) \quad (1)$$

For a new data point p , the cluster to insert is chosen and updated as follows. Firstly, the method calculates the projected distances $P' dist_i(p, s_i, \mathcal{E}_i, l_{min})$ between this new data point and the centroid of every cluster by formula (2). The l_{min} is defined as the minimum dimension among all the subspaces \mathcal{E}_i . Secondly, assign the new data p to cluster C_l with minimal projected distances, where l is determined by formula (3). Thirdly, it updates incrementally the parameters related to cluster C_l , including the average projected distances $AvePrDist_{ij}$, the centroid s_l , and the subspace \mathcal{E}_l . The $AvePrDist_{ij}$ are updated by formula (4), and the s_l is updated by formula (5). The symbol “ \leftarrow ” represents an assignment operator, and “ $*$ ” represents vector cross product operator. The subspace \mathcal{E}_l is updated by choosing e_{ij} according to the ranking of $AvePrDist_{ij}$. Finally, it increases $NumPInC_i$ by 1 to stop the cluster updating process. During the incremental updating process, $\{e_{ij}\}$ inherits from the static ASDS algorithm with an assumption that ASDS has already found most outstanding eigenvectors $\{e_{ij}\}$ as the coordinate candidates for extending subspaces. The aim of IASDS method is to choose appropriate eigenvectors to reorganize the subspaces when a new point is inserted. The assumption is reasonable in practice because the static ASDS generates eigenvectors based on a large amount of data points and it extracts almost all the outstanding eigenvectors.

$$P' dist_i(p, s_i, \mathcal{E}_i, l_{min}) = \sum^{l_{min}} (p \cdot e_{ij} - s_i \cdot e_{ij})^2 \quad (2)$$

$$l = \arg \min_{0 \leq i \leq M} \{ P' dist_i(p, s_i, \mathcal{E}_i, l_{min}) \} \quad (3)$$

$$AvePrDist_{ij} \leftarrow (AvePrDist_{ij} \cdot NumPInC_i) + (|p \cdot e_{ij} - s_i \cdot e_{ij}| / (NumPInC_i + 1)) \quad (4)$$

$$s_l \leftarrow (s_l * NumPInC_l + p) / (NumPInC_l + 1) \quad (5)$$

3 Incremental Updating of IPAT Indexing Structure

Corresponding to each cluster, there is an IPAT indexing structure. If a new data point is inserted into a cluster, the corresponding IPAT has to be updated. The first step is to decide which node in the IPAT the new data point should be inserted into. Then the parameters that characterize the chosen node should be updated in an incremental manner, which include *flag*, partition direction, minimum and maximum projected distances on the partition direction g_{min} and g_{max} . The partition direction of a node is defined as a chosen direction on which all of data points are projected and partitioned into children nodes.

3.1 Choosing a Node to Insert a New Data Point

Let n_y be the number of data points in a node, and n_c be the number of its children nodes. If n_y is less than n_c , there will be no children nodes for this node and it

becomes a terminal node. Otherwise, it is an internal node. The number of data points in each child node equals $\lfloor n_y / n_c \rfloor$ or $\lfloor n_y / n_c \rfloor + 1$. A *flag* parameter is used to mark whether a child node is full or not. The definition of *flag* is as follows:

$$flag = \begin{cases} false & NP = \lfloor n_y / n_c \rfloor \\ true & NP = \lfloor n_y / n_c \rfloor + 1 \end{cases} \quad (6)$$

where *NP* is the number of data points in the child node.

A new data point is inserted into a certain node as follows. The root node of an IPAT tree is denoted as *rootNode*. Let *cNode_i* be the *i*-th child node of the *rootNode*. At the beginning, a new point \bar{p} is inserted into the *rootNode*. If the *rootNode* is an internal node, \bar{p} is projected on the partition direction of the *rootNode*. If the projection is between g_{min} and g_{max} of *cNode_i*, insert this new data point into *cNode_i* at the second level of the IPAT. Otherwise, if the projection is less than g_{min} of the first child node or greater than g_{max} of the last child node, the new data point is inserted into the first or the last child node. The process is repeated until \bar{p} is inserted into a terminal node in the last level of the IPAT.

3.2 Incremental Updating of the IPAT Node Structure

The node structure should be updated, too. The updating of *flag*, g_{min} and g_{max} is straightforward according to their definitions. This section focuses on how to update the partition direction. Based on the FastMap idea in [7], a new method is suggested in this paper to initialize and update the partition direction. The partition direction is initialized as the direction of the beeline between two points whose projections on the principal axis produce the maximum distance among other points in the node. Let \bar{d}_1 and \bar{d}_2 represent these two points, then the partition direction is $\overline{d_1 d_2} = \bar{d}_1 - \bar{d}_2$. Once \bar{d}_1 and \bar{d}_2 are initialized, $\overline{d_1 d_2}$ replaces the principal axis as the partition direction for the subsequent incremental updating steps. It means that the principal axis is only required at the initialization step, but not in the incremental updating steps. Therefore it is not necessary to be updated. Instead, the partition direction is incrementally updated and approximately plays the role of the principal axis in the IPAT indexing. When a new data point \bar{p} is inserted into a node, this paper proposes the following steps to incrementally update \bar{d}_1 and \bar{d}_2 , and thus the partition direction $\overline{d_1 d_2}$:

- Step 1. Step 1. Calculate distances $DIST_1$ between \bar{p} and \bar{d}_1 , $DIST_2$ between \bar{p} and \bar{d}_2 , and $DIST$ between \bar{d}_1 and \bar{d}_2 .
- Step 2. Step 2. If $DIST_1 > DIST$, set $\bar{d}_2 = \bar{p}$ and keep \bar{d}_1 unchanged.
 Otherwise, if $DIST_2 > DIST$, set $\bar{d}_1 = \bar{p}$ and keep \bar{d}_2 unchanged.

To summarize, the incremental updating is carried out by inserting new data points one by one into the static ASDS plus PAT indexing structure. For each new data insertion, a cluster is chosen and the corresponding subspace is incrementally updated by IASDS as indicated in section 2, and then the IPAT indexing structure corresponding to the chosen cluster is incrementally updated as described in section 3.

4 Experimental Results and Analysis

Experiments are carried out on five high-dimensional datasets: Census, Coral, FaceR, Forest, and Synthetic, in order to evaluate the performance of the proposed method [16]. The Census data set contains 22784 139-dimensional feature vectors. The Coral dataset contains 68040 32-dimensional image feature vectors. The FaceR dataset contains 2000 99-dimensional feature vectors extracted from face images [14]. The Forest dataset contains 41012 54-dimensional feature vectors. Finally, the Synthetic data set, generated by Aggarwal [1], contains 12,040 40-dimensional feature vectors. In our experiments, the proposed incremental updating method is compared with the static ASDS plus PAT method [16] in terms of indexing efficiency and query accuracy on the five data sets. As the comparison of the static ASDS plus PAT method with other clustering-based methods was discussed in detail in the previous work [16], only incremental updating issues are investigated in this paper. For creating the static ASDS plus PAT indexing structure, denoted as Static PAT, all the objects in a data set are clustered by the ASDS algorithm first, and then a PAT tree is built on each cluster [16]. For creating the incrementally updated indexing structure, 50% data objects in a data set are used to build up a static indexing structure first by the ASDS plus PAT method, which is called Half Static PAT, and the remaining 50% data objects are then inserted into the existing clusters and indexing trees by the IASDS plus IPAT method, which is called IPAT for short. In this paper, the indexing efficiency is measured by the indexing time, and the k -NN query time is considered as a measure of efficiency. The query accuracy is measured by a query accuracy ratio, which is defined as the ratio of the number of correctly returned query results to the total number of query results. The number of children nodes is an important parameter that influences the indexing efficiency and query accuracy, so different numbers of children nodes were tested in the experiments.

The indexing times used in building the Static PAT, Half Static PAT, and IPAT structures with different numbers of children nodes on the five data sets are illustrated in Fig. 1, Fig. 2, Fig. 3, Fig. 4 and Fig. 5 respectively. Noticeably, the indexing time for creating the IPAT structure is much less than that of the Static PAT structure. This is because the indexing time of the incremental updating method is linear to the number of inserted data points, whilst in creating a Static PAT structure the partition direction of each node is determined by the principal axis, i.e., the eigenvector with the maximum eigenvalue, the time complexity of which is $O(N^2)$, where N equals the total number of data points in the node. The query times on the five data sets are shown in Fig. 6, Fig. 7, Fig. 8, Fig. 9 and Fig. 10, from which we can see that the Static PAT structure and the IPAT structure have very limited difference in terms of the query time. The query accuracy ratios on the five data sets with different numbers of children nodes are displayed in Fig. 11, which shows that the query accuracy ratios on all the data sets decrease as the number of children nodes increases. However, in Fig. 1, Fig. 2, Fig. 3, Fig. 4 and Fig. 5, the indexing time decreases in general as the number of children nodes increases. Therefore, in order to balance the indexing time and the query accuracy of the incremental updating method, a compromised number

of children nodes should be chosen. For example, using 30 children nodes on Census, Coral, FaceR, and Synthetic data sets gets very satisfactory query accuracy (Query Accuracy Ratios are over 98%) and almost optimal indexing time. On Forest data set, incremental updating is more sensitive because of 44 Boolean attributes among 54 dimensions in the data set. Choosing 10 children nodes on Forest data set is a good choice. To show a CBIR application, a face recognition demo system in Fig. 12 has been developed to implement the IASDS plus IPAT indexing structure on FaceR data set. Twenty nearest neighbours of the selected query image are returned at the right-bottom corner. With the IASDS plus IPAT method, the static indexing structure can be incrementally and efficiently updated.

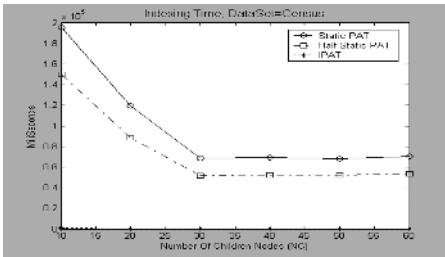


Fig. 1. Indexing time on Census

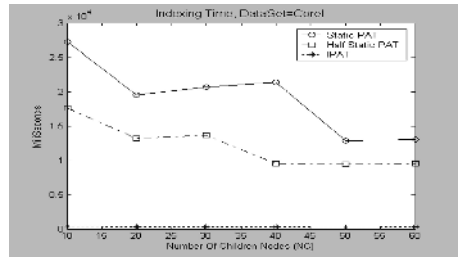


Fig. 2. Indexing time on Coral

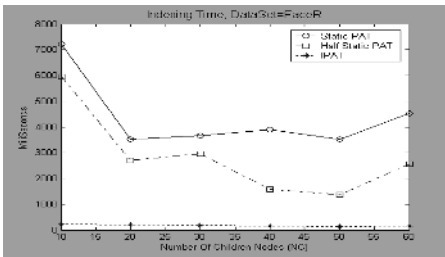


Fig. 3. Indexing time on FaceR

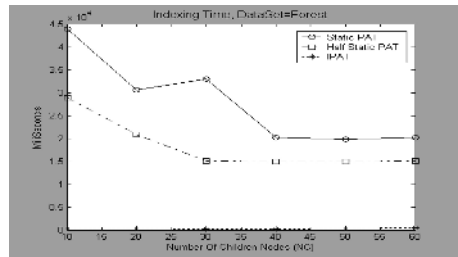


Fig. 4. Indexing time on Forest

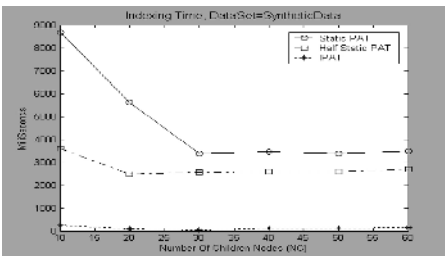


Fig. 5. Indexing time on Synthetic

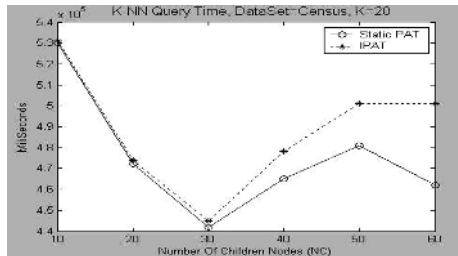


Fig. 6. The k-NN query time on Census

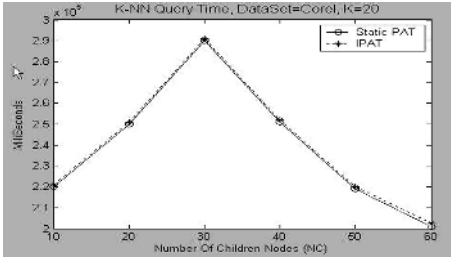


Fig. 7. The k -NN query time on Coral

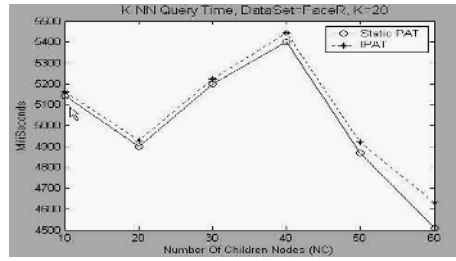


Fig. 8. The k -NN query time on FaceR

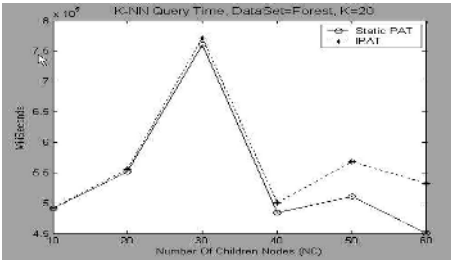


Fig. 9. The k -NN query time on Forest

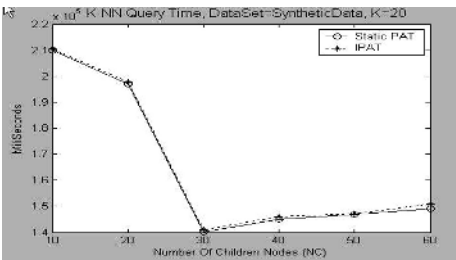


Fig. 10. The k -NN query time on Synthetic

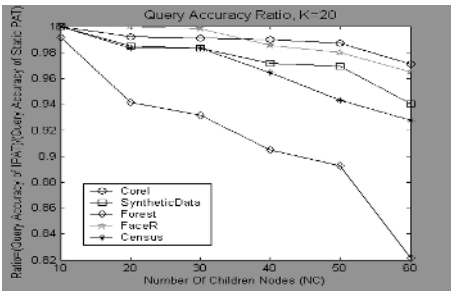


Fig. 11. Query accuracy ratios on five data set

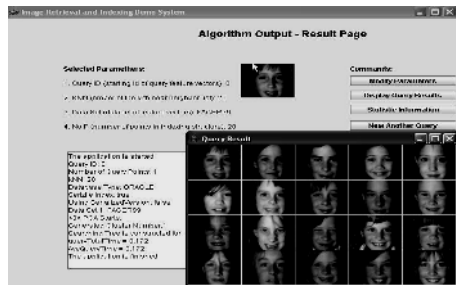


Fig. 12. The face recognition demo system

5 Conclusion

To efficiently support CBIR, this paper proposes an incremental updating method called IASDS plus IPAT for clustering based high-dimensional data indexing. Experimental results have shown that without considerable sacrifice of its query accuracy the IASDS plus IPAT method is able to incrementally update existing clusters and inner-indexing structures when there is a need of inserting new data. In experiments, it is also possible to choose an optimal number of children nodes to increase the query accuracy with little sacrifice of the indexing efficiency. Currently, a face recognition demo system has implemented the IASDS plus IPAT indexing structure. Reasonably, the IASDS plus IPAT method is especially suitable for real-time intelligent CBIR systems on multimedia data.

References

1. Aggarwal, C.C., Procopiuc, C., Wolf, J.L., Yu, P.S., Park, J.S.: Fast algorithms for projected clustering. Proc. of the ACM SIGMOD Conf., Philadelphia, PA (1999) 61-72
2. Aggarwal, C.C., Yu P.S.: Finding generalized projected clusters in high dimensional spaces, Sigmod Record, Vol. 29, (2000) 70-92
3. Castelli, V., Thomasian, A., Li, C.S.: CSVD: Clustering and singular value decomposition for approximate similarity searches in high dimensional space. IEEE Trans. on Knowledge and Data Engineering, Vol. 15, (2003) 671-685
4. Chakrabarti, K. Mehrotra, S.: The hybrid tree: An index structure for high dimensional feature spaces. IEEE Conf. on Data Engineering, Sydney, Australia (1999) 440-447
5. Chandrasekaran, S., Manjunath, B.S., Wang, F.Y., Winkeler, J., Zhang, H.: An eigenspace update algorithm for image analysis. Graphical Models and Image Processing, Vol. 59, (1997) 321-332
6. Li, C., Chang, E., Garcia-Molina, H., Wang, J., Wiederhold, G. Clindex: Clustering for similarity queries in high-dimensional spaces. IEEE Trans. on Knowledge and Engineering, Vol. 14, (2002) 792-808
7. Faloutsos, C., Lin. K.I.: FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia databases. ACM SIGMOD, (1995) 163-174
8. Grabmeier, J., Rudolph, A.: Techniques of cluster algorithms in data mining, Data Mining and Knowledge Discovery, Vol. 6, (2002) 303-360
9. Jain, A., Dubes, R.: Algorithms for Clustering Data. Prentice Hall (1998)
10. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: A review. ACM Computing Surveys, Vol. 31, (1999) 264-323
11. Kavcic, A. and Yang, B.: Subspace tracking with adaptive threshold rank estimation. VLSI Signal Processing, Vol. 14, (1996) 75-91
12. Li, C., Chang, E., Molina, H.G., Wiederhold, G.: Clustering for approximate similarity search in high-dimensional spaces. IEEE Trans. on Knowledge and Data Engineering, Vol. 14, (2002) 792-808
13. McNames, J.: A fast nearest neighbor algorithm based on a principal axis search tree. IEEE Trans. on Pattern Analysis and Intelligence, Vol. 23, (2001) 964-976
14. Pentland, A., Picard, R.W., Sclaro, S.: Photobook: Content-based manipulation of image databases. Int. Journal of Computer Vision Archive, Vol. 18, (1996) 233-254
15. Strobach, P.: Low rank adaptive filtering. IEEE Trans. Signal Processing, Vol. 44, (1996) 2932-2947
16. Wang B. and Gan, J.Q.: Integration of projected clusters and principal axis trees for high-dimensional data indexing and query, IEEE Conf. on Intelligent Data Engineering and Automated Learning, Exeter, UK (2004) 191-196
17. Yeh, C.H. and Kuo, C.J.: Iteration-free clustering algorithm for nonstationary image database, IEEE Trans. on Multimedia, Vol. 5, (2003) 223-236

Typhoon Track Prediction by a Support Vector Machine Using Data Reduction Methods

Hee-Jun Song¹, Sung-Hoe Huh^{2,*}, Joo-Hong Kim^{3,**},
Chang-Hoi Ho³, and Seon-Ki, Park²

¹ Department of Electrical Engineering, Korea University, Seoul, Korea
nyaong7@korea.ac.kr

² Dept. of Environmental Science & Engineering, Ewha Woman's University, Seoul, Korea
sungh@korea.ac.kr, spark@ewha.ac.kr

³ School of Earth & Environmental Sciences, Seoul National University, Seoul, Korea
{jhkim, hoch}@cpl.snu.ac.kr

Abstract. Typhoon track prediction has mostly been achieved using numerical models which include a high degree of nonlinearity in the computer program. These numerical methods are not perfect and sometimes the forecasted tracks are far from those observed. Many statistical approaches have been utilized to compensate for these shortcomings in numerical modeling. In the present study, a support vector machine, which is well known to be a powerful artificial intelligent algorithm highly available for modeling nonlinear systems, is applied to predict typhoon tracks. In addition, a couple of input dimension reduction methods are also used to enhance the accuracy of the prediction system by eliminating irrelevant features from the input and to improve computational performance.

1 Introduction

Some tropical depressions develop to typhoons when their sustained wind speed is greater than 17 m s^{-1} and they cause the most costly and deadly type of natural disaster over East Asian countries. Therefore, it is a critical matter to attain the closest prediction as to which direction they will move so as to reduce the scope of possible damage. There have been a number of efforts to improve prediction skills for typhoon tracks. Numerical methods which consider the dynamics of typhoon motion have been primarily in use, but many statistical methods have been developed as well not only to correct them but to predict in their own way since the numerical models suffer from a systematic error. For example, Aberson updated the climatology and persistence (CLIPER) track forecast model, which calculates a simple regression based upon climatology and persistence [1]. Elsberry et al. developed a statistical post-processing technique to reduce U.S. Navy global model (NOGAPS) track forecast errors [2]. Sohn et al. solved the problem with a state-dependent model [3]. Bessafi et al. attempted to improve the CLIPER model by combining it with the analog concept

* Co-first author.

** Corresponding author.

[4]. Sohn et al. developed a dynamic linear model to remove the systematic error of the operational barotropic adaptive-grid typhoon simulation (BATS) forecasts [5]. Those models are effective in tracking typhoons but they still have problems with accuracy. Therefore, another model which is able to resolve the problem ought to be developed because typhoon track forecasts should be done quickly and with greater accuracy in the operational forecast.

To effectively resolve these problems, a method of typhoon track prediction using a Support Vector Machine (SVM) with data dimensionality reduction is proposed in this study. There has been a small amount of research applying artificial intelligence algorithms such as Neural Networks (NNs) to typhoon track prediction. For example, Baik and Paek used a back-propagation NN in typhoon track and intensity prediction [6]. NNs have showed slightly poorer performance compared with the conventional multiple linear regression models, whence it is believed that the track prediction performance of neural network models would be similar to that of multiple linear regression models if a well-designed NN is used. Thus, further development may support conventional numerical methods in the future. However, there are still many problems unsolved. These include the facts that (1) NNs are relatively time-consuming algorithms due to their learning procedure which considers every single data sample in each iteration. SVM is known to be a much faster algorithm than NNs. (2) Considering the architecture of NNs, their performance in computation is greatly affected by the dimension of the input data. In the present study, we set the input dataset to have up to 200 dimensions and it makes NNs' learning procedure too time-consuming. (3) Even though NNs' tracking system is constructed successfully, there is always a risk of local minima from the nature of NNs so it is not guaranteed that NNs provide the optimal answer to the problem.

Those possibly occurring problems of NNs are effectively resolved by using SVM. Firstly, SVM forms a feature space which is composed of the most effective samples. It doesn't take account of data samples one by one like NNs do. Hence, it shows much faster performance than NNs in the learning period. Secondly, SVM is known to be a much more robust algorithm than NNs with high-dimensional data. Moreover, SVM is a mathematically sound algorithm so that it is easily explained how it works and it is originally designed not to result in local minima.

In spite of those benefits of using SVM, the super-high dimensionality is still a problem. It may increase the nonlinearity and complexity of the typhoon track problem so that it probably leads to unexpected or inappropriate results. In addition, there might be possibilities that the high-dimensional input data contains some non-relevant or redundant components. From this point of view, several dimensionality reduction methods for the typhoon input data were conducted in advance of the SVM learning procedure. The advantages possibly obtained by dimensionality reduction methods are: (1) those methods have the ability to increase the linearity of the dataset by extracting the most affective components to the output. So the system can be said to be better tuned to the problem. (2) Since the results of those algorithms are reduced in dimension, the SVM learning system can be expected to possess higher performance in computation.

The remainder of this paper is organized as follows. The typhoon dataset is presented in Section 2. Section 3 introduces the dimensionality reduction methods. A brief description of the SVM system constructed to model the typhoon track is pro-

posed in Section 4 and the results of experiments are shown. Section 5 offers conclusions of the research.

2 Typhoon Dataset

The typhoon dataset including 6-hourly positions is provided by the RSMC (Regional/Specialized Meteorological Centers) Tokyo Typhoon Center. It consists of 1054 typhoon tracks recorded from 1951 to 2002 formed over the western North Pacific. Among them, 587 typhoon tracks which lasted longer than the average life-time of the typhoon have been considered and clustered by using Fuzzy C-Means Clustering algorithm, resulting in 11 clusters as shown in Fig. 1 [7].

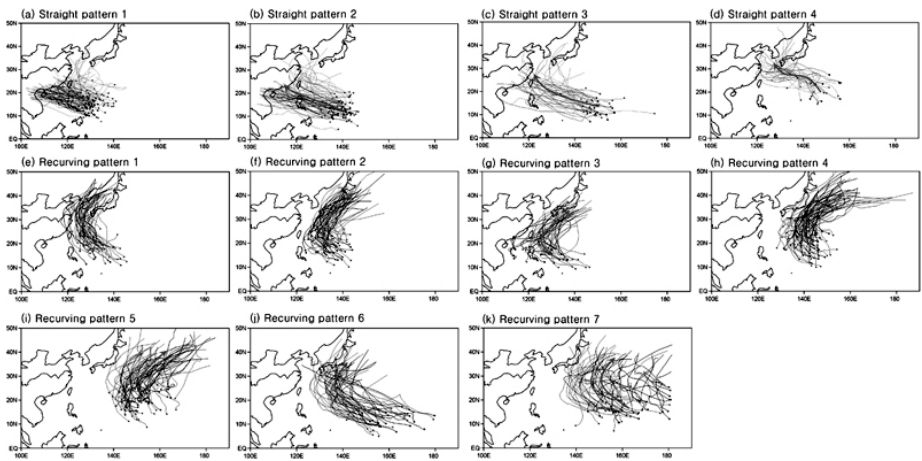


Fig. 1. Clustered 11 Patterns of Typhoon Tracks

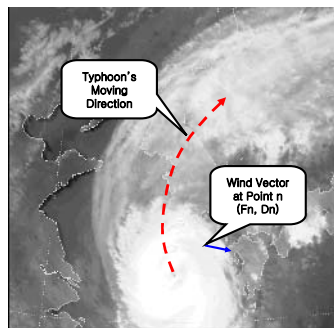


Fig. 2. Example of Typhoon Track and Wind Vector

Each cluster represents the moving trends of typhoons. Among those 11 clusters, Cluster S1 was firstly selected to model the moving route in more detail for the research. Others remain to be modeled in further works. Cluster S1 has 87 typhoon tracks and is made of a number of time series of data measured every 6hr as a time

point containing 100 wind vectors at each time. Each wind vector indicates each wind force and wind direction at the point n denoted as (Fn, Dn) in Fig. 2.

In Cluster S1, there are 1703 times series data which has 200 elements representing wind forces and directions at 100 points in total. To standardize the research this dataset is divided into 2 parts by approximately 50:50: training dataset (862 samples) and test dataset (841 samples).

3 Dimensionality Reduction Methods

A couple of methods for dimensionality reduction have been applied to the typhoon dataset. They are Principal Component Analysis (PCA), Wavelet Decomposition and 2 statistical methods. Brief descriptions are described in the following sections.

3.1 Principal Component Analysis

PCA is known as a useful technique to reduce the dimensionality of large data sets and can also be used to find signals in noisy data. In some cases, the dimension of the input is too large, but the components in the input are highly correlated (redundant), PCA is useful to reduce the dimension of the input. PCA has three representative effects: (1) it orthogonalizes the components of the input vectors so that they are uncorrelated with each other, (2) it orders the resulting orthogonal components (principal components) so that those with the largest variation come first, and (3) it eliminates those components that contribute the least to the variation in the data set. Since the results derived from PCA are orthogonal to each other, there is much less redundancies in the resulting data [8].

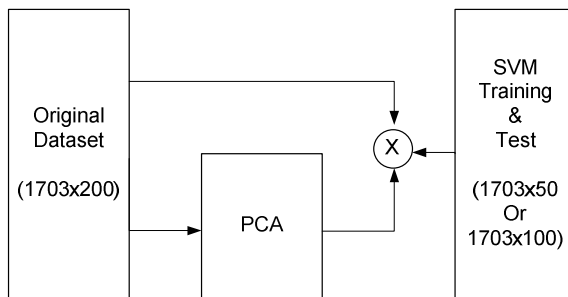


Fig. 3. Principal Component Analysis for Dimensionality Reduction

In this research, two input dataset having different dimension, 50 and 100 were generated to be modeled by SVM. The whole input data (1703 x 200) was transformed by PCA first. Then the generated array having the trends of the original data (200 x 50 or 200 x 100) was multiplied to the original data array (1703 x 50 or 1703 x 100). Hence, the finally resulting arrays contain 50 or 100 dimensional data containing the principal information of the original input data. These arrays were split into training and test dataset and used to train and test using SVM.

3.2 Wavelet Decomposition

A 1-Dimensional Discrete Wavelet Transformation (DWT) was used to reduce the dimensionality of the input typhoon data. From the theory of Wavelet, a signal can be divided into its approximation components and detail components. Those analyzed components then could be separately processed. For instance, each signal components analyzed from a signal are able to pass each different filter then composed into one signal without losing the original information. In addition, an original signal might be able to be completely reconstructed without all the analyzed components. Because detail signals in higher level might be redundancies [9]. This characteristic of the wavelet enables the dimensionality reduction of the typhoon dataset.

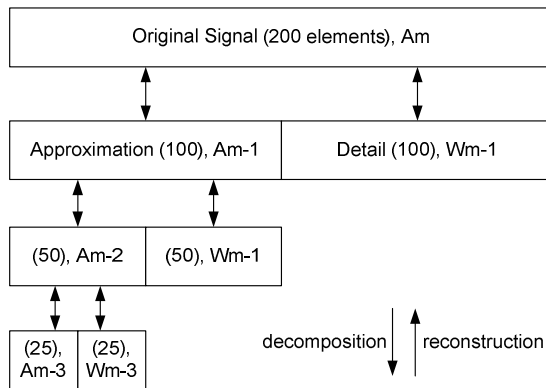


Fig. 4. Wavelet Decomposition (Analysis) and Reconstruction (Synthesis)

In the case of this research, a time point sample containing 200 elements (200 dimensions) is decomposed by DWT into approximation components containing 100 coefficients and detail components containing 100 coefficients. Then the approximation component could be analyzed more to the further steps. As shown in the Fig. 4, the original 200 coefficients were decomposed down to 25 approximation coefficients and 25, 50 and 100 detail coefficients. In this process, lower level coefficients are considered as more important components since they have more information of the original signal contained in lower frequency region. In this research, the case of using the lowest level of 50 coefficients and the case of using the lowest 100 coefficients were conducted. Those are regarded as the principal components in the dataset and applied to the SVM system as input dataset.

3.3 Statistical Methods

Since those methods described in Section 3.1 and 3.2 are effective ways to reduce the redundancies in high dimensional data, it is impossible to figure out which components are more or less affective intuitively because the reduced datasets are generated on transformed spaces. In this research, two kinds of statistical methods were applied to directly find out the most affective components without a data conversion to another space.

- Cross-Correlation (CC): Cross correlation of each component through whole data with the reference data (longitude and latitude values) were calculated. CC values near 1/-1 indicate strong positive / negative linear relationship between 2 data and CC values near 0 indicates no correlation between them. Once CCs are calculated, they are ranked in decreasing orders and the first 50 / 100 components are selected as the principal components for the input data of SVM.
- Regression Analysis (RA): A Linear or Polynomial Regression Analysis is a statistical analysis method to describe or predict a dependent variable with several independent variables. This method can be a measure for evaluating how much statistically appropriate a prediction is, by finding out the degree of relationship between a dependent and a number of independent variables. As a result, RA provides the order of dependent variables from the most principal one.

4 Support Vector Machine (SVM)

A Support Vector Machine (SVM) is an artificially intelligent algorithm well known for its high performance and fast operational speed in image processing and other fields. Fig. 5 shows a brief description of SVM operation. Data samples are located in a space and the support vectors (the most influential data samples) which can classify the dataset the best are selected and a hyper-plane such as is depicted by a hard line in Fig.5 is constructed in the space by minimizing the margin denoted by w . SVM has several advantages compared to neural networks. First, SVM doesn't have local minima. Secondly, SVM doesn't need to consider whole samples in training and test so it shows much faster performance than neural networks in real applications. And since it is possible to construct a hyper-plane having a high dimension, it is extraordinarily robust on high-dimensional problems compared with other AI algorithms [11].

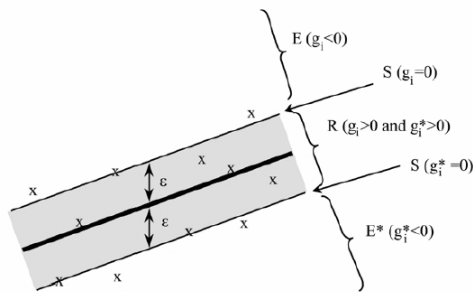


Fig. 5. Support Vector Machine Regression

In this research, an SVM regression model was constructed to track the typhoon track. It has a longitude and a latitude point data as inputs at time point t , and generates the prediction of the typhoon's position at the next time point, $t+1$. Every possible type of kernels and parameters for SVM has been tried and the only best experiment results are presented in Section 5.

5 Experiments

A number of input datasets generated by the dimensionality reduction methods described in Section 2 were applied to the SVM typhoon track estimator. Experimental results are evaluated by comparison between the SVM estimated tracks and the actual typhoon track. The RMSEs (root mean squared errors) between them were presented for numerical evaluation. The results of evaluation of each predictor are presented in the Figures and table below. Only test results are shown because only they are meaningful in real applications.

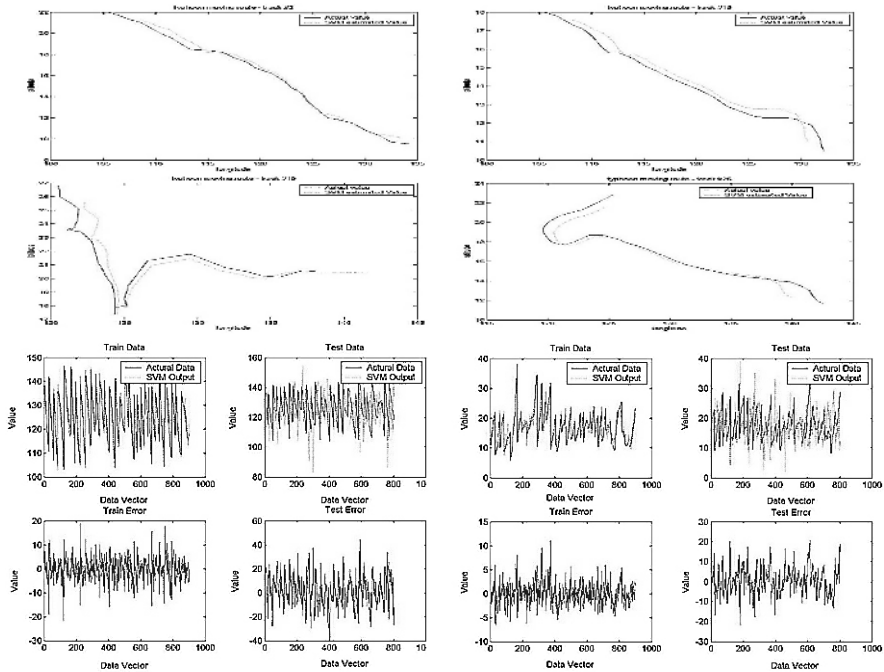


Fig. 6. (a) top left (b) top right (c) middle left (d) middle right: Actual Typhoon Track and SVM Estimated Track of track no. 2, 15, 19 and 25, respectively (e) bottom left: Latitude MSE (f) bottom right: Longitude MSE

Figs. 6a-6d represent the generated typhoon track and the actual track by SVM without any input reduction methods. Fig. 6e and 6f show the latitude and longitude error at each time point. It shows little worse performance than the cases of dimensionality reductions. The high degree of non-linearity in the whole dataset probably causes the decrease of performance. Since the track figures do not conveniently enable recognition of the performance at a glance, only the figures of latitude and longitude errors are given from now on. The results from the PCA and Wavelet Decomposition dimensionality reduction methods in the case with 100 components show reasonable performance with a lot faster operation time than the case without using a dimensionality reduction method. Table 1 shows the comparison of computational time in each case. It indicates the use of dimensionality reduction methods is appro-

Table 1. Comparison of Computational Performance (with Matlab 6.5TM, 2.8 Ghz, 512 Mb memory system)

Simulation Methods	Number of Components	Time (min)
SVM	200	62.38
SVM + PCA	100	32.56
	50	24.28
SVM + DWT	100	33.06
	50	33.10
SVM + CC	100	24.32
SVM + RA	100	33.01

Table 2. Comparison of Accuracy Performance

Simulation Methods	Number of Components	MSE (longitude)	MSE (latitude)
SVM	200	7.4563	15.4554
SVM + PCA	100	5.3408	12.3989
	50	7.2134	16.7678
SVM + DWT	100	6.0234	11.9237
	50	7.8545	16.5345
SVM + CC	100	10.5425	21.5466
SVM + RA	100	8.0245	13.4356

appropriate to model the typhoon track. The results with 50 components generated by PCA and the Wavelet Decomposition method show little worse performance than the case of 100 components. It is probably caused because the more reduced dataset doesn't have enough information to model the problem. In the results driven by statistical analysis methods, the components selected by cross-correlation generate a less appropriate result. This probably is due to the fact that the selected components do not properly represent the whole dataset. On the contrary, the RA generating components derive a better result than the case of CC even though it still has a large amount of RMSE error. These results from statistical methods could give rise to the conclusion that they are not so great data-analyzing methods for problems containing relatively high nonlinearity compared to usual problems. Table 2 shows the RMSE errors for each experiment.

6 Conclusion and Further Research

The typhoon track prediction is a critical problem and it is difficult to be modeled because it has an extremely high nonlinearity. In this research, a method of using SVM with dimensionality reduction methods is proposed. Those proposed dimensionality reduction methods are said to be appropriate from the viewpoint of evaluation of the results and they are concluded to possess abilities in identifying nonlinear problems, provide possibilities of higher computational performance and improve accuracy.

There still are a number of areas to be pursued. One of remaining research ideas is to conduct more experiments with various other parameters of SVM. In this research, the SVM system cannot be said to be perfectly optimized because all possible parameter set were not considered. The SVM optimization process usually depends only on intuition of programmers so that there are a lot more possible cases which might generate better results than shown in this paper.

More experiments with more input factors beside wind vector data must be considered to enhance the accuracy of the typhoon track model. There are possibilities of existence of other factors such as latitude and longitude of typhoon center, typhoon motion vector and intensity of typhoon, etc.

Experiments about typhoon intensity prediction at each point are needed alongside the track prediction. However, in spite of all deficiencies of the experiment conducted, they are still meaningful because the proposed method is more effective than previous researches and those methods might be available for many other problems which possess high nonlinearity.

Acknowledgment

This study was performed for the Research and Development programs on Meteorology and Seismology funded by the Korean Meteorological Administration.

References

1. Aberson, S. D.: The Prediction of the Performance of a Nested Barotropic Hurricane Track Forecast Model. *Weather and Forecasting*, Vol. 12, (1997) 24-30
2. Elsberry, R. L., M. A. Boothe, G. A. Ulses, P. A. Harr: Statistical Postprocessing of NOGAPS Tropical Cyclone Track Forecasts, Vol. 127, (1999) 1912-1919
3. Sohn, K.-T., J.-S. Baik, Y.-S. Kim: Estimation of Typhoon Track using Bivariate State Dependent Model. *Korean Journal of Atmospheric Sciences*, Vol. 35, No. 4, (1999) 613-618
4. Bessafi, M., A. Lasserre-Bigorry, C. J. Neumann, F. Pignolet-Tardan, D. Payet, M. Lee-Ching-Ken: Statistical Prediction of Tropical Cyclone Motion: An Analog-CLIPER Approach. *Weather and Forecasting*, Vol. 17, (2002) 821-831
5. Sohn, K.-T., H. J. Kwon, A.-S. Suh: Prediction of Typhoon Tracks Using Dynamic Linear Models. *Advances in Atmospheric Science*, Vol. 20, No. 3, (2003) 379-384
6. Baik, J.-J., J.-S. Paek: Performance Test of Back-propagation Neural Network in Typhoon Track and Intensity Prediction. *Korean Journal of Atmospheric Sciences*, Vol. 3, No. 1, (2000) 33-38
7. Kim, J. H.: A Study on the Seasonal Typhoon Activity using the Statistical Analysis and Dynamic Modeling. Ph.D Thesis, Seoul National University, (2005)
8. Jolliffe, I.T.: *Principal Component Analysis*, Springer-Verlag, New-York, (1986)
9. Goswami J. C., A. K. Chan: *Fundamentals of Wavelets*. Willey-Interscience Publication, New-Work, (1999)

Forecasting Tourism Demand Using a Multifactor Support Vector Machine Model*

Ping-Feng Pai^{1, **}, Wei-Chiang Hong², and Chih-Sheng Lin³

¹ Department of Information Management, National Chi Nan University,
1 University Rd., Puli, Nantou 545, Taiwan, China
Tel.: +886-4-92-91-0960, ext:4871
paipf@nunc.edu.tw

² School of Management, Da-Yeh University,
112 Shan-Jiau Rd., Da-Tusen, Changhua, 51505, Taiwan, China
d9230006@mail.dyu.edu.tw

³ Department of Industrial Engineering and Enterprise Information,
Tung-Hai University, Box 985, Taichung 40704, Taiwan, China
r9102006@pchome.com.tw

Abstract. Support vector machines (SVMs) have been successfully applied to solve nonlinear regression and times series problems. However, the application of SVMs for tourist forecasting has not been widely explored. Furthermore, most SVM models are applied for solving univariate forecasting problems. Therefore, this investigation examines the feasibility of SVMs with backpropagation neural networks in forecasting tourism demand influenced by different factors. A numerical example from an existing study is used to demonstrate the performance of tourist forecasting. Experimental results indicate that the proposed model outperforms other approaches for forecasting tourism demand.

Keywords: Forecasting, multifactor support vector machines, backpropagation neural networks, tourism demand.

1 Introduction

More accurately forecasting tourism demand would facilitate for assisting managerial, operational and tactical decision making of both the private and the public sectors [1]. Numerous studies have attempted to identify the most effective model for forecasting tourism demand using different mathematical methods. The tourism demand forecasting models are divided primarily into two categories, regression models and time series models [2]. The main distinction between these two types of models is that the latter attempts to construct an equation linking tourism demand and past data. On the other hand, the former tries to find the relationships between the dependent variable and other socioeconomic factors, including income, traveling costs, and exchange rate and daily expenses for the destination [3,4]. However, the explanatory ability of the regression model can be increased by scrapping some variables. Accordingly, more attention could be paid to collecting data related to these variables [5]. Lim [6] hypothesized that

* This research was conducted with the support of National Science Council NSC 94-2213-E-260-023.

** Corresponding author.

the independent variables in a regression model can be mainly included to avoid the aforementioned problems. Lim and McAleer [7] developed a series of various autoregressive integrated moving average (ARIMA) models to forecast tourist arrivals from Malaysia to Australia. These models aggregated several economic variables, including real income, tourism prices, transportation costs and exchange rates. The experimental results demonstrated that real income was not a significant influence factor on those models. Greenidge [8] proposed a structure time series modeling (STM) to capture the high value of the coefficient of determination of a tourism demand regression model. Owing to the several variables employed, three simultaneous time series, trend, seasonality and tourism lifecycle, were implemented to forecast tourist arrivals.

The feasibility of applying artificial intelligence techniques to enhance the accuracy of tourist forecasting models has received considerable interest recently. Pattie and Synder [9] utilized back-propagation neural network (BPNN) models with two hidden layers to forecast monthly overnight backcountry stays in US national parks. Their forecasts are more accurate than those obtained using other traditional time series models with large samples. Law and Au [10] designed a feed-forward neural network incorporating six input nodes and one output node to forecast tourist arrivals in Hong Kong. The experimental results indicated that the proposed neural network outperforms the other multivariate forecasting models. Law [11] applied a BPNN model for forecasting Taiwanese tourist arrivals in Hong Kong. The proposed neural network model contains six independent factors as input nodes, and one output node. The author used a non-linear function to randomly separate data on independent variables into a training data set. This model significantly increased forecasting accuracy. Moreover, Burger et al. [12] presented eight methods for forecasting tourist arrivals from the United States to Durban in South Africa between 1992 and 1998. Burger et al. reported that the neural network approach performed best for unstructured tourist arrival data. Cho [13] compared three models, i.e., exponential smoothing, univariate ARIMA and Elman's model of artificial neural networks, to forecast tourist arrivals from six countries to Hong Kong. According to those forecasts, Elman's neural network model outperforms the other two methods owing to its ability to elucidate the behavior associated with tourist arrival data without clear patterns.

SVMs were originally created to solve classification problems. Recently, SVMs have been successfully applied to solve forecasting problems in numerous fields. Tay and Cao [14] applied SVMs to forecast financial time series. Their numerical results showed that SVMs are superior to a multi-layer back-propagation neural network in financial time series forecasting. Moreover, Wang et al. [15] used SVMs to forecast air quality. They reported that SVMs outperformed conventional radial basis function networks. Cao [16] applied SVM experts in forecasting time series. Generalized expert SVMs have a two-stage neural network architecture. The numerical results demonstrated that SVM experts can achieve better generalization than single SVM models. Pai and Hong [17] applied recurrent SVM with genetic algorithms to forecast regional electric load in Taiwan. They concluded that the presented model outperforms the SVM, ARIMA and regression models. Mohandes et al. [18] applied SVMs to predict wind speed. Their experimental results indicated that the SVM model outperforms multilayer perceptron neural networks as measured by root mean square errors. Pai and Lin [19] proposed a hybrid model with the strength of an ARIMA model and the SVM model in forecasting stock prices problems. By using ten stocks to examine the performance, their numerical results indicated that the proposed hybrid model outperformed other forecasting methods.

Inspired by the mixture of experts (ME) [20,21], this study develops a multifactor support vector machine (MSVM) model for forecasting tourist arrivals. This paper is organized as follows: The MSVM model is introduced in section 2. Subsequently, section 3 uses a numerical example from the literature to investigate the forecasting performance of the proposed model. Finally, conclusions are given in section 4.

2 Multifactor Support Vector Machine Model (MSVM)

2.1 Back-Propagation Neural Networks

The multi-layer back-propagation (BP) neural network is one of the most widely used neural network models. Consider the simplest BP network architecture (**Fig. 1.**) including three layers: an input layer (x), an output layer (o), and a hidden layer (h). The computational procedure of this network is described below:

$$o_i = f\left(\sum_j g_{ij}x_{ij}\right). \tag{1}$$

where o_i denotes the output (final tourism demand forecasting values) of node i , $f(\cdot)$ represents the activation function, g_{ij} is the connection weight between nodes i and j in the lower layer which can be replaced with v_{ji} and w_{kj} , and x_{ij} denotes the input signal from the node j in the lower layer.

The BP algorithm attempts to improve neural network performance by reducing the total error through changing the gradient weights. The BP algorithm minimizes the sum-of-error-square, which can be calculated by:

$$E = \frac{1}{2} \sum_{p=1}^P \sum_{j=1}^K (d_{pj} - o_{pj})^2 \tag{2}$$

where E denotes the square errors, K represents the output layer neurons, P is the training data pattern, d_{pj} denotes the actual output and o_{pj} represents the network output. The BP algorithm is expressed as follows. Let Δv_{ji} denote the weight change for any hidden layer neuron and Δw_{kj} for any output layer neuron,

$$\Delta v_{ji} = -\eta(\partial E/\partial v_{ji}), \quad \Delta w_{kj} = -\eta(\partial E/\partial w_{kj}), \quad i=1,\dots,I; j=1,\dots,J-1; k=1,\dots,K. \tag{3}$$

where η represents the learning rate parameter. Notably, the J th node in **Fig. 1.** is the bias neuron without weight. Eq. (4) express the signal (s_j) to each hidden layer neuron and the signal (u_k) to each neuron in the output layer.

$$s_j = \sum_{i=1}^I v_{ji}x_i; \quad u_k = \sum_{j=1}^{J-1} w_{kj}y_j \tag{4}$$

The error signal terms for the j th hidden neuron δ_{yj} , and for the k th output neuron δ_{ok} are defined as Eq. (5).

$$\delta_{yj} = -\partial E/\partial s_j; \quad \delta_{ok} = -\partial E/\partial u_k \tag{5}$$

Applying the chain rule, the gradient of the cost function with respect to weights v_{ji} and w_{kj} is

$$\partial E/\partial v_{ji} = -\delta_{yj}x_i; \quad \partial E/\partial w_{kj} = -\delta_{ok}y_j \tag{6}$$

the weight change from Eq.(3) can now be written as

$$\Delta v_{ji} = \eta \delta_{yj} x_i ; \Delta w_{kj} = \eta \delta_{ok} y_j . \tag{7}$$

Furthermore, Eq.(5) can be calculated as follows,

$$\delta_{ok} = (d_k - o_k) f'(u_k) ; \delta_{yj} = \left\{ \sum_{k=1}^K o_k \cdot w_{kj} \right\} \cdot f'_j(u_j) . \tag{8}$$

the weights, v_{ji} and w_{kj} , are changed as follows

$$w_{kj} = w_{kj} + \eta \delta_{ok} y_j ; v_{ji} = v_{ji} + \eta f'_j(u_j) x_i \sum_{k=1}^K \delta_{ok} w_{kj} . \tag{9}$$

The constant term, η , is specified at the start of training cycle, and determines the training speed and stability of the network. The most common activation functions are the squashing sigmoid function, such as the logistic and tangent hyperbolic functions.

2.2 Multifactor Support Vector Machine Model

The support vector machines (SVMs) were proposed by Vapnik [22]. The basic concept of the SVM regression is to map nonlinearly the original data x into a higher dimensional feature space. Hence, given a set of data $G = \{(x_i, d_i)\}_{i=1}^N$ (where x_i is the input vector; d_i is the actual value, and N is the total number of data patterns), the SVM regression function is

$$y = f(x) = w \phi_i(x) + b . \tag{10}$$

where $\phi_i(x)$ is the feature of inputs x , and both w and b are coefficients which are estimated by minimizing the regularized risk function:

$$R(C) = C \frac{1}{N} \sum_{i=1}^N L_\epsilon(d_i, y_i) + \|w\|^2 / 2 . \tag{11}$$

where

$$L_\epsilon(d, y) = \begin{cases} |d - y| - \epsilon & |d - y| \geq \epsilon . \\ 0 & \text{others} \end{cases} . \tag{12}$$

and C and ϵ are prescribed parameters, d_i is the actual value at period i . y_i is the estimation value at period i . In Eq. (11), $L_\epsilon(d, y)$ is called the ϵ -insensitive loss function. The loss equals zero if the forecasted value is within the ϵ -tube (Eq. (12)). The second term, $\|w\|^2 / 2$, measures the flatness of the function. Therefore, C is considered to specify the trade-off between the empirical risk and the model flatness. Both C and ϵ are user-determined parameters. Two positive slack variables ζ and ζ^* , which represent the distance from actual values to the corresponding boundary values of ϵ -tube, are introduced. Then, Eq. (11) is transformed into the following constrained form;

Minimize: $R(w, \zeta, \zeta^*) = \|w\|^2 / 2 + C \left(\sum_{i=1}^N (\zeta_i + \zeta_i^*) \right)$. (13)

with the constraints,

$$w \phi(x_i) + b_i - d_i \leq \epsilon + \zeta_i^* , d_i - w \phi(x_i) - b_i \leq \epsilon + \zeta_i , \zeta_i, \zeta_i^* \geq 0 , i=1,2,\dots,N$$

This constrained optimization problem is solved using the following primal Lagrangian form:

$$L(w, b, \zeta, \zeta^*, \alpha_i, \alpha_i^*, \beta_i, \beta_i^*) = \frac{1}{2} \|w\|^2 + C \left(\sum_{i=1}^N (\zeta_i + \zeta_i^*) \right) - \sum_{i=1}^N \alpha_i [w\phi(x_i) + b - d_i + \varepsilon + \zeta_i] - \sum_{i=1}^N \alpha_i^* [d_i - w\phi(x_i) - b + \varepsilon + \zeta_i^*] - \sum_{i=1}^N (\beta_i \zeta_i + \beta_i^* \zeta_i^*) \tag{14}$$

Equation (14) is minimized with respect to primal variables w , b , ζ and ζ^* , and maximized with respect to nonnegative Lagrangian multipliers α_i , α_i^* , β_i and β_i^* . Finally, Karush-Kuhn-Tucker conditions are applied to the regression, and Eq. (12) thus yields the dual Lagrangian,

$$J(\alpha_i, \alpha_i^*) = \sum_{i=1}^N d_i (\alpha_i - \alpha_i^*) - \varepsilon \sum_{i=1}^N (\alpha_i + \alpha_i^*) - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) K(x_i, x_j) \tag{15}$$

subject to the constraints,

$$\sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0, \quad 0 \leq \alpha_i, \alpha_i^* \leq C, \quad i = 1, 2, \dots, N$$

In Eq. (15), α_i and α_i^* are called Lagrangian multipliers that satisfy the equalities, $\alpha_i * \alpha_i^* = 0$. After calculating α_i and α_i^* , an optimal desired weights vector of the regression hyperplane is represented as

$$w^* = \sum_{i=1}^N (\alpha_i - \alpha_i^*) \phi(x_i) \tag{16}$$

Therefore, the regression function could be Eq.(28)

$$f(x, \alpha, \alpha^*) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) K(x, x_j) + b \tag{17}$$

Here, $K(x_i, x_j)$ is called the kernel function. The value of the kernel equals the inner product of two vectors x_i and x_j in the feature space $\phi(x_i)$ and $\phi(x_j)$, i.e., $K(x_i, x_j) = \phi(x_i) * \phi(x_j)$. Any function that satisfies Mercer’s condition by Vapnik [22] can be used as the Kernel function. In this investigation, the Gaussian function, $\exp(-\|x - x_i\|^2 / 2\sigma^2)$, is used in the SVM. The selection of three positive parameters, σ , ε and C , of a SVM model is important to the accuracy of forecasting. However, structural methods for confirming efficiently the selection of parameters efficiently are lacking. Therefore, genetic algorithms are employed in the proposed SVM model to optimize parameter selection (Pai and Hong [17]).

The use of different SVM models to obtain the input vectors of back-propagation is inspired by the so-called “divide-and-conquer” principle. A mixture of experts (ME) architecture was proposed to deal with non-stationary data. The whole input space is partitioned into several disjointed regions. Particularly, the expert networks are used to handle different input regions. The outputs of the expert networks then are combined and fed into a gating network to solve the problem. This work combines support vector machines with back-propagation neural networks to solve multifactor forecasting problems. Fig. 2. illustrates the MSVM architecture. The primary advantage of the MSVM architecture is its ability to automatically obtain suitable input vectors for the back-propagation neural network.

The MSVM model involves three stages. During the first stage, the back-propagation neural networks are trained by the training data set. Secondly, the SVM

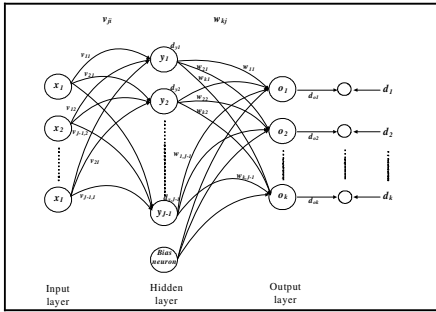


Fig. 1. A Back-propagation network

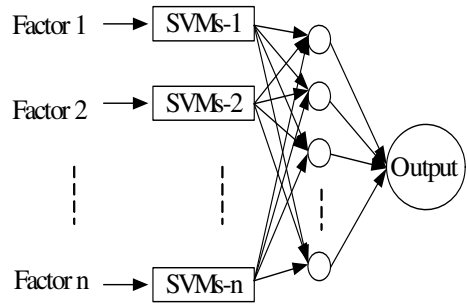


Fig. 2. The MSVM architecture

models are applied for obtaining the suitable input vectors of BP neural networks. Finally, the appropriate input vectors from the SVM model are fed into the well-trained back-propagation network model to yield the final forecasting value (Y), which can be represented as

$$Y = f(\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_n) \quad (18)$$

where f denotes the nonlinear function of back-propagation, and \tilde{y}_n represents the output value from the SVM model of factor n . Rather than using the actual data of each factor, the proposed model adopts the forecasting values from the SVM models as input vectors for well trained back-propagation neural networks. The SVM models use the information of the past behavior of time series to identify the dependency between the future and the past for each factor.

3 A Numerical Example

A numerical example given by Law [11] is used to demonstrate the forecasting capabilities of the proposed model. The data used by Law was Taiwanese demand for travel to Hong Kong from 1967 to 1996. Furthermore, the number of Taiwanese visitors to Hong-Kong (NVT-HK) was assumed to be a function of six factors: service price in Hong-Kong (SP), foreign exchange rate (FER), population of Taiwan (Pop), market expenses in Hong-Kong (Mkt), gross domestic expenditure per person in Taiwan (GDE), and average hotel rate in Hong-Kong (AHR).

The data set is divided into two parts, namely the modeling data set (from 1967 to 1991) and the testing data set (from 1992 to 1996). The modeling data set is used to train the SVMs and the back-propagation neural network. In the SVMs model, 20% of the modeling data is used for validation to avoid over-fitting (Schalkoff [23]). Therefore, five periods of modeling data are used for validation (from 1987 to 1991). The testing data set has never been used for model building or selection, and is used to estimate model performance for future unseen data.

Since the tourist forecasting model contains six factors, the forecasting values of six factors are obtained by a time-series forecasting procedure. Finally, the forecasting values of six factors provide the appropriate input vector for inclusion in the back-propagation neural network forecasting model to obtain the forecast results. Table 1. lists comparisons among forecasting results obtained using the proposed model and

Table 1. Forecasting results of Taiwanese demand for traveling to Hong Kong

Year	Actual	MSVM	Back propagation	Feed forward	Holt's (1,0,1)	Moving average (3)	Multiple regression	Naive
1992	1,625,231	1,656,355	1,609,700	1,191,960	1,376,039	1,243,326	2,305,340	1,284,786
1993	1,762,410	1,755,889	1,627,800	1,192,850	1,741,403	1,412,725	2,328,787	1,625,231
1994	1,650,081	1,679,914	1,672,800	1,193,440	1,880,683	1,557,476	2,375,186	1,762,410
1995	1,776,244	1,772,498	1,799,700	1,194,030	1,745,294	1,679,241	2,544,398	1,650,081
1996	1,820,335	1,772,759	1,865,900	1,194,430	1,874,552	1,729,578	2,638,850	1,776,244

Table 2. Relative percentage errors in the testing data set

Error	MSVM	Back-propagation	Feed-forward	Holt's (1,0,1)	Moving average (3)	Multiple regression	Naïve
MSE	3.53×10⁹	2.51×10 ¹⁰	1.45×10 ¹²	2.39×10 ¹⁰	2.39×10 ¹¹	2.57×10 ¹²	1.65×10 ¹¹
MAPE	2.47%	2.76%	30.76%	7.04%	11.88%	41.23%	9.01%

those obtained using six other forecasting models. The six forecasting models includes back-propagation, feed-forward models, naïve, moving average(3), Holt's two-parameter exponential smoothing(1,0,1) and multiple regression, simply as proposed by Law [11]. The accuracy of the above forecasting models is measured in terms of mean square error (MSE) and mean absolute percentage error (MAPE), as given by Eq.(19). The minimum values of MSE and MAPE indicate that the deviations between actual values and forecast values are very small. According to the comparison results, illustrated in Table 2, various accuracy measurements of MSVM outperformed other forecasting models. Especially, the MSE and the MAPE of the MSVM model are extremely low only 3.53×10⁹ and 2.47 percent, respectively. Janacek & Swift [24] indicated that MSE is a useful criterion for representing the prediction error. Similarly, Witt & Witt [25] also pointed out directly that forecasting accuracy increases with reducing MAPE.

$$MSE = \sum_{i=1}^N (a_i - f_i)^2 / N ; MAPE = \frac{1}{N} \sum_{i=1}^N |a_i - f_i| / |a_i| \times 100\% . \tag{19}$$

4 Conclusions

Based on the architecture of the mixture of experts, support vector machines are combined with back-propagation neural networks for forecasting tourist arrival problems. This investigation has demonstrated that the MSVM model can improve accuracy of forecasting tourist arrivals for non-linearly separable dependent variables. The superior performance of the MSVM model over other approaches has the following causes. First, the mixture of experts is flexible, and can apply the most appropriate model category to capture the data patterns of each input space region. Second, the MSVM model minimizes structural risk, rather than minimizing training errors. This process seeks to minimize the upper bound of the generalization error to enhance generalization performance. This investigation is, to our knowledge, the first attempt to apply SVM models in forecasting tourism demand influenced by various factors. The promising results indicate that the proposed technique provides an effective alternative in forecasting tourism demand for non-linearly separable dependent variables. Future research could use some other search approaches to choose free parameters in the MSVM model to increase the forecasting performance.

References

1. Athiyaman, A., Robertson, R. W.: Time Series Forecasting Techniques: Short-Term Planning in Tourism. *International Journal of Contemporary Hospitality Management* 4 (1992) 8-11
2. Wong, K.: The Relevance of Business Cycles in Forecasting International Tourist Arrivals. *Tourism Management* 18 (1997) 581-586
3. González, P., Moral, P.: An Analysis of The International Tourism Demand in Spain. *International Journal of Forecasting* 11 (1995) 233-251
4. Martin, C. A., Witt, S. F.: Substitute Prices in Models of Tourism Demand. *Annals of Tourism Research* 15 (1998) 255-268
5. Sheldon, P. J., Var, T.: Tourism Forecasting: A Review of Empirical Research. *Journal of Forecasting* 4 (1985) 183-195
6. Lim, C.: An Econometric Classification And Review of International Tourism Demand Models. *Tourism Economics* 3 (1997) 69-81
7. Lim, C., McAleer, M.: A Cointegration Analysis of Annual Tourism Demand by Malaysia for Australia. *Mathematics and Computers in Simulation* 59 (2002) 197-205
8. Greenidge, K.: Forecasting Tourism Demand--An STM Approach. *Annals of Tourism Research* 28 (2001) 98-112
9. Pattie, D.C., Synder, J.: Using A Neural Network to Forecast Visitor Behavior. *Annals of Tourism Research* 23 (1996) 151-164
10. Law, R., Au, N.: A Neural Network Model to Forecast Japanese Demand For Travel to Hong Kong. *Tourism Management* 20 (1999) 89-97
11. Law, R.: Back-propagation Learning in Improving The Accuracy of Neural Network-based Tourism Demand Forecasting. *Tourism Management* 21 (2000) 331-340
12. Burger, C. J. S. C., Dohnal, M., Kathrada, M., Law, R.: A Practitioners Guide to Time-Series Methods For Tourism Demand Forecasting—A Case Study of Durban, South Africa. *Tourism Management* 22 (2001) 403-409
13. Cho, V.: A Comparison of Three Different Approaches to Tourist Arrival Forecasting. *Tourism Management* 24 (2003) 323-330
14. Tay, F. E. H., Cao, L.: Application of Support Vector Machines in Financial Time Series Forecasting. *OMEGA* 29 (2001) 309-317
15. Wang, W., Xu, Z., Lu, J. W.: Three Improved Neural Network Models for Air Quality Forecasting. *Engineering Computations* 20 (2003) 192-210
16. Cao, L., Gu, Q.: Dynamic Support Vector Machines For Non-stationary Time Series Forecasting. *Intelligent Data Analysis* 6 (2002) 67-83
17. Pai, P.F., Hong, W.C.: Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms. *Electric Power Systems Research* 74 (2005) 417-425.
18. Mohandes, M. A., Halawani, T. O., Rehman, S., Hussain, A. A.: Support Vector Machines For Wind Speed Prediction. *Renewable Energy* 29 (2004) 939-947
19. Pai, P. F., Lin, C. S.: 2004, A Hybrid ARIMA and Support Vector Machines Model in Stock Price Forecasting. *OMEGA* 33 (2005) 497-505.
20. Kwok, T. J.: Support Vector Mixture For Classification And Regression Problems. In: Proceedings of the 14th International Conference on Pattern Recognition, pp 255-258 (1998)
21. Milidui, R. L., Machado, R. J., Rentera, R. P.: Time-Series Forecasting Through Wavelets Transformation And A Mixture of Expert Models. *Neurocomputing* 28 (1999) 145-156
22. Vapnik, V. (eds.): *The Nature Of Statistic Learning Theory*. Springer-Verlag, New York (1995)
23. Schalkoff, R. J.: *Artificial Neural Networks*. McGraw-Hill, New York (1997)
24. Janacek, G., Swift, L.: *Time Series Forecasting, Simulation, Applications*. Ellis Horwood, New York (1993)
25. Witt, S. F., Witt, C. A.: *Modeling And Forecasting Demand In Tourism*. Academic Press, London (1992).

A Study of Modelling Non-stationary Time Series Using Support Vector Machines with Fuzzy Segmentation Information

Shaomin Zhang, Lijia Zhi, and Shukuan Lin

College of Information and Engineering, Northeastern University,
Shenyang 110004, China
zhilijia@hotmail.com

Abstract. We present a new approach for modelling non-stationary time series, which combines multi-SVR and fuzzy segmentation. Following the idea of Janos Abonyi [11] where an algorithm of fuzzy segmentation was applied to time series, in this article we modify it and unite the segmentation and multi-SVR with a heuristic weighting on ϵ . Experimental results showing its practical viability are presented.

1 Introduction

Recently support vector machines [2] have been a promising method for data classification and regression. Also, its application in time series modelling can be found in, for example, [1] [3] [4] [5].

A sequence of observed data, x_1, x_2, \dots, x_l , ordered in time, is called time series. Real-world time series can be taken from business, physical, social and behavioral science, economics, etc. Usually these time series are non-stationary. One useful method to handle such time series is to find a partitioning of several segments that are internally homogeneous [7] and then model each of them. In the context of the real-world environment, the changes of the time series are usually vague and do not focused on any particular time point. Therefore, it is not practical to define crisp bounds of the segments. Naturally, fuzzy logic can be exploited to present such data points [8] [9] [13] [14] [15], and references therein.

In this paper, we aim to model non-stationary time series by combing segmentation information. Due to the nature of the fuzzy algorithm, we should especially consider the fuzziness of the data points in the intersectional part between two larger segments. In addition to a modified formulation, this paper is novel in presenting a practical modelling non-stationary time series approach which combines the fuzzy information of data points in the intersectional part with multi-SVR.

This paper is organized as follows. In Section 2, we define the problem and present our approach. Section 3 contains experimental results on some data sets. Finally, conclusions are given in Section 4.

2 Modelling Non-stationary Time Series with Fuzzy Segmentation Information

2.1 Fuzzy Time Series Segmentation

Given a stationary time series $T = \{y_t \mid 1 \leq t \leq l\}$, usually we can model it by using $y_t = f(x_t)$, where $x_t = [y_t - 1, y_t - 2, \dots, y_t - d]^T$, and d is the embedding dimension. However, this is not true if this time series is non-stationary. We can not determine one unique f to satisfy the whole sequence. Instead, we can assume that there are m functions (i.e. m modes) such that

$$y_t = \sum_{i=1}^m p_i^t f_i(x_t), \quad \forall t = 1, \dots, l, \tag{1}$$

subject to

$$\sum_{i=1}^m p_i^t = 1, \quad \forall t = 1, \dots, l$$

$$0 \leq p_i^t \leq 1, \quad \forall t = 1, \dots, l, \quad i = 1, \dots, m$$

to model the given time series. Usually, when treating real-world time series, we can not get the number of data sources exactly. So, often we estimate the value of m based on some information.

Segmentation is a useful way to handle this problem. Traditional c -segmentation strategy is to part time series T into c non-overlapping segments $S_T^c = \{S_i(a_i, b_i) \mid 1 \leq i \leq c\}$, a segment of T is a set of consecutive time points. And then, model these segments respectively. Often use one unique function to model one segment.

Due to the changes of the variables are usually vague. So, it is rational and practical to consider a set of consecutive data points should be as turning points between different modes. Therefore fuzzy logic is naturally exploited to describe this problem.

In [11], the authors proposed a fuzzy clustering based time series segmentation algorithm which assumes that the data can be modelled as a mixture multivariate (including time as a variable) Gaussian distribution. Here, we use a cost function similar to [11] but considering one time-variant variable except for time t :

$$J = \sum_{i=1}^c \sum_{k=1}^l (\mu_{i,k})^m D_{i,k}^2(\eta_i, \mathbf{z}_k) = \sum_{i=1}^c \sum_{k=1}^l (\mu_{i,k})^m D_{i,k}^2(v_i^t, t_k) D_{i,k}^2(v_i^x, x_k) \tag{2}$$

subject to

$$\sum_{i=1}^c (\mu_{i,k}) = 1, \quad k = 1, \dots, l$$

where $\mu_{i,k}$ represents the degree of membership of the vector $\mathbf{z}_k = [t_k, x_k]^T$ is in the i -th cluster ($i = 1, \dots, c$); and $m \in [1, \infty]$, is a weighting exponent that determines the fuzziness of the resulting clusters (usually $m = 2$); and η_i represents cluster prototypes.

Distance measure $D_{i,k}^2(\eta_i, \mathbf{z}_k)$ consists of two terms, $D_{i,k}^2(v_i^t, t_k)$ is the distance between the k -th data point and the v_i^t center of the i -th segment in time. $D_{i,k}^2(v_i^x, x_k)$ is the distance between the cluster prototype and the data point.

The probability $p(\mathbf{z}_k | \eta_i)$ that the \mathbf{z}_k data point belongs to the i -th cluster can get by:

$$p(\mathbf{z}_k | \eta_i) = \frac{1}{D^2(\mathbf{z}_k, \eta_i)} \tag{3}$$

We summarize the algorithm in [11] as follows, with one time-variant variable.

Algorithm 2.1

- 1) Determine the number of clusters c based on some distance measure;
- 2) Optimize (2), update $\mu_{i,k}$ with

$$\mu_{i,k}^{(l)} = \frac{1}{\sum_{j=1}^c (D_{i,k}/D_{j,k})^{2/(m-1)}}, \quad 1 \leq i \leq c, \quad 1 \leq k \leq l. \tag{4}$$

- 3) If certain stopping criteria are met, terminate the algorithm. Otherwise, go to step 2.

Then we can get $p(\mathbf{z}_k | \eta_i)$. In [11] [16], can find more details.

Next we propose to modify formulation of support vector regression (SVR).

2.2 A Modified SVR Formulation

The idea of SVR is to map input data into a higher dimensional space and find a function which approximates the hidden relationships of the given data. The standard formulation is as follows:

$$\begin{aligned} \min_{w,b,\xi,\xi^*} \quad & \frac{1}{2} \omega^T \omega + C \sum_{t=1}^l (\xi^t + \xi^{t,*}) \\ \text{subject to} \quad & -\epsilon - \xi^{t,*} \leq y_t - (\omega^T \phi(x_t) + b) \leq \epsilon + \xi^t \\ & \xi^t \geq 0, \quad \xi^{t,*} \geq 0, \quad t = 1, \dots, l \end{aligned} \tag{5}$$

where ω and b are unknown parameters to be estimated, ξ and ξ^* indicate errors, ϵ is the tolerance, and ϕ is the function that maps data x_t to a higher dimensional space.

Here, we take $\hat{f}(x) = \omega^T \phi(x) + b$.

The standard SVR considers a uniform precision ϵ for all data. We propose to use different weights ϵq_i^t for each data point. For each $i = 1, 2, \dots, m$, we have the following modification

$$\begin{aligned} \min_{w_i,b_i,\xi_i,\xi_i^*} \quad & \frac{1}{2} \omega_i^T \omega_i + C \sum_{t=1}^l (\xi_i^t + \xi_i^{t,*}) \\ \text{subject to} \quad & -\epsilon q_i^t - \xi_i^{t,*} \leq y_t - (\omega_i^T \phi(x_t) + b_i) \leq \epsilon q_i^t + \xi_i^t \\ & \xi_i^t \geq 0, \quad \xi_i^{t,*} \geq 0, \quad t = 1, \dots, l \end{aligned} \tag{6}$$

different ϵ means different precision requirement for each data point.

The Lagrangian dual problem of (6) can be derived by a similar way as that of standard SVR. Here we give the resulting formulation:

$$\begin{aligned} \min_{\alpha, \alpha^*} \quad & \frac{1}{2}(\alpha - \alpha^*)^T K(\alpha - \alpha^*) - \epsilon q_i^t \sum_{t=1}^l (\alpha_t + \alpha_t^*) + \sum_{t=1}^l y_t (\alpha_t - \alpha_t^*) \quad (7) \\ \text{subject to} \quad & \sum_{t=1}^l (\alpha_t - \alpha_t^*) = 0, \\ & 0 \leq \alpha_t, \alpha_t^* \leq C, \quad t = 1, \dots, l. \end{aligned}$$

2.3 Combining the Modified SVR with Fuzzy Segmentation Information

In this subsection, we aim to determine the number of modes m in (1); and approximate each $f_i(x_t)$ by using the above modified SVR and exploiting the fuzzy segmentation information as heuristic weight on ϵ .

First we use algorithm 2.1 to segment given time series. The segmentation results in several overlapping segments S_i ($i = 1, \dots, R$). Then we define some thresholds to divide these segments into two classes: larger segments LS_j ($j = 1, \dots, h, h < R$) and intersectional parts I_k ($k = 1, \dots, g, g < R$), the sum of h and g may less than R . They are non-overlapping. We use I_k to stand for the intersectional part between two larger segments LS_k and LS_{k+1} . Experiment in Section 3 show this situation.

Here, we set a threshold for probability $p(\mathbf{z}_k | \eta_i)$, $p_{thre} = 0.9$; and a low bound of segment size $L_{low} = 20$. A segment $S_i(a_i, b_i)$ is a larger segment LS if there exists bl_i and br_i ($a_i < bl_i < br_i < b_i$), which satisfy the following conditions:

$$p(\mathbf{z}_k | \eta_i) < p_{thre} \quad \forall k < bl_i \text{ and } k > br_i, \quad |S_i(bl_i, br_i)| \geq L_{low} \quad (8)$$

Because of the existing of noise, not all $p(\mathbf{z}_k | \eta_i)$ ($bl_i \leq k \leq br_i$) strictly exceed p_{thre} . Then, using bl_i and br_i ($i = 1, \dots, h$) as the borderlines, we repartition the whole data set into several non-overlapping data sets LS and I .

In (2), time distance is a considered factor, $p(\mathbf{z}_k | \eta_i)$ is a exponential function with base e of variables t_k and x_k . For the i -th intersectional part I_i , the influence from two adjacent larger segments LS_i and LS_{i+1} should be taken into account, and that from other larger segments LS_j ($j < i \cup j > i + 1$) can be ignored.

Owing to the internal homogeneity, we use a single standard SVR to model one larger segment LS , as the way treating stationary time series. So for each LS_i ($i = 1, \dots, h$) the value of m in (1) is set to 1.

But for the intersectional part I_i , we should consider it more carefully. I_i can be considered as the turning points between different modes. I_i may contain several segments S . Therefore, we must consider the effects of such segments.

We use $\mathbf{z}_k \in I_i$ to indicate that data point \mathbf{z}_k is in I_i ; $t \in I_i$ to indicate that the time point t is in I_i ; $\eta_j \cap I_i \neq \phi$ and $S_j \cap I_i \neq \phi$ indicate the same meaning that the segment S_j is in I_i . For each data point \mathbf{z}_k in I_i ($\mathbf{z}_k \in I_i$), we check

the sum of $p(\mathbf{z}_k | \eta_j)(\eta_j \cap I_i \neq \phi)$, and find that most of them are less than 1. That means the data points in an intersectional part I_i also have their own property. For I_i 's own property, we assume it can be effectively modelled as a Gaussian distribution. Then, the number of I_i 's modes m in (1) can be get. That is $m = n + 1$, where n is the number of segments S_j in $I_i(S_j \cap I_i \neq \phi)$, and '1' is the assumed a Gaussian distribution function. Now, we propose an algorithm to achieve the goal of this subsection:

Algorithm 2.3

- 1) Using the result of **Algorithm 2.1** and the method presented above to determine n that is the number of segments S in an intersectional part I_i ;
- 2) Get the number of the functions, m :

$$m = \begin{cases} 1, & \text{if } \mathbf{z}_t \in LS \cap \forall t = 1, \dots, l \\ n + 1, & \text{if } \mathbf{z}_t \in I \cap \forall t = 1, \dots, l \end{cases} \tag{9}$$

3)

1. For a larger segment LS_i , use a single standard SVR to model it;
2. For an intersectional part I_i , Solve (7) by weighting ϵ of $f_1(x_t), \dots, f_n(x_t)$ with setting the weighting factor $q_j^t = 1 - p(\mathbf{z}_k | \eta_j)(\mathbf{z}_k \in I_i, \eta^j \cap I_i \neq \phi)$ ($j = 1, \dots, n$) respectively. Then get approximate functions $\hat{f}_1(x_t), \dots, \hat{f}_n(x_t)$;

4) As for the Gaussian assumption, weight the ϵ of $f_m(x_t)$ according to Gaussian distribution:

$$q_m^t = e^{-\frac{(t-\mu)^2}{2\sigma^2}} \tag{10}$$

set μ, σ through following formulations:

$$\mu \in \arg \min_t \{sum(p(\mathbf{z}_t | \eta_j)) | \mathbf{z}_t \in I_i, \eta_j \cap I_i \neq \phi\} \tag{11}$$

$$e^{-\frac{(t-\mu)^2}{2\sigma^2}} = 1 - sum(p(\mathbf{z}_b | \eta_j)) \quad (\eta_j \cap I_i \neq \phi, \quad t \in I_i) \tag{12}$$

$$\text{where} \quad b \in \arg \max_t \{|t - \mu| | t \in I_i\} \tag{13}$$

$$\text{therefore} \quad \sigma^2 = -\frac{(t - \mu)^2}{2 \ln(1 - sum(p(\mathbf{z}_b | \eta_j)))} \quad (\eta_j \cap I_i \neq \phi, \quad t \in I_i) \tag{14}$$

Then by solving (7), we get the last function $\hat{f}_m(x_t)$.

Through **Algorithm 2.3**, we can get m functions $\hat{f}_1(x_t), \dots, \hat{f}_m(x_t)$, for each larger segment $LS_j(j = 1, \dots, h)$ and intersectional part $I_k(k = 1, \dots, g)$. Next, we calculate each function's weight in (1), and complete the model.

2.4 Computing p_i^t

For each larger segment LS_j , we just use a single SVR to model it, so it has nothing to do with functions' weight in (1). Therefore in this subsection, we only discuss how to model the intersectional part I_i .

Based on the current functions (i.e. $\hat{f}_i(x_t), i = 1, \dots, m$), we then find the best p_i^t . This is via solving a convex optimization problem with variables p_i^t :

$$\begin{aligned} \min_{p_i^t} & \sum_{t=1}^l (y_t - \sum_{i=1}^m p_i^t \hat{f}_i(x_t))^2 + \lambda \sum_{t=2}^l \sum_{i=1}^m (p_i^t - p_i^{t-1})^2, \\ \text{subject to} & \sum_{i=1}^m p_i^t = 1, t = 1, \dots, l \\ & 0 \leq p_i^t \leq 1, \forall t = 1, \dots, l, i = 1, \dots, m \end{aligned} \tag{15}$$

where l stands for the number of data points in the intersectional part I_i . Here λ serves as a penalty parameter so that p_i^t and p_i^{t-1} are close, which reflects that $p_i^t, t = 1, \dots, l$ are slowly changing.

3 Experiments

We choose 350 consecutive silicon content of the hot metal as a data set. For the modified SVR, we set $C = 0.5, \epsilon = 0.25$. The RBF kernel $K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2}$ is used with $\gamma = 1.0$, in (15) $\lambda = 0.005$. The embedding dimension is five. That is y_t is the one-step ahead value of $x_t = [y_{t-1}, y_{t-2}, \dots, y_{t-5}]^T$.

The original data set is in Fig.1. By **Algorithm 2.1** we get the fuzzy segments, and then we divide them into two larger segments and two intersectional parts using (8). The results are showed in Fig.2. In the intersectional part I, $m = 3$. Using **Algorithm 2.3** and solving (15), we get a multi-mode consisting 3 functions. For intersectional part II $m = 4$. As a comparison, we use a

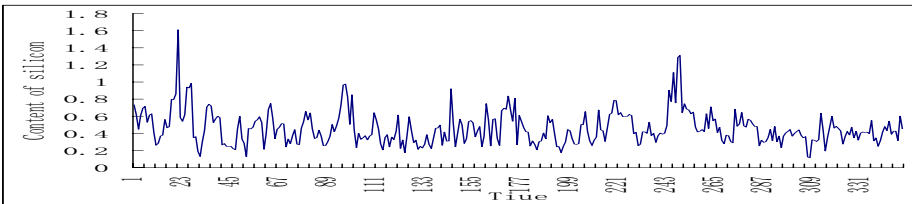


Fig. 1. The original data set of the silicon content

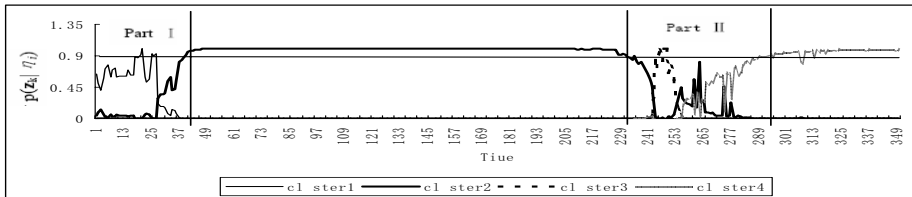


Fig. 2. Fuzzy segmenting and dividing results, Part I and Part II are two intersectional parts

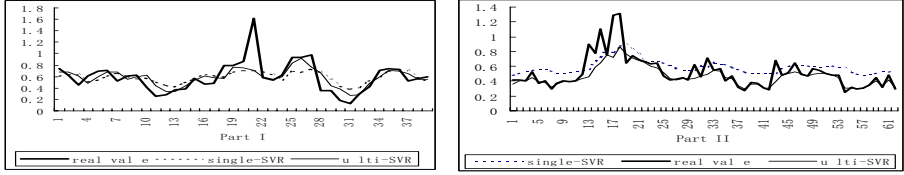


Fig. 3. Comparison between the single-SVR and the multi-SVR

single SVR to model the intersectional part. Fig.3 reflects the results from two methods.

In order to evaluate the result accuracy, the following formula is used to compute the approximation error:

$$NMSE = \frac{\sum_{j=1}^N (t_j - t_j^*)^2}{\sum_{j=1}^N (t_j - \bar{t})^2} \tag{16}$$

where N is the number of the data in the testing set, t_j is the real value. t_j^* is its approximate value.

For Part I: $NMSE_{single-SVR} = 0.656625$ $NMSE_{multi-SVR} = 0.50201$
 For Part II: $NMSE_{single-SVR} = 0.695591$ $NMSE_{multi-SVR} = 0.414596$.

4 Conclusions

This paper presents a new modelling method for non-stationary time series. The method combines multi-SVR with fuzzy segmentation. We choose SVR as the base mode because of its better generalization performance. In terms of the feature of fuzzy segmentation, we proposes a weighting method using probability $p(\mathbf{z}_k|\eta_i)$ as the heuristic weight on toleranceto train SVR and get approximated functions $\hat{f}_i(x_t)(i = 1, \dots, n)$. For Gaussian assumption, we weight ϵ with the value of Gaussian distribution, and get the function $\hat{f}_m(x_t)$. Then we get the best value of each function’s weight p_i^t in (1) by solving a convex quadratic optimization problem.

Generally, the quantity of the data in an intersectional part is small. In the experiment, intersectional part I has 39 data points and part II has 62 data points. For such quantity of training data, it is difficult for a single SVR to learn the underlying relations. So multi-SVR is proper to modelling such data.

During our experiments, we use the cross validation for reference to choose parameters, but set C a small value (we set $C = 0.5, 1$), to prevent SVR becoming overcomplicated and to lose its good generalization performance [2]. The experiments show that the proposed approaches enhance approximating accuracy for the data of intersectional parts than single-SVR. In our future work, we hope to promote the approximating accuracy without increasing the complexity drastically.

Acknowledgments

This work is supported by the Natural Science Foundation of Shenyang City, Liaoning Province, China (Grant No.1041036-1-06-07), and the Natural Science Foundation of Liaoning Province, China(20042015).

References

1. Muller, K., Smola, A., Ratsch, G., Scholkopf, B., Kohlmorgen, J., Vapnik, V.: Using Support Vector machines for Time Series Prediction. Scholkopf, B., Burges, J., Smola, A., *Advances in Kernel Methods: Support Vector Machine* MIT Press. (1999)
2. Vapnik, V.: *Statistical Learning Theory*. Wiley, New York, NY. (1998)
3. Muller, K., Smola, A., Ratsch, G., Scholkopf, B., Kohlmorgen, J., Vapnik, V.: Predicting time series with support vector machines. Gerstner, W., Germond, A., Hasler, M., Nicoud, J. editors, *Proceedings of ICANN97::Proc of the Int. Conf. on Artificial Neural Networks*, Berlin, Springer.(1997)999–1004..
4. M. W. Chang, C.-J. Lin, R. C. Weng.: Analysis of switching dynamics with competing support vector machines. In *Proceedings of IJCNN*. (2002)
5. M.W. Chang, C.-J. Lin, R. C. Weng.: Analysis of nonstationary time series using support vector machines. In Seong-Whan Lee and Alessandro Verri, editors, *Pattern Recognition with Support Vector Machines – First International Workshop, SVM 2002, LNCS 2388*, Springer, Aug (2002)160–170
6. Hathaway, R.J., Bezdek, J.C.: Switching regression models and fuzzy clustering. *IEEE Transactions on Fuzzy Systems*, **1(3)** (1993)195–204,
7. Vasko, K., Toivonen, H.: Estimating the number of segments in time series data using permutation tests, *IEEE International Conference on Data Mining*. (2002)466–473
8. Geva, A.B.: Hierarchical-fuzzy clustering of temporal-patterns and its application for time-series prediction, *Pattern Recognition Letters* **20** (1999)1519–1532.
9. Abonyi, J.: *Fuzzy Model Identification for Control*, Birkhauser Boston. (2003)
10. Abonyi, J., Babuska, R., Szeifert, F.: Modified Gath-Geva fuzzy clustering for identification of takagi-sugeno fuzzy models. *IEEE Transactions on Systems, Man, and Cybernetics*, **32(5)** (2002)312–321
11. Abonyi, J., Feil, B., Nemeth, S., Arva, P.: Fuzzy Clustering Based Segmentation of Time-Series. In Berthold, M.R. editors, *IDA 2003, LNCS 2810* (2003)275–285
12. B. J. Chen, M. W. Chang and C. J. Lin.: Load forecasting using support vector machines: a study on EUNITE competition 2001, report for EUNITE Competition for Smart Adaptive System. Available:<http://www.eunite.org>
13. Baldwin, J.F., Martin, T.P., Rossiter, J.M.: Time series modeling and prediction using fuzzy trend information. *Proceedings of 5th International Conference on Soft Computing and Information Intelligent Systems*.(1998)499–502
14. Hoppner, F., Klawonn, F.: Fuzzy clustering of sampled functions. In *NAFIPS00, Atlanta, USA*.(2000)251–255
15. Last, M., Klein, Y., Kandel, A.: Knowledge discovery in time series database. *IEEE Transactions on Systems, Man, and Cybernetics*, **31(1)** (2000)160–169
16. Abonyi, J., Feil, B., Nemeth, S., Arva, P.: Modified Gath-Geva Clustering for Fuzzy Segmentation of Multivariate Time-series. available:[http://www.fmt.vein.hu/softcomp/segment/Abonyi segment.pdf](http://www.fmt.vein.hu/softcomp/segment/Abonyi%20segment.pdf)

Support Vector Machine Based Trajectory Metamodel for Conceptual Design of Multi-stage Space Launch Vehicle

Saqlain Akhtar and He Linshu

School of Astronautics Beihang University, 37-XueYuan Lu 100083,
Beijing, China
saqlainghumman@hotmail.com

Abstract. The design of new Space Launch Vehicle (SLV) involves a full set of disciplines – propulsion, structural sizing, aerodynamics, mission analysis, flight control, stages layout – with strong interaction between each other. Since multidisciplinary design optimization of multistage launch vehicles is a complex and computationally expensive. An efficient Least Square Support Vector Regression (LS-SVR) technique is used for trajectory simulation of multistage space launch vehicle. This newly formulation problem-about 17 parameters, linked to both the architecture and the command (trajectory optimization), 8 constraints - is solved through hybrid optimization algorithm using Particle Swarm Optimization (PSO) as global optimizer and Sequential Quadratic Programming (SQP) as local optimizer starting from the solution given by (PSO). The objective is to find minimum gross launch weight (GLW) and optimal trajectory during launch maneuvering phase for liquid fueled space launch vehicle (SLV).The computational cost incurred is compared for two cases of conceptual design involving exact trajectory simulation and with Least Square Support Vector Regression based trajectory simulation.

1 Introduction

The optimization of aerospace systems is an iterative process that requires computational models embodied in complex and expensive analysis software. This paradigm is well exemplified by the field of Multidisciplinary Design Optimization (MDO) which attempts to exploit the synergism of mutually interacting disciplines in order to improve the performance of a given design, while increasing the level of confidence that the designer places on the outcome of the design itself. MDO methods, particularly those based on high-fidelity analyses; greatly increase the computational burden and complexity of the design process. For this reason, high-fidelity analysis software typically used in single discipline designs may not be suitable for direct use in MDO^[1]. Faced with these problems, the alternative of using approximation models of the actual analysis software has received increased attention in recent years. A second advantage of using approximation models during the optimization process is that, since their evaluation is inexpensive, they can be used with optimization algorithms which do not rely on the computation of sensitivity derivatives.

1.1 Conceptual Design Implementation

Conceptual design refers to systems studies conducted early in the design process and intended to reveal trends and allow relative comparisons among alternatives. Such conceptual design studies provide quantitative data that can be used by decision makers while the design is still flexible and before the greatest share of life cycle costs are committed. A brief overview for conceptual design of launch vehicle has accounted for in [2]. Population based, non-gradient, stochastic direct search optimization methods are therefore attractive choices for such problem as they are easy to implement and effective for highly nonlinear problems. The Particle Swarm Optimization (PSO) is a highly effective tool for the exploration of large-scale, nonlinear design spaces and, when combined with gradient based search techniques, may provide a more computationally efficient means of identifying near optimal designs.

2 Objective Selection in Launch Vehicle Design

Minimum gross launch weight is taken as objective function. Dynamic penalty function is used to handle in flight and terminal constraints^[3]. In this analysis, design of an expendable 3-stage liquid fueled (LOX-Kerosene) Space Launch Vehicle is sized to deliver a 450kg payload to 500km altitude low earth circular orbit. SLV mission velocity and corresponding altitude are formulated as trajectory constraints. Axial overload constraint is implemented to restrict it below 12g. The parameter, angle of attack times dynamic pressure $(q-\alpha)_{\max}$ was constrained below 15000 kpa-deg structural design limit to ensure that the aerodynamic loads did not exceed the structural capability of the vehicle. During launch maneuver; the maximum angle of attack is constrained to be below 8 deg and to ensure that it is zero during transonic phase.

2.1 Design Variables

For conceptual design of liquid fueled SLV the system design variables for each stage are; chamber pressure p_c , nozzle exit plane pressure p_e , body diameter d , fineness ratio λ , throat diameter d_t , where maximum angle of attack α_m and variable b are for launch maneuver trajectory optimization. The total number of design variables is 17.

3 Weight Analysis

The nozzle throat area may be expressed as:
$$A_t = \frac{F}{p_c \cdot C_f} \quad (1)$$

Mass of propellant:
$$m_{pi} = V_{pi} \cdot \rho_{pav} \quad (2)$$

The rest of the mass model can be found in [4].

4 Trajectory Analysis

To analyze the ascent fight-path, a three degree-of-freedom trajectory model was developed and simulated in SIMULINK. Within the model, the equations of motion are numerically integrated from an initial to a terminal set of state conditions. Within the present investigation, the vehicle is treated as a point-mass, flight in 2D over spherical non-rotating earth was assumed, which implied that the Coriolis and centrifugal pseudo forces were negligible. The 1976 standard atmosphere (no winds) was used. The trajectory was optimized during maneuvering phase, after transonic phase SLV turning was considered under gravity. A vehicle is modeled as a group of stages, with each stage having its own aerodynamic and propulsion inputs. After two power stages there was a coasting stage until mission height was achieved and finally last stage engine was ignited to achieve mission orbit velocity. The complete set of differential equations for this trajectory model is taken from [4]. Interference forces are not considered in this simple, but fast and efficient approach.

5 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a recent addition to a growing collection of non-gradient based, probabilistic search algorithms. PSO is based on the social behavior that a population of individuals adapts to its environment by returning to promising regions that were previously discovered. This adaptation to the environment is a stochastic process that depends on both the memory of each individual as well as the knowledge gained by the population. More information on PSO may be found in [5]. Over successive generations; the particles “evolves” towards an optimal solution to locate near optimum launch mass of SLV. Simple Parameters of PSO used for SLV design in this study are e.g. Swarm Size as 80, Number of Generations as 200, Cognitive Acceleration c_1 as 1.5, Social Acceleration c_2 as 1.5, Minimum velocity step V_{min} as 0.4, Maximum velocity step V_{max} as 0.9 and Error goal as $1e-4$. The solution found by the PSO using exact analysis is given as an initial guess to SQP^[6] which acts as a local optimizer and uses exact analysis to determine objective function. The performance of this combination is shown in Table 1 for five test runs. Fig 1 shows evolution of PSO and overall optimization architecture is shown in Fig 2.

Table 1. Solution from Particle Swarm Optimization and SQP using exact analysis

Test Run	Solution from PSO			Solution from SQP with initial guess From PSO			Total Exact Analysis
	Generations	Fun Count	Weight [ton]	Iterations	Fn Count	Weight [ton]	
1	159	12720	30.23	25	511	27.65	13231
2	152	12160	30.66	18	424	26.91	12584
3	132	10560	30.65	21	492	27.15	11052
4	149	11920	29.51	17	405	26.85	12325
5	187	14960	29.95	15	357	27.05	15317

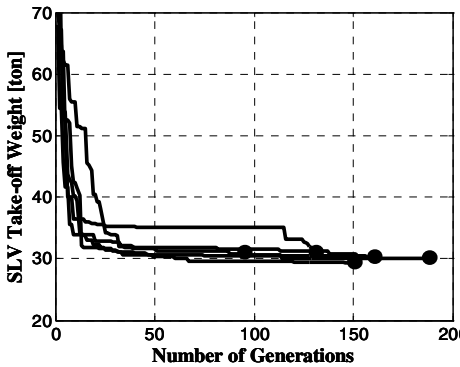


Fig. 1. Evolution of PSO (Exact Analysis)

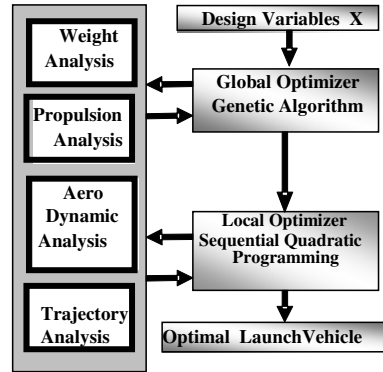


Fig. 2. Objective Function computation process (Exact Analysis)

6 Trajectory Metamodel

Much of today's engineering analysis requires running complex and computationally expensive analysis and simulation codes, such as Multidisciplinary Optimization of multistage space launch vehicle in aerospace area. Optimizing techniques based on Particle Swarm Optimization are very attractive in terms of robustness and finding global solution but at the expense of large number of fitness evaluation and such of exact analysis and substantial amount of computational resources in each of exact analysis. To circumvent this adverse behavior it is proposed to incorporate Least Square Support Vector Machine (LS-SVM) as a metamodel to replace the exact trajectory analysis module. Support Vector Machines is a powerful methodology for solving problems in nonlinear classification, function estimation which has also led to many other recent developments in kernel based methods in general^[7].

6.1 Architecture of Least Square Support Vector Machine

Some typical choices of positive definite kernels are linear, polynomial, Radial Basis Function (RBF) and Multi Layer Perceptrons (MLP) kernel. In classification problems one frequently employs the linear, polynomial and RBF kernel. In nonlinear function estimation and nonlinear modeling problems one often uses RBF kernel^[7].

In order to make a LS-SVM model with RBF kernel, we need two important parameters: first is γ (gam), which is the regularization parameter, determining the trade-off between the fitting error minimization and smoothness second is σ^2 (sig2), which is the bandwidth. The tuning of these two parameters is utmost important trade job. For the metamodeling of SLV trajectory a cascade of three LS-SVM are proposed with Multiple Output Regression. The value of sig2 & gam

Table 2. Least Square Support Vector Machine Architecture and approximation performance

S.No	RBF Kernel Least Square Support Vector Machine	σ^2 (sig2)	γ (gam)	Mean Absolute Error (MAE)
1	1 st Stage Velocity, km/s	15	10	0.0244
2	2 nd Stage Velocity, km/s	20	5	0.0468
3	Coasting Velocity, km/s	15	10	0.0954
4	3 rd Stage Velocity, km/s	15	10	0.06666
5	1 st Stage Height, km	15	10	2.2059
6	2 nd Stage Height, km	20	5	2.4994
7	Final Height, km(Coasting)	15	10	5.9549
8	1 st Stage Axial Overload, g	15	10	0.1554
9	2 nd Stage Axial Overload, g	20	10	0.2915
10	3 rd Stage Axial Overload, g	15	10	0.6283
11	1 st Stage Flight path angle, deg	15	10	0.1021
12	2 nd Stage Flight path angle, deg	20	5	0.0116
13	1 st Stage Lateral Overload, g	15	10	0.0627

parameters and model capability to map the input & output by Mean Absolute Error are given in Table 2.

6.2 Training of Least Square Support Vector Machine

Training data of 2808 Space-filling design points is generated using Latin Hypercube Sampling (LHS) technique. The LS-SVMs are trained from a set of design points obtained from LHS. The performance of trained LS-SVM is evaluated by comparing its simulated output with target values from test data of 950 Latin Hypercube Sampled design points. This training and evaluation procedure is repeated by different values of sig2 and gam combinations. After training, the support value and bias term for each LS-SVM is saved for LS-SVM cascade being used in optimization routine. The performance of LS-SVM cascade is shown in Fig 3.

Table 3. Solution from Particle Swarm Optimization and Sequential Quadratic Programming using Least Square Support Vector Regression

Test Run	Fn Count in LS-SVM Training = 2808	Solution from PSO			Solution from SQP with initial guess From PSO			Total Exact Analysis
		Generations	Fn Count	Weight [ton]	Iterations	Fn Count	Weight [ton]	
1		178	14240	33.21	35	815	27.15	3623
2		183	14640	32.21	63	1176	26.97	3984
3		126	10080	32.19	57	1068	27.11	3876
4		197	15760	31.78	44	834	26.64	3642
5		200	16000	31.33	37	673	26.89	3481

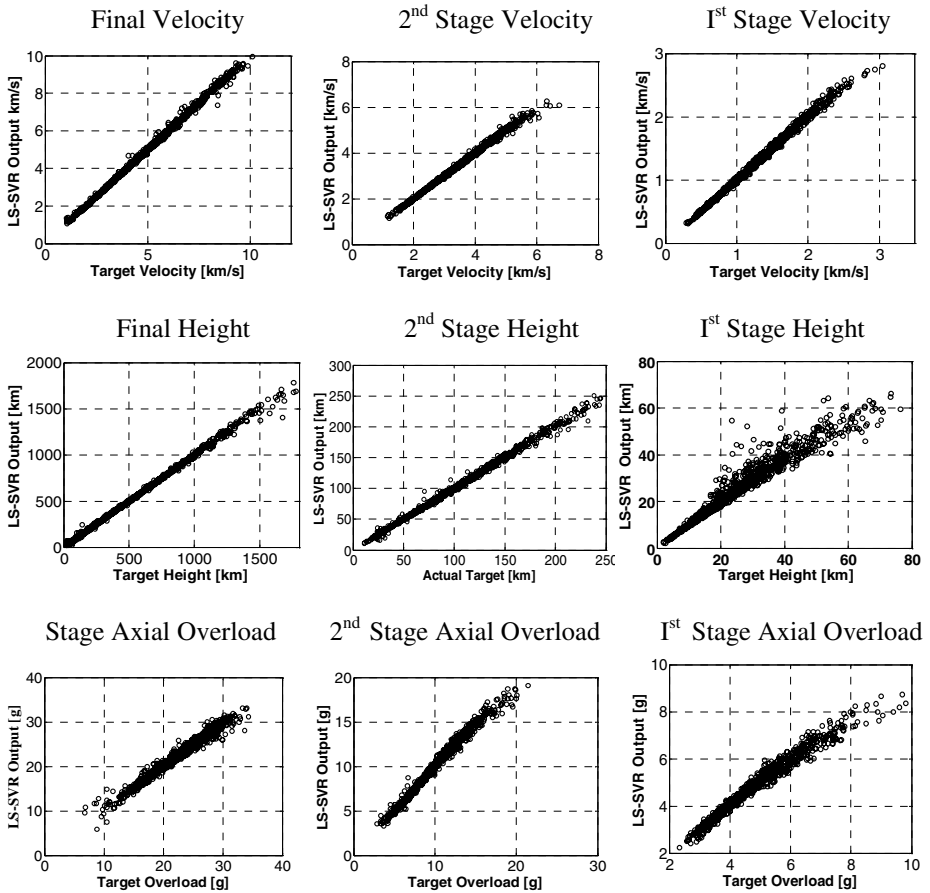


Fig. 3. Training performance of LS-SVM

Two algorithms architecture i.e. Particle Swarm Optimization in hybrid mode was Executed five times with exact function count and trajectory metamodel to find performance index for conceptual design problem of SLV. The best feasible fitness value is the lowest fitness ever encountered that does not violate the constraint. The total number of function counts required to reach the stopping criterion was also computed. Both of these output values were used to compare the performance of the computational algorithm. Most significant contribution is the drastic reduction in number of exact analysis required for convergence by using trajectory metamodel based on Support Vector Machine. The proposed scheme is especially beneficial for resource hungry expensive engineering analysis required for optimization. The number of function counts needed to converge reflected the performance of these two architectures shown below. Obviously metamodel architecture was observed computationally more attractive choice than exact analysis architecture.

Our analysis shows gradient methods are capable of exploring a continuous region, While also producing a relatively superior result providing initial guess from PSO.

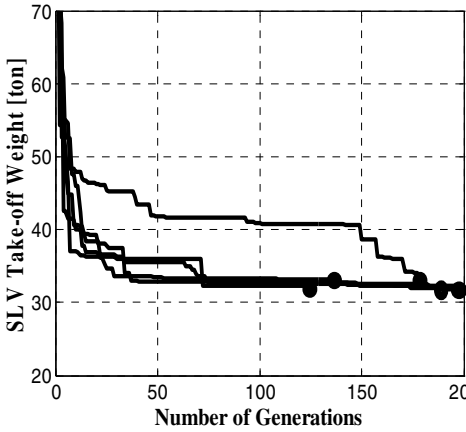


Fig. 4. Evolution of PSO (Metamodel)

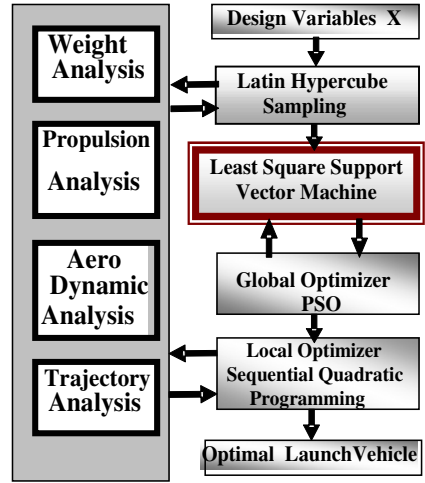


Fig. 5. Objective Function computation process (Metamodel)

Analysis results given in Table 1 and Table 3 reflects the advantage of gradient method in terms of 10% decrease in take-off mass for exact analysis case and 16% decrease in case trajectory metamodel case. The most significant achievement gained through Support Vector Regression is nearly 72% reduction in exact analysis function count during SLV conceptual design. Given this performance advantage, a hybrid approach is recommended for such multidisciplinary optimization problem. PSO can be used for global space exploration while gradient based methods can be applied in parallel to fine-tune local solutions. The proposed scheme is especially beneficial for complex, nonlinear and expensive engineering analysis required for optimization.

Table 4. Optimum values/performance of Design Variables

Design Variables	STAGE I	STAGE II	STAGE III
Chamber pressure, p_c [bar]	57.32	42.73	37.91
Nozzle exit pressure, P_e [bar]	0.611	0.454	0.173
Stage diameter, d [m]	1.53	1.30	1.00
Fine ness ratio, λ	5.413	3.477	3.20
Throat diameter, d_{th} [m]	0.2766	0.1839	0.121
Maximum angle of attack , α_m [deg]	5.95	-	-
Flight program design factor, a	0.113	-	-
Mass flow rate [kg/s]	188	62	24
Specific Impulse [s]	275	309	324
Burn out time [s]	78	87	103

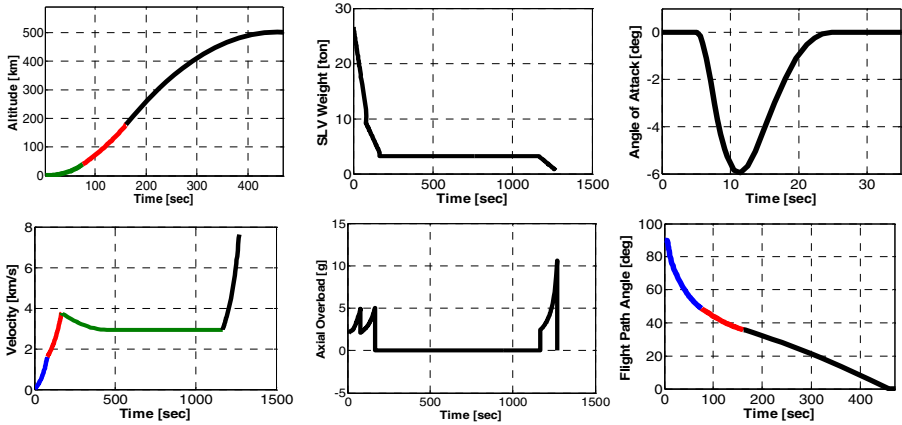


Fig. 6. Trajectory Simulation with Optimum Design Parameters

7 Conclusions

In this research the trajectory metamodel based on LS-Support Vector Machine has been successfully developed for rapid optimization and conceptual design for a liquid fueled Space Launch Vehicle. The most significant achievement gained through this analysis is nearly 72% reduction in computational cost by using trajectory metamodel during conceptual design level. In this work global search potential of stochastic direct search method PSO and its hybrid architecture with SQP has also been demonstrated for the conceptual design of liquid fueled SLV.

References

1. Martin, J., Alonso, J.J.: Complete Configuration Aero-Structural Optimization Using a Coupled Sensitivity Analysis Method. 9th AIAA/ISSMO, 2002-5402 Atlanta, GA (2002)
2. Tsuchiya, T., Mori, T.: Optimal Conceptual Design of Two-Stage Reusable Rocket Vehicles including Trajectory Optimization. *Journal of Spacecrafts and Rockets* vol.41 (2004)
3. Parsopoulos, K.E., Vrahatis, M.N.: Particle Swarm Optimization Method for Constrained Optimization Problems. University of Patras AI Research Center, GR-26110 Patras Greece
4. Akhtar, S., Linshu, H., Qazi, M.: Comparison of Pattern Search and Genetic Algorithm for Conceptual Design of Multi- Stage Liquid Rocket Missile. *IFSA* (2005) 1353-1359
5. Venter, G., Sobieszcanski-Sobieski, J.: Multidisciplinary Optimization of a Transport Aircraft Wing using Particle Swarm Optimization. *AIAA* 2002-5644
6. Boggs, P. T., Tolle, J.W.: Sequential quadratic programming. *Acta Numerica* (1995)
7. Johan, A.K., Suykens, Gestel, T.V., Brabanter, J.D., Moor, B.D., Vandewalle, J.: Least Squares Support Vector Machines. World Scientific (2002)

Transductive Support Vector Machines Using Simulated Annealing

Fan Sun and Maosong Sun

State Key Laboratory of Intelligent Technology and Systems,
Department of Computer Science & Technology,
Tsinghua University, Beijing 100084, China
sunfan_00@mails.tsinghua.edu.cn

Abstract. Transductive inference estimates classification function at samples within the test data using information from both the training and the test data set. In this paper, a new algorithm of transductive support vector machine is proposed to improve Joachims' transductive SVM to handle various data distributions. Simulated annealing heuristic is used to solve the combinatorial optimization problem of TSVM, in order to avoid the problems of having to estimate the ratio of positive/negative samples and local optimum. The experimental result shows that TSVM-SA algorithm outperforms Joachims' TSVM, especially when there is a significant deviation between the distribution of training and test data.

1 Introduction

Support vector machines (SVM) have been founded by Vapnik [8] and are gaining popularity due to many attractive features and promising empirical performance [2][8]. Most current works on SVM try to optimize the classification performance over all possible future test data, which is called inductive inference. Note that in most cases, it is neither possible nor necessary to classify all the possible future samples. In fact, only a particular set of the test data is of real interest. This observation leads to the concept of transductive inference [8]. By explicitly including the test data in problem formulation, a better generalization can be expected on problems with insufficient labeled samples [3].

Transductive support vector machine (TSVM) is a well-known algorithm that realizes transductive learning in the field of support vector classification [5]. TSVM often achieves a high performance using a small number of labeled samples and a large number of unlabeled samples. However, in TSVM, the ratio of positive and negative samples in unlabeled data requires to be manually determined, which is often difficult to estimate. Accordingly, the possibility of performance decreasing remains when there is a significant deviation between the positive/negative ratio and its estimation value. In addition, the local search strategy limits the generalization performance of TSVM.

This paper tries to find a more practical transductive inference solution for support vector classification applications. An algorithm combining transductive support vector machine with simulated annealing (TSVM-SA) is designed to implement

effective transductive inference in support vector learning. The idea of simulated annealing makes TSVM-SA algorithm effectively overcomes the shortages of the original TSVM approach and achieves better generalization performance. In TSVM-SA, the maximum margin width over labeled and unlabeled data is gradually achieved in the process of simulated annealing.

The remainder of this paper is organized as follows. Section 2 addresses the concept and importance of transductive inference, together with the review of a conventional transductive support vector machine provided by Joachims. In Section 3, the TSVM-SA algorithm is detailed out. Section 4 analyzes the performance of TSVM-SA with the experimental results in detail, and indicates some further possible improvement. Finally, in Section 5, we summarize and conclude our works.

2 Transductive Inference and TSVM

Currently, the most important work of transductive inference in the field of support vector learning is Joachims’ transductive support vector machine (TSVM), which will be briefly discussed in the remainder of this section. Detailed descriptions and proofs can be found in [5].

Given a set of independent, identically distributed labeled samples

$$(x_1, y_1), \dots, (x_l, y_l), \quad x_i \in R^n, \quad y_i \in \{-1, +1\} \tag{1}$$

and another set of unlabeled samples from the same distribution,

$$x_1^*, x_2^*, \dots, x_k^* \tag{2}$$

in a general linearly non-separable data case, the learning process of transductive SVM can be formulated as the following optimization problem:

$$\begin{aligned} \min \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i + C^* \sum_{j=1}^k \xi_j^* \\ \text{s.t.} \quad & \forall_{i=1}^l : y_i [w \cdot x_i + b] \geq 1 - \xi_i \\ & \forall_{j=1}^k : y_j^* [w \cdot x_j^* + b] \geq 1 - \xi_j^* \\ & \forall_{i=1}^l : \xi_i \geq 0 \\ & \forall_{j=1}^k : \xi_j^* \geq 0 \end{aligned} \tag{3}$$

where C and C* are user-specified parameters. C* is called the “effect factor” of the unlabeled samples and C*ξ_j^{*} is called the “effect term” of the *i*th unlabeled sample in the objective function of Equation (3).

Training the TSVM amounts to solving process of the above optimization problem. Firstly, all of the unlabeled samples are given temporary category labels by the SVM classifier trained using only the labeled data. The classifiers, after that, are iteratively constructed by focusing only on the temporary labeled samples. If a pair of positive and negative samples near the classification boundary could be found such that an exchange

of their temporary labels decreases the objective function value in Equation (3), their category labels will be exchanged and the classifier retrained. The operations of exchanging labels and retraining classifiers are repeated until there is no pair of test samples for which labels have to be exchanged, so that a separating hyperplane fitting the distribution of the test data is finally obtained.

Transductive SVM can achieve better performance than inductive learning because it successfully takes into account the distribution information implicitly embodied in the large quantity of the unlabeled samples. However, TSVM has its drawbacks. The fraction of the positive samples in the labeled data is used to estimate the number of positive samples in the unlabeled ones. This method may lead to a fairly large estimating error, especially when the number of the labeled samples is small. When the estimated N has a big deviation from the actual number of the positive samples, the performance of the learner may decrease significantly. Moreover, the local search method in TSVM generally only finds a local optimum instead of the global optimum. These drawbacks limit the usage and the extension of TSVM in practice.

3 Transductive SVM Using Simulated Annealing

Since the category labels of the unlabeled data are unknown, there are 2^k possible assignments of these labels, so that the task of transductive SVM is a combinatorial optimization problem, while the object is to find the maximum margin width in all the labeled and unlabeled data. In our works, simulated annealing heuristic is used to solve this problem.

Simulated annealing (SA) [6] is a generic probabilistic meta-algorithm for the global optimization problem, namely locating a good approximation to the global optimum of a given function in a large search space. Directed by the Metropolis criteria [7] in the searching process, an uphill stop may be accepted, so as to escape from local optima. For any given finite problem, the probability that the simulated annealing algorithm terminates with the global optimal solution approaches 1 as the annealing schedule is extended.

3.1 Objective Function and Solution Representation

The object of transductive SVM is to maximize the margin width in all the labeled and unlabeled data, which can be represented in $2/\|w\|$. In our task, the objective function is set as $\|w\|$, while the task is to minimize it. To get the value of objective function, the SVM classifier must be retrained each time.

The solution of transductive SVM is a sequence of category labels of unlabeled data, which could be represented in a bit array. At the beginning of the simulated annealing process, we get an initial solution predicted by the support vector classifier which is trained only from the labeled data. In the annealing process, the solution is gradually improved and finally optimized.

3.2 State Transition

In each transition, the previous solution is perturbed and the learning machine is retrained. Unlabeled samples are predicted by the new SVM and then a new solution generated. If the new solution has a better evaluation than the previous one, it is

accepted immediately. Otherwise, the new solution is accepted with the Boltzmann probability.

Our solution perturbation strategy is as follows. Since the initial solution is an acceptable solution with acceptable performance, we only have to adjust the separating hyperplane slightly to find a better solution. In our implementation, the q -nearest samples to the separating hyperplane are picked out, since they have the most probabilities to affect the shifting of separating hyperplane. Category label of each unlabeled sample is changed with this probability

$$\Pr_i^{ptb} = \frac{\log D_i}{\log D_0} \quad (4)$$

where D_i and D_0 denote the Euclidian distance from the i -th nearest unlabeled sample vector and the nearest one to the separating hyperplane, respectively. Parameter q is often set as a small integer (i.e. 5 or 10).

3.3 Annealing Schedule

The annealing schedule determines the degree of uphill movement permitted during the search and is thus critical to the algorithm's performance. The principle underlying the choice of a suitable annealing schedule is easily stated: the initial temperature should be high enough to "melt" the system completely and should be reduced towards its "freezing point" as the search progresses. The following parameters should therefore be specified.

3.3.1 Initial Temperature

A suitable initial temperature T_0 is one that results in an average increase of acceptance probability p_0 of about 0.8. In other words, there is an 80% chance that a change which increases the objective function will be accepted. The value of T_0 will clearly depend on the scaling of objective function f .

3.3.2 Stopping Criterion

Simulated annealing search can be halted when it ceases to make progress. The degree of progress is defined as the acceptance ratio at a certain temperature. When it falls below a given small value ε_p (i.e. 0.01), the annealing process is considered as lack of progress and can be halted.

3.3.3 Iteration Time

In the settings of simulated annealing, iteration time at each temperature is a value that depends on the size of the problem. In order to guarantee that the solution is sufficiently perturbed, we set the iteration time as k/q , which is proved acceptable. Here k denotes the number of unlabeled samples.

3.3.4 Decrementing the Temperature

After a certain times of iteration the temperature should be decremented as "cooling". Here we choose a simple and common decrementing rule:

$$T_{i+1} = \alpha T_i \quad (5)$$

where α is a constant close to, but smaller than, 1.

4 Experimental Results

4.1 Experimental Settings

We implement TSVM-SA based on Chang and Lin’s LIBSVM, which is a popular shareware [1]. Three kernel functions are tested in the experiments, that is, linear kernel, polynomial kernel and Gaussian radial basis function kernel, and all of them yield similar results. The following results are generated by the linear kernel function.

Our experiments were conducted using the Reuters-21578 corpus, which contains 21,578 articles collected from Reuters’ newswire in 1987. We perform a binary text categorization task for simplicity. Documents are represented as feature vectors, and each feature corresponds to a word stem. All the following experiments use a common labeled training data set, including 5 positive and 5 negative samples classified a priori. The unlabeled samples are selected with different sizes and different positive-negative ratios to test the performance of the algorithm on various data distributions. A common test data set composed of 300 positive and 300 negative samples is used in all the test jobs. Test accuracy is used to evaluate performance of the learning machine.

4.2 Comparison Between Different Sizes of Data Sets

Intuitively, the more unlabeled samples in training, the smaller width of margin band could be achieved. Figure 1 shows solution evaluations in the annealing process on three different data sets, in which the numbers of positive and negative samples are equal. The solution evaluation is represented in $1/w$, inverse proportion of the margin width (Section 3.1). From Figure 1 we can see that larger data set leads to a larger solution evaluation and slower convergence.

Figure 2 shows the test accuracy of the SVMs generated in the annealing process on the three data sets above. It can be shown that the biggest data set, “300 vs. 300”, leads to a better performance than the other two. This is because unlabeled samples provide extra information in transductive learning. The more unlabeled samples used in training, the better they can reflect the actual distribution of the whole data set, thus the better classifier yielded by the learning.

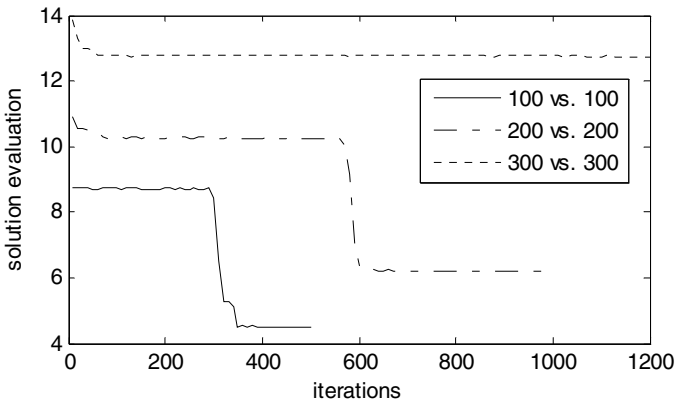


Fig. 1. Solution evaluations in simulated annealing processes on three different data sets

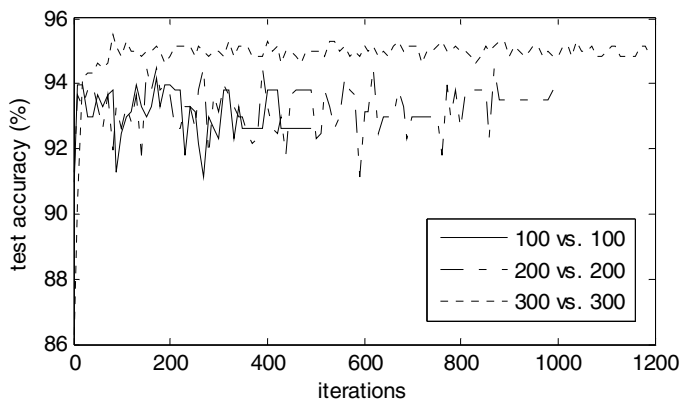


Fig. 2. Test accuracies in simulated annealing processes on the same data sets as Fig. 1

4.3 Comparison Between TSVM and TSVM-SA

Results of six different experiments are given in Table 1, in which the numbers of positive and negative unlabeled samples are equal. The first training data set contains no unlabeled samples. That is, for this training data, both TSVM and TSVM-SA perform conventional inductive learning, so the learning process and the result are exactly the same. Table 1 shows that both TSVM and TSVM-SA result in better performance compared to the original inductive learning on the other five training data sets that contain unlabeled samples of different sizes.

When comparing TSVM and TSVM-SA on the same training data set, we can see that when the actual number of positive and negative samples is equal, TSVM gives an approximately equal or even better test result than TSVM-SA because positive estimation is done correctly. However, it is not a general case. Six other experiments on training data of different numbers and positive-negative ratios of unlabeled samples are shown in Table 2. We can see that when there is a big difference between

Table 1. Results comparison of TSVM and TSVM-SA on 6 balanced datasets

Unlabeled samples	Learning algorithm	Test errors	Accuracy (%)	SV number
Pos=0	TSVM	165	72.50	10
Neg=0	TSVM-SA	165	72.50	10
Pos=10	TSVM	114	81.00	28
Neg=10	TSVM-SA	109	81.83	29
Pos=50	TSVM	62	89.67	105
Neg=50	TSVM-SA	73	87.83	109
Pos=100	TSVM	49	91.83	154
Neg=100	TSVM-SA	44	92.67	161
Pos=200	TSVM	37	93.83	258
Neg=200	TSVM-SA	37	93.83	262
Pos=300	TSVM	22	96.33	365
Neg=300	TSVM-SA	29	95.17	359

Table 2. Results comparison of TSVM and TSVM-SA on 6 unbalanced datasets

Unlabeled samples	Learning algorithm	Test errors	Accuracy (%)	SV number
Pos=10	TSVM	181	69.83	90
Neg=100	TSVM-SA	154	74.33	102
Pos=100	TSVM	151	74.83	109
Neg=10	TSVM-SA	158	73.67	114
Pos=50	TSVM	132	78.00	231
Neg=300	TSVM-SA	119	80.17	252
Pos=300	TSVM	132	78.00	284
Neg=50	TSVM-SA	123	79.50	282
Pos=100	TSVM	103	82.83	271
Neg=300	TSVM-SA	71	88.17	266
Pos=300	TSVM	108	82.00	293
Neg=100	TSVM-SA	103	82.83	285

the number of positive and negative samples, TSVM-SA outperforms TSVM because of its ability of automatic adjusting and global optimization. Except for the cases where the numbers of positive and negative samples are approximately equal in the training data, training results of TSVM-SA are always better approximations to the actual situations.

4.4 Further Discussions

Although simulated annealing presents an optimization technique that statistically guarantees finding an optimal solution, it is not without its critics. One negative feature of simulated annealing is that it is deemed difficult to fine tune to specific problems. Moreover, it is always time-consuming to find an optimal solution. While the size of data sets growing larger, execution time of this algorithm might become unacceptable. To achieve faster convergence and better performance, some advanced simulated annealing techniques could be taken into consideration, i.e. fast annealing or adaptive simulated annealing [4].

5 Conclusions

TSVM-SA algorithm is presented in this paper as a new attempt in the field of transductive inference and support vector machine. Compared with TSVM, TSVM-SA can automatically adapt to different data distributions and realize a transductive learning of support vectors in a more general sense.

Still many open questions are left regarding transductive inference and support vector learning. How well does the algorithm presented here approximate the global solution? How is a proper transductive inference algorithm for data sets with different features chosen? How could TSVM-SA be improved with different approached to work with different data distribution? All the questions are very interesting and each deserves further effort.

Acknowledgements

The research is supported by the National Natural Science Foundation of China under grant number 60321002, the Tsinghua-ALVIS Project co-sponsored by the National Natural Science Foundation of China under grant number 60520130299 and EU FP6, and the National 863 Project of China under grant number 2001AA114210-03.

References

1. Chang, C.C., Lin, C.J., 2001. LIBSVM: a library for support vector machines. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
2. Cortes, C., Vapnik, V., 1995. Support vector networks. *Mach. Learn.* 20, 273–297.
3. Gammerman, A., Vapnik, V., Vowk, V., 1998. Learning by transduction. In: *Conference on Uncertainty in Artificial Intelligence*, pp. 148–156.
4. Ingber, L., 1993. Simulated annealing: Practice versus theory, *Mathl. Comput. Modelling* 18, 11, 29-57.
5. Joachims, T., 1999. Transductive inference for text classification using support vector machines. In: *International Conference on Machine Learning (ICML)*, 1999, pp. 200–209.
6. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by Simulated Annealing, *Science*, 220, 4598, 671-680.
7. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E., 1953. Equation of State Calculations by Fast Computing Machines, *J. Chem. Phys.*, 21, 6, 1087-1092.
8. Vapnik, V., 1998. *Statistical Learning Theory*. Wiley.

Input Selection for Support Vector Machines Using Genetic Algorithms

Hee-Jun Song^{1,*}, Seon-Gu Lee^{1,**}, and Sung-Hoe Huh^{2,**}

¹ Department of Electrical Engineering, Korea University, Seoul, Korea
{nyaong7, cr98102}@korea.ac.kr

² Dept. of Environmental Science & Engineering, Ewha Woman's Univ., Seoul, Korea
sungh@korea.ac.kr

Abstract. In this paper, an effective and simple method of input selection for nonlinear regression modeling using Support Vector Machine combined with Genetic Algorithm is proposed. Genetic Algorithm is used in order to extract dominant inputs from a large number of potential inputs in input selection process. Support Vector Machine is used as a nonlinear regressor with the selected dominant inputs. The proposed method is applied to the Box-Jenkins furnace benchmark to verify its effectiveness.

1 Introduction

Real-world modeling processes generally require a large number of inputs to represent complex dynamic systems. A large number of inputs obviously increases the complexity in computation and causes other problems such as the lack of memory spaces etc. Especially, in case that a model possesses high nonlinearity or contains many parameters, those problems become more serious. Therefore, it is necessary to adequately reduce the number of inputs. The main purpose of input selection is reducing the complexity in modeling process by removing redundant inputs.

For the purpose, the dominant inputs which significantly affect the system output are extracted by Genetic Algorithms (GAs) [3], [4] in precedence of being applied to SVMs. GAs is powerful search algorithm in finding the optimal solution.

The foundation of Support Vector Machine (SVM) has recently been developed by Vapnik et al. [1],[2]. SVM is a powerful method for solving classification and regression problems. SVM is useful in building the best model for both linear and nonlinear problems as a statistical measure.

Considering the functions of GAs and SVM, we take into account of the combination of SVM and GAs in order to effectively deal with nonlinear regression tasks. For verification, the proposed method combined GAs and SVM is applied to the Box-Jenkins gas furnace data [5], which is a nonlinear regression modeling problem. The simulation results are presented to demonstrate the effectiveness of the proposed method.

This paper is organized as follows. In Section 2, the basic of SVM is introduced. GAs is briefly introduced in Section 3. In Section 4, input selection process in nonlinear regression modeling is described. Experiment results are given in Section 5. Section 6 concludes the paper.

* Co-first author.

** Corresponding author.

2 Support Vector Regression (SVR)

Support Vector Machine (SVM) is an artificial intelligent algorithm well known for its high performance and fast operational speed in classification and regression problems. For regression problems, Support Vector Regression (SVR) is applied. Fig. 1 shows the brief description of SVR operation. Data samples are located in a space, and support vectors (most influencing data samples) which can classify the dataset the best are selected. A margin support vectors S , error support vectors E , error support vectors star E^* and remaining vectors R in the Fig.1 are constructed in the compact metric feature space by minimizing the margin denoted by ε using Lagrange. For nonlinear problems, methods of defining non-linear shape kernels are used, called 'kernel trick.' SVM has several advantages compared to NNs. First off, SVM doesn't have local minima. Secondly, SVM doesn't need to consider whole samples in training and test so that it shows much faster performances than NNs in real applications. And since it is possible to construct a hyper-plane having a high dimension, it is extraordinarily robust on high dimensional problems than other AI algorithms.

In this paper, SVR is used to model the Box-Jenkins gas furnace problem. Through these procedures as above, SVR functions to find the optimal regression model by finding the minimal margin. In our research, a non linear SVR was used to model the Box and Jenkins data.

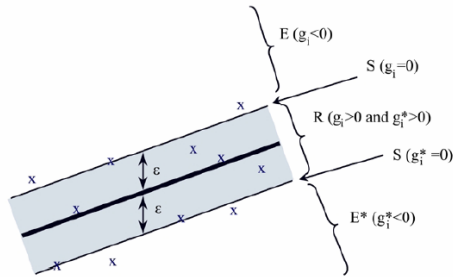


Fig. 1. Support Vector Machine Regression

3 Genetic Algorithms

The Genetic Algorithms (GAs) is used as a dominant input selector as a pre-process of SVR modeling. GAs persists in such survival evolution principles as the fittest to live, the better to be accepted and the worse to be eliminated throughout natural selections. In Fig. 2, GAs searches for the optimal solution by automatically selecting the best individuals through generations. The individuals coded as binary strings of finite length represent the potential solutions. The performances of individuals are evaluated by the fitness function. In the next step, the dominant individuals form a population of the next generation by simulating the natural selection processes based on artificial crossover, mutation and selection. In this procedure, the evolution phase loops until the optimal solution is found or the generation number reaches the maximum. When one of the two conditions is satisfied, the optimal solution is produced. GAs is used to

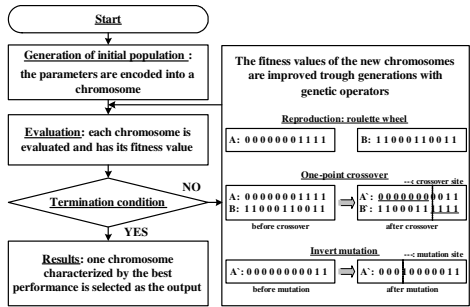


Fig. 2. Genetic Algorithm Procedure

find the dominant inputs from the Box and Jenkins data and the dominant input data-set is applied to SVR to generate the final model.

4 Input Selection for Nonlinear Regression Model

In order to eliminate irrelevant inputs and find dominant inputs which influence independently and significantly the system output, the GAs is employed with many candidate inputs with SVR model and each MSE is calculated as the GAs' fittest function to evaluate each individual.

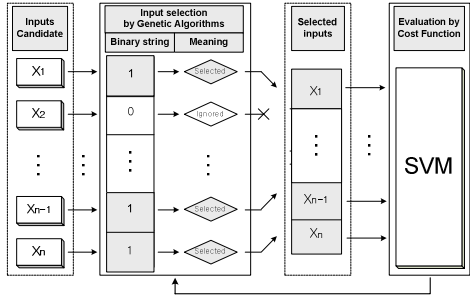


Fig. 3. Input Selection

After evaluations of all generation finished, the GAs enables the system to have reduced input dataset by eliminating irrelevant inputs. Finally, the selected input set with the smallest MSE is applied to the object function, SVR to generate the final model. This procedure is depicted in Fig. 4.

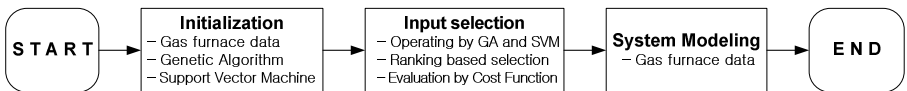


Fig. 4. Overall block diagram of proposed method

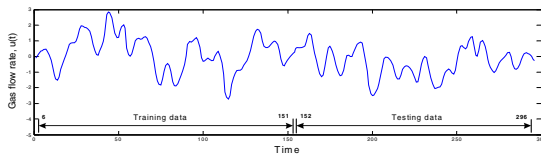
In Fig. 4, the initialization part shows the determining process of the initial conditions for SVR and GAs. Upper bound and kernel function of SVR and population size, generation number, etc. of GAs are set at this step. Next in the input selection part, the best input set extracted by GAs is applied to the fittest function. The finally generated dominant input dataset with the lowest MSE is applied to the object function, the complete version of SVR, in the system modeling part.

5 Experiment Results

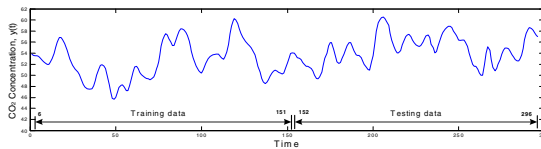
In order to verify the effectiveness, the proposed method is applied and tested with Box and Jenkins gas furnace data. Box and Jenkins gas furnace data is a time series data which possibly has many inputs by the combination of time step, such as $y(t-1)$, $y(t-2)$, $y(t-3)$, and $u(t-1)$, $u(t-2)$, $u(t-3)$, etc, where, $u(t)$ is the gas flow rate and $y(t)$ is the CO2 concentration. Box and Jenkins data is frequently used as a benchmark example to test nonlinear identification problems.

The data consists of 296 pairs of $u(t)$ and $y(t)$ from $t=1$ to 296 in total. To be realistic, the GAs limit the candidate input have a range of time steps from t to $t-4$ for $u(t)$, from $t-1$ to $t-5$ for $y(t)$ so the potential inputs may have up to ten input candidates. Therefore, excluding the first 5 data samples in $u(t)$ and $y(t)$, 291 data sets are extracted and the first 146 data sets are used as training dataset and the remaining 145 data sets are used as test dataset. Training and test phase are shown in Fig. 5.

As previously mentioned, we reduced the input sets among ten input candidates by dominant input selection using GAs with parameters of maximum epoch, 160, desired



(a) Box and Jenkins gas flow rate, $u(t)$



(b) Box and Jenkins gas CO2, $y(t)$

Fig. 5. Box –Jenkins gas furnace data

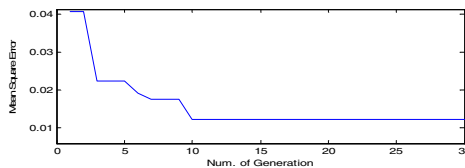


Fig. 6. MSE Curve

MSE, 0.01, population size, 10 and maximum generation, 30. After the dominant input selected, the best input set which has the smallest training MSE is found, which is $[u(t), u(t-3), y(t-1), y(t-4)]$. Then this set is applied to the object function, SVR, to identify the nonlinear regression problem.

Fig. 6 shows the minimum MSE at each generation. SVR training result denoted as $\bar{y}(k)$ and its error are shown in Fig. 7(a) and (b). Fig. 8 shows the result in Fig. 8(a)

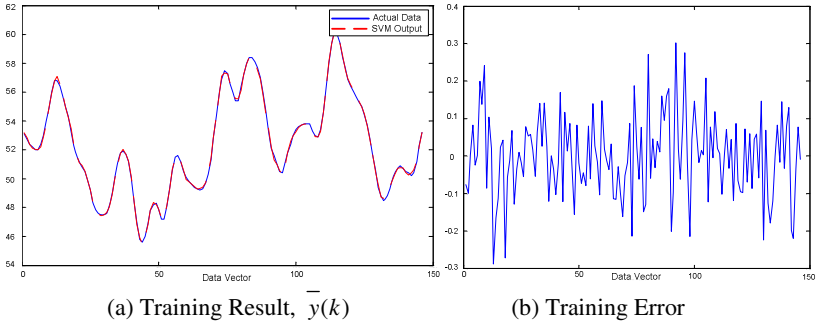


Fig. 7. Training Result

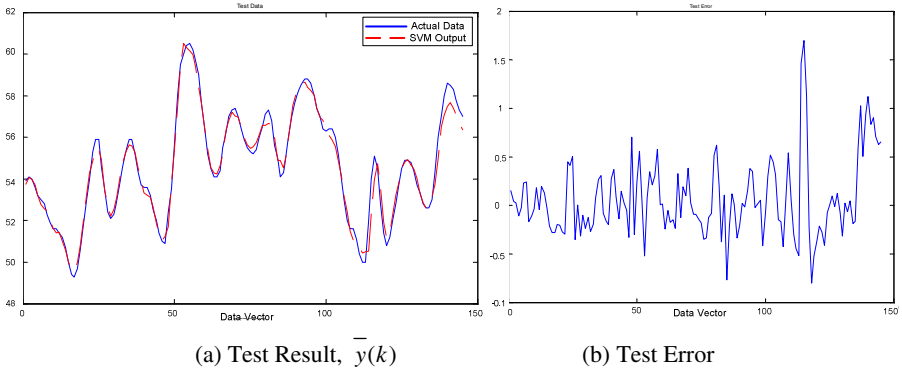


Fig. 8. Test Result

Table 1. Comparison with Other Methods

Models	Inputs	Mean Square Error	
		Training	Test
Tong [6]	$u(t-4), y(t-1)$	0.469	
Xu [7]	$u(t-4), y(t-1)$	0.328	
Pedrycz [8]	$u(t-4), y(t-1)$	0.320	
Lin [9]	$u(t-3), u(t-5), (t-6), y(t-1), y(t-2)$	0.071	0.261
Sugeno [10]	$u(t-1), u(t-2), (t-3), y(t-1), y(t-2), (t-3)$	0.068	
Proposed Model	$u(t), u(t-3), y(t-1), y(t-4)$	0.0123	0.1615

and its test error in Fig. 8(b). In Fig. 7(a) and 8(a), the solid and the dotted line mean the actual data and the estimated data, respectively. From the results in Fig. 7(b) and Fig. 8(b), the training and test MSEs of the proposed model are 0.0123 and 0.1615, respectively. In Table 1, the proposed method is compared with other methods identifying the same data. It can be seen from Table 1 that the performance of our model is better than other models.

6 Conclusion

In this paper, an effective method of input selection by Genetic Algorithm for nonlinear regression models for the use of Support Vector Machine is proposed. For verification, the proposed model is applied to Box and Jenkins gas furnace problem. Our method shows better results so that it is concluded to possess the higher availability than other models. In addition, it is expected that our proposed method might be successfully used to resolve the computation and memory space problems by reducing the number of inputs without losing the accuracy in applying SVM or SVR to real problems having high nonlinearities. Therefore, the proposed method is highly available when it is used in real problems.

References

1. V. Vapnik: *The Nature of Statistical Learning Theory*, Springer, (1995)
2. C. Cortes, V. Vapnik: *Support Vector Networks*. *Machine Learning*, Vol. 20. (1995) 273-297
3. D. E. Goldberg: *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, MA (1989)
4. J. Holland: *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor, M.I. (1975)
5. G. Box, G. Jenkins: *Time Series Analysis: Forecasting and Control*, Holden Day, San Francisco (1970) 532-533
6. R. M. Tong: *The Construction and Evaluation of Fuzzy Models*. In *Advances in Fuzzy Set Theory and Applications*, Eds. Amsterdam: North-Holland, (1979) 559-576
7. C. W. Xu, Y. Z. Lu: *Fuzzy Model Identification and Self-Learning for Dynamic Systems*. *IEEE Trans. Syst., Man, Cybern.*, Vol. 17, (1987) 683-689
8. W. Pedrycz: *An Identification Algorithm in Fuzzy Relational System*. *Fuzzy Sets Systems*, Vol. 13, (1984) 153-167
9. Y. Lin, G. A. Cunningham III.: *A New Approach to Fuzzy-Neural System Modeling*, *IEEE Transactions on Fuzzy Systems*, 3 (2), (1995) 190-198
10. M. Sugeno, K. Tanaka: *Successive Identification of a Fuzzy Model and Its Application to Prediction of a Complex System*, *Fuzzy Sets and Systems*, 42, (1991) 315-334

Associating k NN and SVM for Higher Classification Accuracy

Che-Chang Hsu², Chan-Yun Yang¹, and Jr-Syu Yang²

¹ Department of Mechanical Engineering, Northern Taiwan Institute of Science and Technology, No. 2 Xue Yuan Rd., Beitou, Taipei, Taiwan, China
cy.yang@ntist.edu.tw

² Department of Mechanical and Electro-Mechanical Engineering, Tamkang University Taipei, Taiwan, China
692342792@es92.tku.edu.tw, 096034@mail.tku.edu.tw

Abstract. The paper proposed a hybrid two-stage method of support vector machines (SVM) to increase its performance in classification accuracy. In this model, a filtering stage of the k nearest neighbor (k NN) rule was employed to collect information from training observations and re-evaluate balance weights for the observations based on their influences. The balance weights changed the policy of the discrete class label. A novel idea of real-valued class labels for transferring the balance weights was therefore proposed. Embedded in the class label, the weights given as the penalties of the uncertain outliers in the classification were considered in the quadratic programming of SVM, and produced a different hyperplane with higher accuracy. The adoption of k NN rule in the filtering stage has the advantage to distinguish the uncertain outliers in an independent way. The results showed that the classification accuracy of the hybrid model was higher than that of the classical SVM.

1 Introduction

In the past years, a powerful learning machine of support vector machines (SVMs) has received much attention. From the beginning, the conceptual mathematics has been founded by Vapnik [1] based on the theoretical learning theory. The basic concept of the theory is sought to design a classification rule for an optimal function which is obtained by the minimization of the generalization error.

Considering a binary classification problem, a set of classifiers regularized by the relevant parameters is generated to minimize the expected risk over all available and unavailable training samples. In general, the expected risk comes up with unknown probability density unfortunately. An approximation thus is usually adopted by replacing the expected risk with an empirical risk. The empirical risk, empirically measured from the loss function, has the advantage that we can easily and readily compute only from the available training samples. From the point of loss function, if a hard sample located distant from its native class, it increases the opportunity of misclassification, and also increases its loss in the empirical risk.

In the theory of statistical learning, the loss function analogous to the residual in the regression plays a key role in the classification. As the rule of the loss function,

samples with positive margin are known as those classified correctly. In the contrast, samples with negative margin are those misclassified. According to the definition, the goal of the learning is to produce positive margin as frequently as possible. The loss function used in classification should penalize a sample with negative margin more heavily than a sample with positive one. Several typical surrogates of loss function, including the 0-1, the exponential, the binomial deviance, the squared error, and the hinge loss functions are often used in the statistical learning. The common essential property of them is to continuously penalize the samples with negative margin. The differences among these surrogates are the degree of the penalization exerted on the samples with negative margin. However, from the spirit of the surrogates, we proposed a surrogate of loss function with various penalties. The surrogate still fulfills the criterion of increasing penalization of those samples with negative margin, but the degree of the penalization is depended on the class belonging of the nearest neighbors.

The k nearest neighbors (k NN) rules, mining locally the useful information among the training samples, give a special independent aspect to evaluate the samples' class belonging. With the evaluation, various penalties are then determined in the loss function. The loss function with various penalties takes effect in the optimization procedure will produce a set of new Lagrangian multipliers and form a separating hyper-plane different from the set of original multipliers.

2 A Hybrid Model of k NN and SVM

2.1 k -Nearest Neighbor Rules

The k NN methods [2] are well-known decision-making methods that are widely used in the field of pattern recognition. The basic of such an algorithm is starting with a set of n training samples, $S = \{(\mathbf{x}_i, y_i)\}$, $i=1, 2, \dots, n$, $\mathbf{x}_i = [x_i^1, x_i^2, \dots, x_i^d]^T \in R^d$, called the prototypes drawn in the feature space R^d . The class belonging of a new arrival query point is determined locally by the k nearest neighbors of the query point. It is an intuitive method that classifies unlabeled examples based on the similarity to the neighbors in the prototypes. The k NN method does not matter whatever the distribution form or global probability density of the training data, and therefore is characterized as a nonparametric method. Due to the basic idea of k NN rule, some variations have been proposed to improve its performance. Two important variations sought to give different weights from k nearest neighbors and were stated as a milestone in the field. The first one is the distance-weighted k NN (DW- k NN) rule. Instead of the implicit equal weights of the k neighbors assumed in the voting k NN rule, the DW- k NN rule intuitively gives different weights to the k neighbors based on their distance to the unlabeled sample [3]. The modification enhances the influence of the closer neighbors on the greater the weights. The second one is fuzzy k NN (F- k NN) rule [4]. The fuzzy k nearest neighbor algorithm assigns class membership to a sample rather than assigning the sample to a particular class. After, the memberships devote themselves as weights in the classification. From the viewpoint, the sample's membership value provides a level of assurance to accompany the resultant classification. In the study, a k NN rule will be employed to assign the weights of training samples. The

variation of k NN rule which can provide sufficient information to produce the weights of the samples is a good choice in the scheme.

2.2 Support Vector Machines with Real-Valued Class Label

The support vector machines were developed based on the structural risk minimization principle of statistical learning theory [1, 5] by learning from the training samples, the decision function therefore can be obtained. The basic form of decision function in SVM is given as $f(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle + b)$ which is described by a vector of orientation weights \mathbf{w} and a bias term b . The goal of the SVM is to find a separating hyperplane with maximal margin while the classification error can be minimized by the training samples. With the same notation in the k NN rule, a set of training samples, $S = \{(\mathbf{x}_i, y_i)\}, i=1, 2, \dots, n$, is proposed the change of input training set in the set S^* [6]:

$$S^* = \{(\mathbf{x}_i, y_i^*)\}, i = 1, 2, \dots, n. \tag{1}$$

where y_i^* , having its sign identical to y_i in the training set S , denotes a relaxed real-valued class label to represent the potential weights that sample i should be taken. The expression of S^* tries to carry more information about the training set, regardless both S and S^* contains a set of the same patterns \mathbf{x}_i .

The change of S in S^* involves a remapping regarding to the idea of assigning various weights for the samples in different situations. In (1), class label y_i^* is no longer a discrete value; instead it becomes a real value to represent an implicit relationship to the sample's native class. The value of y_i^* can be obtained from a modification of k NN rule, such as

$$y_i^* = \eta \cdot y_i \cdot \left(\frac{k}{h + \varepsilon}\right), \tag{2}$$

where k is the value that k NN method has adopted previously, and h is a value counting the neighbors with an identical sign of class label in the k nearest neighbors. The expression of (2) adopts the strategy of voting k NN scheme to scale the value of y_i^* . In general, the ratio of k/h is greater than 1. In the expressions, we use the ratio to magnify y_i . To avoid dividing by zero, a sufficient small value of ε is added to the denominator. Parameter η in the expressions, called an acceleration factor, should be a positive real and greater than 1

$$y_i^* \geq 1, \quad \text{for } y_i = +1, \tag{3}$$

$$y_i^* \leq -1, \quad \text{for } y_i = -1. \tag{4}$$

In (3) and (4) real-valued class label y_i^* provides a stricter penalty in the optimization, and is able to conduct the classification more accurate. Especially, the improvement will be more significant in a much more confused dataset with many uncertain outliers. In the present study, the uncertain outliers are denoted as those confused samples, which are frequently classified into incorrect partition by a separating scheme. In the study, an uncertain outlier is designed to carry more additional weight in order to receive more penalties in the optimization.

With the real-valued class label y_i^* , the canonical hyperplane of SVM can be re-written as:

$$\mathbf{w}^{*T} \mathbf{x}_i + b^* \geq +1 - \xi_i^*, \quad \text{for } y_i^* \geq +1, \text{ and} \tag{5}$$

$$\mathbf{w}^{*T} \mathbf{x}_i + b^* \leq -1 + \xi_i^*, \quad \text{for } y_i^* \leq -1, \tag{6}$$

where ξ_i^* denotes the slack variable for solving an inseparable problem. By re-substituting (5)-(6) into a classical formulation of SVM, the quadratic programming problem becomes:

$$\max_{\alpha^*} L_D(\alpha^*) = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i^* y_j^* K(\mathbf{x}_i, \mathbf{x}_j) \alpha_i^* \alpha_j^* + \sum_{i=1}^n \alpha_i^*, \tag{7}$$

subject to

$$\sum_{i=1}^n \alpha_i^* y_i^* = 0, \text{ and } 0 \leq \alpha_i^* \leq C^*, \quad i = 1, 2, \dots, n. \tag{8}$$

2.3 Associating k NN with SVM

As described in Section 1, the loss function always focuses on the samples with negative margins, and tries to penalize these samples heavier. From the idea, a filtering stage of k NN rule will be inserted in front of the classification stage. From the idea, a filtering stage of k NN rule is inserted in front of the classification stage. A two-stage model (Fig. 1) is proposed in order to fit the criterion of heavy penalization. In the model, k NN rule has filtered all the possible uncertain outliers, and has transferred the set of evaluated strength of class belonging of the uncertain outliers to the classification stage. With a real-valued class label y_i^* , taken the strength as an index to reassign its value, the classification stage, organizes y_i^* in the input set of S^* to produce a new hyperplane by the set of Lagrangian multipliers α_i^* . The induced hyperplane, with the additional penalties for the uncertain outliers, tends towards higher accuracy in classification. The advantage of adopting two different methods in the both stage is they are independent to each other. The independence potentially reduces the opportunity of over-penalization if the identical method is used twice in the both stage. Since the identical method is often inclined to behave in a particular way, and is tend to focus on the same crew of training samples.

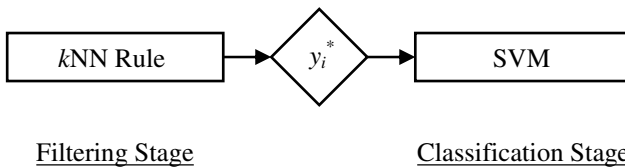


Fig. 1. Schematic of k NN-SVM

3 Experimental Results

A set of TwoNorm dataset [7] for binary classification was generated to examine the effects of k NN-SVM. A separating hyperplane of dataset TwoNorm depicted by the k NN-SVM is shown in Fig. 2b comparing with that in Fig. 2a by the classical SVM. In this example, both C^* and C in the compete models are setting at the same value of 1×10^3 . The other relevant parameter settings are as shown in the figure. In Fig. 2, the counts of the misclassified points – those marked with a number – are reduced from 17 to 4 due to the moderate change of the hyperplane. Due to the change, the hyperplane becomes more rugged. This is caused by a margin reduction form 2.61×10^{-2} to 2.47×10^{-2} in both the classical SVM and the k NN-SVM, respectively. The amount of margin reduction is as a result going to trade off the reduction of misclassification counts.

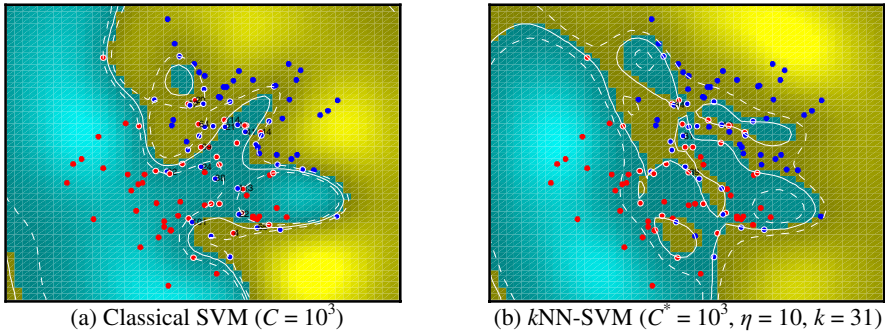


Fig. 2. A classification example of k NN-SVM

For more detailed examination, a diagram of misclassification rate varying the increasing value of acceleration factor η is given in Fig. 3a. Regardless of the different setting of C^* and C , the misclassification rates of k NN-SVM, illustrated as the solid lines, are lower than that of classical SVM, illustrated as the dashed lines, generally. It is interesting that the reduction of misclassification rate is approximately increasing with a higher value of η . The reduction improves the classification accuracy. In accompanying with the lower misclassification rate in Fig. 3a, a margin reduction can be found in Fig. 3b as the sacrifice to trade the improvement of classification accuracy.

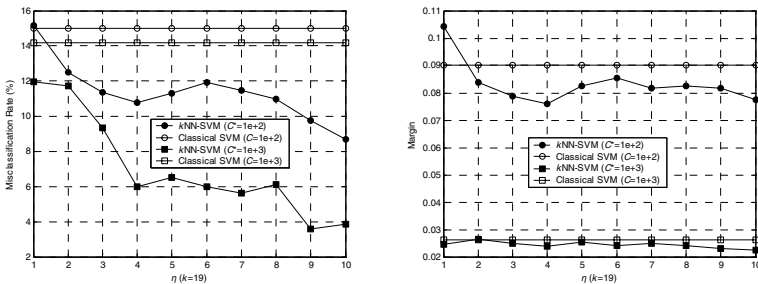


Fig. 3. The changes of misclassification rate and related margin in different setting of η

4 Conclusion

The hybrid model of k NN-SVM tending to increase the classification accuracy was developed by applying heavier penalties of the uncertain outliers. In the model, not only the relaxation of real-valued class label for transferring additional penalties is a crucial point, but also the use of k NN rule independent to the approach of SVM for filtering the uncertain outliers is a key to success. In the paper, consistent consequences from the experiments are included to support the theory. The experimental results show that the method of k NN-SVM, dealing with much more information than the training data can provided, is more accurate and robust than existing techniques. The extension of an implementation of the two-stage model, such as a SVM ensemble by an iterated or an adapted scheme, seems possible for further study. We now devote ourselves to the researches on this direction.

References

1. Vapnik, V. N.: An Overview of Statistical Learning Theory. IEEE Transactions on Neural Networks, Vol. 10 (1999) 988–999
2. Cover, T. M., Hart, P. E.: Nearest Neighbor Pattern Classification. IEEE Transactions on Information Theory, Vol. 13 (1967) 21–27
3. Dudani, S. A.: The Distance-Weighted k -Nearest-Neighbor Rule. IEEE Transactions on Systems, Man and Cybernetics, SMC-6(4) (1976) 325–327
4. Keller, J. M., Gray, M. R., Givens, J. A.: A Fuzzy k -Nearest Neighbor Algorithm. IEEE Transactions on Systems, Man and Cybernetics, SMC-15, No. 4 (1985) 580–585
5. Vapnik, V. N.: Statistical Learning Theory. Wiley, New York (1998)
6. Yang, C. -Y.: Support Vector Classifier with a Fuzzy-Value Class Label. Lecture Notes in Computer Science, Vol. 3173, Springer-Verlag, Berlin (2004) 506–511
7. Hettich, S., Bay, S. D.: The UCI KDD archive <http://kdd.ics.uci.edu>. Irvine, CA: University of California, Department of Information and Computer Science. (1999)

Multi-class SVMs Based on SOM Decoding Algorithm and Its Application in Pattern Recognition

Xiaoyan Tao and Hongbing Ji

School of Electronic Engineering, Xidian University, Xi'an 710071, China
tao_xiaoyan@hotmail.com, hbji@xidian.edu.cn

Abstract. Recently, multi-class SVMs have attracted much attention due to immense demands in real applications. Both the encoding and decoding strategies critically influence the effectiveness of the multi-class SVMs. In this work, a multi-class SVMs based on the SOM decoding algorithm is proposed. First, the binary SVM classifiers are trained according to the ECOC codes. Then the SOM network is trained with the output of the training samples and the optimum weights are obtained. Finally the unknown data is classified. By this method, the confidence of the binary classifiers is completely considered with the case avoided that the same minimum distance to several classes is obtained. The experimental results on the Yale face database demonstrate the superiority of the new algorithm over the widely-used Hamming decoding method.

1 Introduction

Recently, classification with multi-class SVMs has become an active research area because of the demands in many real world applications [1]. There are two methods for multi-class SVM [2]. The first solves a multi-class optimization problem directly based on the classical SVM theory; The second constructs a series of binary classifiers by some codes and integrates them for multi-classification. Although the first one looks simple, the number of the variables is much more than that used in the second one in solving the optimization problems with high complexity and without advantages in classification accuracy. This is much extrusive for the case of more categories. Thus the second one is dealt with in this paper due to its common use. The implementation of the second type consists of two stages: (1) Reducing a multi-class problem into several binary sub-problems by selecting a set of codes. A general scheme is called error correcting output codes (ECOC) based on the information theoretic method [3, 4], and the simplest encoding strategy is called "one-vs-all"; (2) In the test phase, an unknown sample is first mapped to a real vector formed by the outputs of the binary classifiers. A target class is then computed from this vector by means of a decoding function. The general one is the Hamming distance decoding scheme. The virtue of this method lies in its simpleness. But it ignores the confidence of every classifier, and when there are several codewords whose distance to that output code is

the minimum, the class that comes first in the class ordering is the target, influencing the classification accuracy. For this, we introduce the Self-Organizing Map [5, 6](SOM) algorithm. Based on its nonlinearity and topological ordering, an SOM-based decoding method can achieve better classification accuracy. This paper is organized as follows: In Section 2 we briefly review the theory of the binary SVM and ECOC. The SOM-based decoding method for multi-class SVM is introduced in Section 3. Section 4 presents the experimental results and performance analysis. We conclude this paper in Section 5.

2 Binary SVM and ECOC

2.1 Binary SVM

For the binary pattern recognition problem, the support vector approach has been well developed. The main idea is to construct an optimal hyperplane to separate the two classes so that the margin is maximal.

Let the training set $\{(x_i, y_i)\}_{i=1}^N$, where $x_i \in \mathbb{R}^d (i = 1, 2, \dots, N)$ is the input pattern of the i th training sample, $y_i \in \{-1, +1\}$ is the output target for each input, N is the number of the training samples. SVM gives the following optimization problem:

$$\min \quad \frac{1}{2}w \cdot w + C \sum_{i=1}^N \xi_i \tag{1}$$

$$s.t. \quad y_i(x_i \cdot w + b) \geq (1 - \xi_i)$$

$$\xi_i \geq 0, i = 1, 2, \dots, N$$

The dual problem is:

$$\max \quad \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N y_i y_j \alpha_i \alpha_j K(x_i, x_j) \tag{2}$$

$$s.t. \quad 0 \leq \alpha_i \leq C, i = 1, \dots, N$$

$$\sum_{i=1}^N \alpha_i y_i = 0$$

Giving the decision function:

$$f(x) = \text{sign}[(\sum_{i=1}^N \alpha_i y_i K(x, x_i)) + b]. \tag{3}$$

2.2 ECOC and Hamming Decoding Strategy

In ECOC, let $\Omega = \{\omega_1, \dots, \omega_c\}$ be a set of class labels. Suppose that each classifier codes the compound class $\Omega^{(1)}$ as 1 and compound class $\Omega^{(-1)}$ as -1 respectively. Then every class $\omega_j, j = 1, \dots, C$ will have a binary "profile" or a codeword. Thus a codematrix $M \in \{-1, 1\}^{(C \times S)}$ is produced for the real classification, where C is the number of the classes, S binary classifiers trained. Each class is assigned a row code denoted by m_c , called a codeword. The i th bit in every codeword constitutes a binary function \cdot . During prediction a new input x is classified by computing the vector formed by the outputs of the classifiers, $f(x) = (f_1(x), \dots, f_s(x))$ and choosing the class whose corresponding row is closest to $f(x)$. The decoding operation is denoted by:

$$\arg \min_{c=1}^C d(m_c, f(x)) \tag{4}$$

Where d is the decoding function. In [3], d is chosen to be the Hamming distance:

$$d(m_c, f) = \sum_{s=1}^S \frac{|m_{cs} - \text{sign}(f_s)|}{2} \tag{5}$$

In this equation, the output real vector $f(x)$ is transformed into a binary vector directly, which doesn't consider the magnitude, a measure of the confidence of each binary SVM, thus affecting the classification accuracy.

3 Multi-class SVMs Based on SOM Decoding Algorithm

3.1 Self-organization Map

SOM is an artificial neural network based on the unsupervised, competitive learning. It constructs a topology that the high-dimensional space is mapped onto map units in such a way that relative topology distances between input data are preserved. The map units usually form a one- or two-dimensional regular lattice. SOM has two states: training and classification. The basic idea is that when the input vector is closest to a weight vector connecting the input nerve cells and the output ones, the output nerve cell is activated as the final output. At this time, this connecting weight is corrected to be more close to the input, while the weights in neighborhood is done either until the ultimate conditions are satisfied.

3.2 Multi-class SVMs Based on SOM

To deal with the problem happened in Hamming decoding method, an SOM-based multi-class SVMs is introduced, which not only takes into account the output magnitude of the binary classifier but also keeps the topological ordering to make the mapped data whose input characters are similar is close.

The classification process of the new method is as follows:

Step1. Train each SVM binary classifier on the training set in terms of ECOC

Step2. Initialize the weights $w_j(0)$ of SOM with the small random values

Step3. Input the $f(x)$ of the training samples

Step4. At time n , select the optimum matching cell i (competition process)

$$i(x) = \arg \min_j \|x(n) - w_j\|, \quad j = 1, 2, \dots, N \quad (6)$$

Step5. Determine the neighbor function $\lambda_i(n)$ (collaboration process), and correct the weights

$$w_j(n+1) = w_j(n) + \eta(n)\lambda_{j,i}(n)[x(n) - w_j(n)] \quad (7)$$

Step6. $n \leftarrow n + 1$, return to 3 until the weights had no clear change

Step7. Input the test samples' $f(x)$ into the trained SOM network for classification.

Learning step $\eta(n)$ and neighbor function $\lambda_i(n)$ should vary with n , they are the key factors to learning efficiency.

4 Experiments

A comparison of the proposed algorithm with the Hamming decoding scheme is performed on the Yale face database. In order to avoid the possibility that a fortuitous choice of the parameter could affect the results, an extensive series of experiments for 10 different values are carried out, and LIBSVM [7] is used to train binary SVMs. The results verify the superiority of the new method in terms of the performance.

The photos in Yale Face Database come from Yale Center for Computational Vision and Control. There are 15 individuals and 11 images for each. The samples from this database are depicted in Fig.1. Before classification, a PCA is performed to extract the features from the face images for dimension decrease. 40 eigenfaces are used to form the main subspaces in two experiments and a 40-dimension vector is obtained for every image projected into this space. In order to avoid the affection of the gray levels, the datum are normalized within $[-1, 1]$. Here a linear SVM is used. And the first five images of each person for training and the remaining five for testing so the number of samples for training and testing are both 75 respectively.

At first, the classification performance and the decoding time are evaluated on three ECOC codes: OVA, BCH15, BCH31. The recognition rate is shown in Fig.2. The results indicate that the performance of the new method is better for all ECOC and most of C , and the accuracy can be improved by C adjusted. In addition, due to the SOM network needs training, the decoding time is increased.

Then, the recognition rate v.s. the number of binary SVM used is tested. The number of the base classifiers is selected from 3 to 31 based on the BCH31code. The result is depicted in Fig.3, which reveals that SOM decoding method is



Fig. 1. Yale face database

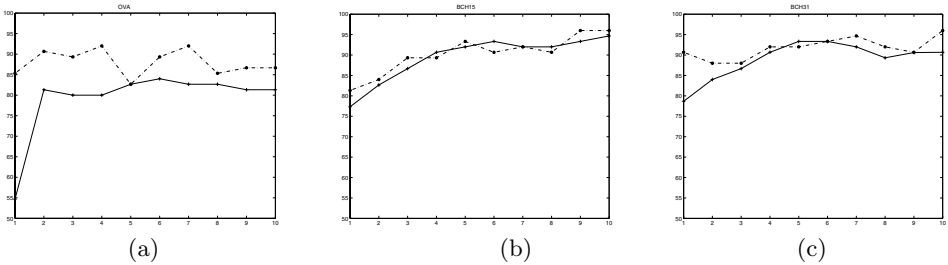


Fig. 2. Recognition rate plotted against C (solid line denotes Hamming decoding scheme and dotted line shows SOM-based decoding scheme) (a) OVA, (b) BCH15, (c) BCH31

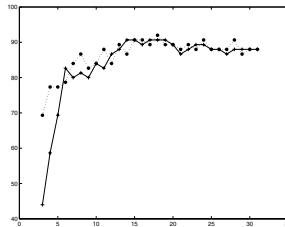


Fig. 3. The change of the recognition with the number of the binary classifiers used (solid line denotes Hamming decoding scheme and dotted line shows SOM-based decoding scheme)

valid. The best effect is gotten when $n = 15$ for hamming decoding (90%) and $n=19$ for SOM (92%) respectively, so that there exists an optimum code length in ECOC.

5 Conclusions and Future Works

An SOM-based decoding method for multi-class SVM is proposed in this paper. This approach fully considers the output magnitude information of the binary SVM by using the topological ordering of SOM. The experiments demonstrate that the new method can achieve better performance than the commonly used

Hamming decoding strategy. As SOM needs training, the decoding time is improved. From the simulations, it is found that the choice of the encoding scheme can also affect the recognition capability, so for a multi-class problem, how many and which binary SVMs to select for combination in order to achieve the best performance is our future research aspect.

References

1. Vapnik, V. N.: The Nature of Statistical Learning Theory. Springer-Verlag, New York (1999).
2. Hsu, C.-W, Lin, C.-Jen.: A Comparison of Methods for Multiclass Support Vector Machines. IEEE Transaction on Neural Networks. **13** (2002) 415-425
3. Dietterich, T.G., Bakiri, G.: Solving Multiclass Learning Problems via Error-correcting Output Codes. Artificial intelligence Research **3** (1995) 263-286
4. Masulli, Valentini, E.: Comparing Decomposition Methods for Classification. In: Proceedings of International Conference on Knowledge-Based Intelligent Engineering Systems and Applied Technologies, Brighton, UK (2000)
5. Yan, P.F., Zhang, C.S.: Artificial Neural Network and Simulant Evolution Computation. Tsinghua University Press, Beijing(2000),(in chinese)
6. Kohonen T.: Self-Organization and Associative Memory. Springer-Verlag, New York (1988)
7. Lin, C.J.: Libsvm: a library for support vector machines version 2.36 <http://csie.ntu.edu.tw/~cjlin/> (2002)

Selective Dissemination of XML Documents Using GAs and SVM

K.G. Srinivasa¹, S. Sharath², K.R. Venugopal¹, and Lalit M. Patnaik³

¹ Department of Computer Science and Engineering,
University Visvesvaraya College of Engineering, Bangalore - 560001, India
kgsrinivas@msrit.edu, venugopalkr@gmail.com

² Infosys Technologies, Bangalore, India
sharathsrinivas@gmail.com

³ Microprocessor Applications Laboratory, Indian Institute of Science, India
lalit@micro.iisc.ernet.in

Abstract. XML has emerged as a medium for interoperability over the Internet. As the number of documents published in the form of XML is increasing there is a need for selective dissemination of XML documents based on user interests. In the proposed technique, a combination of Self Adaptive Migration Model Genetic Algorithm (SAMGA)[5] and multi class Support Vector Machine (SVM) are used to learn a user model. Based on the feedback from the users the system automatically adapts to the user's preference and interests. The user model and a similarity metric are used for selective dissemination of a continuous stream of XML documents. Experimental evaluations performed over a wide range of XML documents indicate that the proposed approach significantly improves the performance of the selective dissemination task, with respect to accuracy and efficiency.

1 Introduction

Extensible Markup Language (XML) is a widely used as a standard for document exchange on the Internet. XML being a markup language, associates structure to the contents of the document using user defined tags. Both the structure and contents of the document can help in effective filtering. The self describing tags and the hierarchical structure of the knowledge can help in efficient selective dissemination of XML documents. In this paper, We have explored the possibility of selective dissemination of XML documents based on a user model. Self adaptive and real coded genetic algorithms are used to learn the user interests. A SVM is used to build a user model based on their interests. A similarity metric between the continuously streaming XML documents and the profiles stored in the user model is proposed [1, 2].

2 Genetic Learning of User Interests

Consider an XML document collection with n documents. Each of the distinct tags in this document collection is stored in a tag pool. Let $T = \{t_1, t_2, \dots, t_m\}$

be the tag pool with m tags and t_i represents i^{th} tag. Usually, for document collection the tag pool is huge. The purpose of Self Adaptive Genetic Algorithms is to select from the tag pool, the tag combinations which are interesting to a user. For the system to learn the user interests, the user has to first issue a set of search queries $q = \{k_1, k_2, \dots, k_m\}$. The documents satisfying the search terms are retrieved as results. The user has to classify the results relevant to him. This is the feedback given to the system in order to learn the user interest. The fitness function used in the GA is given by,

$$fitness = \alpha * \left(\sum_{i=1}^N \frac{freq(i, S_{tag})}{rank(i)} \right) + (1 - \alpha)N \tag{1}$$

where N is the number of documents retrieved with a specific tag configuration, S_{tag} is the set of top k tags with highest tag weights. $freq(i, S_{tag})$ is the frequency of occurrence of the terms of the query $q = \{k_1, k_2, \dots, k_m\}$ within the tags in S_{tag} in the i^{th} retrieved document. The retrieved documents are ranked according to the frequency of occurrence of the terms. The $rank(i)$ denotes the rank of the i^{th} retrieved document provided the document is also classified as relevant by the user. α is a parameter that is used to express the degree of user preference for accuracy of the search results or the total number of documents that are retrieved.

The Self Adaptive Migration Model Genetic Algorithms[5] used in our approach extracts from the tag pool, the combination of tags which are interesting to a user. It builds the profile of a user. Such a profile can be generated using a single query and relevance feedback session with the system or by using multiple sessions to generate aggregate profile. The accuracy and the utility of the selective dissemination tags improve with the number of sessions of interaction between the user and the system.

3 User Model Construction

The Self Adaptive Genetic Algorithms [5] constructs a user profile, which is a set of tag combinations interesting to a user. A user model which can judge the interest category of a user from his profile is to be constructed. We consider this as a supervised learning problem. From the collection of user profiles, random profiles are sampled and a decision on the category to which they belong is made. A feature extractor and a SVM are used for the user model construction.

Feature Extraction: The user profiles constructed using SAMGA [5] can have tags which are rarely used and also tags spread over all the categories. Such tags can deteriorate the performance of SVM. Hence a feature extractor is used to eliminate such tags. The feature extraction is performed using the measure of expected entropy loss. The entropy loss ranks the features high that are discriminators among the categories. It also assigns low ranks to features that cannot act as discriminators.

SVM for User Model Construction: SVM is a machine learning approach for the task of classification which is based on structural risk minimization [4]. Here, the decision surface chosen must minimize the test error on unseen samples. The binary SVM can be extended to support multiclass classification using the *one against one* approach. Here $k(k-1)/2$ SVMs are used where, k is the number of classes. Each SVM trains data from two different classes. A voting vector with a dimension for each class is also used for classification. There are as many votes as the number of SVMs and the class having the maximum number of votes is the result. The result of application of SVM is the user model. The user model has two functions. First, it classifies the various profiles into user interest category. Second, the same model can assign a user interest category to an incoming XML document from among the various prespecified categories.

4 Selective Dissemination

The selective dissemination is the task of disseminating the documents to the users, based on their profiles to whom the incoming documents would be most relevant. The first step in this task is determining the user interest category of an incoming XML document. Next, the similarity between the incoming XML document and the user profiles belonging to the same user interest category are determined. A high similarity value indicates that the document is relevant to the corresponding user.

The similarity between the XML document and the user profile is determined by modeling the XML document as a directed tree $G = (V_g, E_g)$. Each node in V_g corresponds to an XML element and E_g is a set of edges which defines the relationships between the nodes in V_g . A vertex in V_g is said to be at a level lev_i if it is at distance of lev_i from the root. Let $level_i(D_x)$ represent the set of all tags of an XML document D_x at a level lev_i . Let $userp_j$ represent the j^{th} user profile and $userp_j = \{tag_1, tag_2, \dots, tag_l\}$, where l is the total number of tags in the j^{th} user profile. The similarity between a user profile $userp_j$ and the incoming XML document D_x is given by

$$S(D_x, userp_j) = \frac{\sum_{i=1}^d \frac{|userp_j \cap level_i(D_x)|}{i * |level_i(D_x)|}}{|userp_j \cup D_x|} \tag{2}$$

where d is the depth of the XML document tree. The following observations can be made about the similarity metric.

- $0 \leq S(D_x, userp_j) < 1$; Since the XML document tree and the user profiles are structures of different kinds, a perfect similarity between the two can never be achieved.
- $S(D_x, userp_j) = 0$, iff there exists no common tags between the XML documents and the user profile.
- $S(D_x, userp_j) = S(userp_j, D_x)$
- Let D_{x_1} and D_{x_2} be two XML documents so that $|userp_j, D_{x_1}| > |userp_j, D_{x_2}|$ i.e., the number of tags shared between $userp_j$ and D_{x_1} is greater than the

number of tags shared between $userp_j$ and D_{x_2} . However, this does not imply that $S(D_{x_1}, userp_j) > S(D_{x_2}, userp_j)$ i.e., the number of tags shared between the incoming XML document and the user profiles is not the only factor which decides their similarity.

The architecture for selective dissemination of XML documents based on the user model learnt using SAMGA and SVM is given in Figure 1. The user model is used for two purposes. First, it classifies the user profiles among the various user interest categories. The profile is then stored under the corresponding category. Second, for streaming XML documents it determines the user interest category. The metric of Equation 2 is used to find the similarity between the user profiles and the XML document. A high similarity value represents that the corresponding user is interested in the document. The document is disseminated to the top k users whose profiles have the greatest similarity with the input XML document.

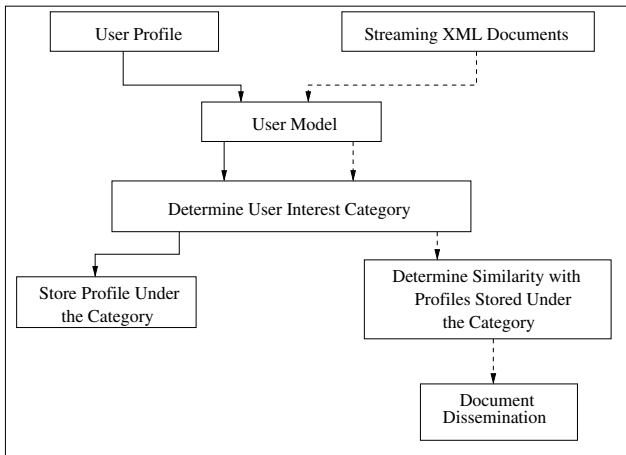


Fig. 1. Architecture of the Selective Dissemination System

5 Performance Studies

The Self Adaptive Migration Model GA used in our approach takes a small number of user queries and the documents adjudged as relevant by the user as input. The GA explores all possible tag combinations from the tag pool and tries to find the best tag combination which satisfies the maximum number of queries. This tag combination forms the profile of a user. Even for small XML document collections the tag pool is usually large. Thus, the GA is trying to find solutions from a problem space which is huge. The experimental result in Fig 2(a) shows the average fitness for the generations of population. The performance of SAMGA and a Simple GA (SGA) are compared. Note that the fluctuations in the curve representing SAMGA is because of the adaptiveness introduced in the migration rate and population size. For SAMGA the average fitness steadily

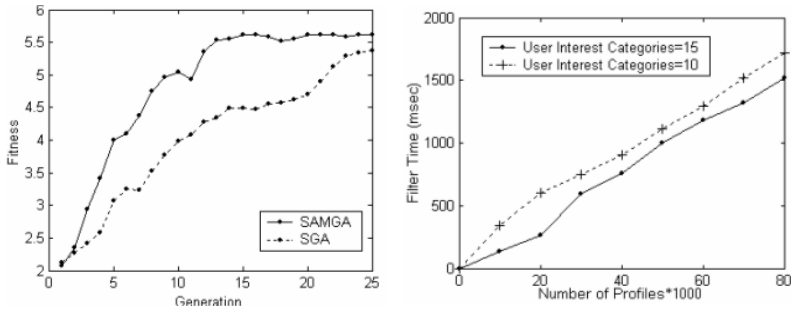


Fig. 2. (a): Average Fitness of Populations; (b): Time for Selective Dissemination

raises until about the fifteenth generation and then the fitness increases slowly. As the generation progresses further, the increment of fitness falls, as most of the individuals have already converged to their best fitness values. In contrast, a SGA fails to converge even after 20 generations. Thus, the application of SAMGA helps in faster convergence when compared to SGA.

The time taken for the dissemination of the documents depends upon two factors: The number of stored user profiles and the number of user interest categories. The user interest categories are various divisions like *sports*, *books*, *politics*, *religion*, etc., to which the user profiles are assigned. It is important to have sufficient numbers of such categories. If the number of user interest categories is less large number of profiles will come under a single category and the time to find a matching profile increases. Thus, maintaining a optimal number of user interest categories results in good performance. The time taken for selective dissemination of XML documents is shown in Fig 2(b).

The number of user interest categories utilized also determines the accuracy of the selective dissemination task. The accuracy of the selective dissemination system is the proportion of disseminated documents that are relevant to the user. The variation of the accuracy with the number of user interest categories is shown in Fig 3(a). In order to validate the accuracy and efficiency of the proposed technique, we compare it with Multi-level Indexing Technique proposed in [3]. From Fig 3(b) it can be observed that the accuracy of selective dissemination

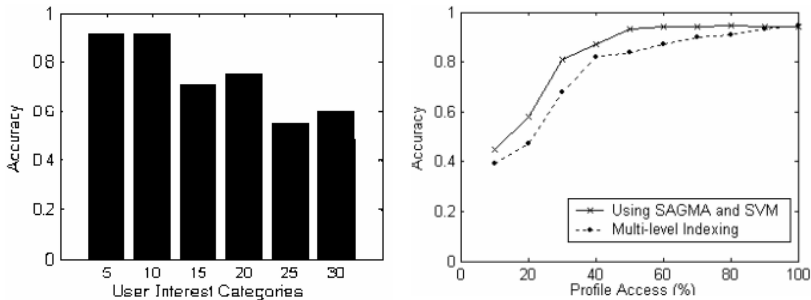


Fig. 3. (a): Acc. of Selective Dissemination; (b): Acc. versus No. of Profile Access

increases with the number of profiles accessed. If an exhaustive search over the user profiles is performed, both the accuracy and time for dissemination increase. Since selective dissemination systems should be able to handle a large number of user profiles, the number of profiles accessed for an incoming document must be limited. From Fig 3(b), the accuracy of both the techniques is same when the percentage of profiles accessed is high. When the percentage of the profiles accessed is in the range of 30-40%, the proposed technique outperforms the Multi-level Indexing strategy in [3]. Thus the application of SAMGA and SVM helps in accurate and fast selective dissemination of XML documents.

6 Conclusions

We have proposed a framework for Selective dissemination of XML documents that makes use of genetic algorithms to learn the user interests. A Self Adaptive Genetic Algorithm is used in particular as the problem space is large and a rapid exhaustive search is required to determine the user interests. Profiles are built which represent the user interests. A model which assigns user interest categories to the user profiles as well as the streaming XML documents is also proposed. A similarity metric between the user profiles and the incoming XML document is used. This metric can find the users to whom the documents would be most relevant. The efficiency and the accuracy of the selective dissemination of XML documents using the proposed technique is superior.

References

1. M. Altinel and M. Franklin: Efficient Filtering of XML Documents for Selective Dissemination of Information. Intl. Conf. on Very Large Databases (VLDB'00), Cairo, Egypt (2000) 53-64
2. Ioana Stanoi, George Mihaila, Sriram Padmanabhan: A Framework for Selective Dissemination of XML documents based on inferred user profiles. In Proc. of the nineteenth International Conference on Data Engineering (ICDE 2003)
3. J. Yang and R.R. Korfhage: Effects of Query Term Weights Modification in Annual Document Retrieval: A Study Based on a Genetic Algorithm. In Proc. of the Second Symposium on Document Analysis and Information Retrieval (1993) 271-185
4. Thorsten Joachims: Transductive inference for text classification using support vector machines. Machine Learning - Proc. Sixteenth Int'l. Conf. (ICML '99), San Francisco, CA, USA (1999) 200-209
5. Srinivasa K. G , Karthik Sridharan, P. Deepa Shenoy, Venugopal K.R, L. M: Patnaik A Dynamic Migration Model for Self-Adaptive Genetic Algorithm. In Proc. of IDEAL 2005, Brisbane, Australia (2005)

A Smoothing Support Vector Machine Based on Exact Penalty Function

Zhiqing Meng¹, Gengui Zhou¹, Yihua Zhu¹, and Lifang Peng²

¹ College of Business and Administration,
Zhejiang University of Technology, Zhejiang 310032, China

² College of Information Engineering,
Xiangtan University, Hunan 411105, China
mengzhiqing@zjut.edu.cn

Abstract. In this paper, we study a smoothing support vector machine (SVM) by using exact penalty function. First, we formulate the optimization problem of SVM as an unconstrained and nonsmooth optimization problem via exact penalty function. Second, we propose a two-order differentiable function to approximately smooth the exact penalty function, and get an unconstrained and smooth optimization problem. Third, by error analysis, we can get approximate solution of SVM by solving its approximately smooth penalty optimization problem without constraint. Compared with artificial neural network and time sequence, the precision of prediction of our smoothing SVM which is illustrated with the numerical experiment is better.

1 Introduction

Recently, support vector machines (SVM) have been employed for classification and prediction in pattern recognition and regression problems. SVM problem needs to solve a constrained mathematical programming [1]. Generally, a simple method dealing with SVM problem is to formulate the problem as a Lagrangian problem [2]. However, the Lagrangian problem has still some constraint and many additional variables, and so it is difficult to solve. In order to overcome the difficulty, some researchers studied smooth SVM [3]. On the other hand, the penalty function is a good tool to solve the constrained optimization problems, such as SVM problem, since it will transform the constrained optimization problems into the unconstrained optimization problems for solving. Recently, Quan and Yang applied the penalty function to SVM [4] in 2003. It is well known that the exact penalty function is better than inexact penalty function for the constrained optimization problems [8]. However, many exact penalty functions are not differentiable so that many good algorithm, such as Newton algorithm, can not be applied. Therefore, exact penalty function needs be approximately smoothed [5,6,7].

In this paper, by using exact penalty function, we introduce a new smoothing function of twice differential to apply to SVM with nonlinear kernel for prediction. We show that when the smoothing parameter ϵ approaches to 0, the

approximate solution of the smoothing problem converges to the solution of SVM. Numerical experiment shows that the method is efficient.

2 Smoothing Support Vector Machine

Let E be an n dimension space. Given a set of data $\{(\mathbf{x}_1; y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$, where $\mathbf{x}_i \in R^m, y_i \in R$, the nonlinear SVM first map the input data $\{\mathbf{x}_1; \mathbf{x}_2; \dots \mathbf{x}_N\}$ into a high-dimensional feature space E by using a nonlinear function column vector $\Phi(\cdot: R^m \rightarrow R^n)$ and then perform a linear regression in this feature space so that

$$y = w^T \Phi(\mathbf{x}) + b,$$

where $w \in E, b$ is the threshold, $w^T \Phi(\mathbf{x})$ denotes the dot product of w^T with Φ and the superscript ‘T’ denotes transpose. To determine the two unknowns (w, b) , we define the following an optimization problem.

$$(P_{SVM}) \min f(w, b, u, v) := \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N [\xi(u_i) + \xi(v_i)] \tag{1}$$

$$s.t. \ g_i(w, b, u, v) := y_i - w^T \Phi(\mathbf{x}_i) - b - \varepsilon - u_i \leq 0, i = 1, 2, \dots, N \tag{2}$$

$$h_i(w, b, u, v) := w^T \Phi(\mathbf{x}_i) + b - y_i - \varepsilon - v_i \leq 0, i = 1, 2, \dots, N \tag{3}$$

$$u_i, v_i \geq 0, i = 1, 2, \dots, N. \tag{4}$$

where $\|\cdot\|$ denotes the Euclidean norm, ε is the e-insensitive loss zone[2], C is a positive constant, $u = (u_1, u_2, \dots, u_N), v = (v_1, v_2, \dots, v_N), u_i, v_i$ are training errors and $\xi(\cdot)$ is the loss function. A squared loss function is often used in the study:

$$\xi(t) := \frac{1}{2} t^2.$$

Note that with the use of the squared e-insensitive loss zone, errors $|(y_i - w^T \Phi(\mathbf{x}_i) - b)|$, which are less than ε , will not contribute any cost to the optimization function of equation (1) since the training errors are zero in this case, u_i or v_i is zero in (2) or (3).

To solve (P_{SVM}) , we define a penalty function as

$$F(w, b, u, v, \rho) := f(w, b, u, v) + \rho \sum_{i=1}^N (\sqrt{\max\{g_i(w, b, u, v), 0\}} + \sqrt{\max\{u_i, 0\}} + \sqrt{\max\{h_i(w, b, u, v), 0\}} + \sqrt{\max\{v_i, 0\}}). \tag{5}$$

Define the corresponding penalty optimization problem without any constraint as follows.

$$(P_\rho) \min_{(w, b, u, v)} F(w, b, u, v, \rho).$$

According to [7,8], F is exact penalty function. Therefore, we can find a solution to (P_{SVM}) by solving (P_ρ) for some ρ . However, since the penalty

function $F(w, b, u, v, \rho)$ is not smooth, we will smooth function F as a twice differential as follows.

Let $p : R^1 \rightarrow R^1, p(t) = \sqrt{\max\{t, 0\}}$. We define $q_\epsilon : R^1 \rightarrow R^1$:

$$q_\epsilon(t) = \begin{cases} 0, & \text{if } t \leq 0, \\ \frac{1}{15\epsilon^2}t^{\frac{5}{2}}, & \text{if } 0 \leq t \leq \epsilon, \\ (t^{\frac{1}{2}} + \frac{2}{3}\epsilon t^{-\frac{1}{2}} - \frac{8}{5}\epsilon^{\frac{1}{2}}), & \text{if } t \geq \epsilon, \end{cases}$$

It is easily shown that $q_\epsilon(t)$ is twice differential. Then, $\lim_{\epsilon \rightarrow 0} q_\epsilon(t) = p(t)$.

Consider the nonlinear penalty functions for (P):

$$G(w, b, u, v, \rho, \epsilon) = f(w, b, u, v) + \rho \sum_{i=1}^N (q_\epsilon(g_i(w, b, u, v)) + q_\epsilon(u_i) + q_\epsilon(h_i(w, b, u, v)) + q_\epsilon(v_i)). \tag{6}$$

where $\rho > 0$. $G(w, b, u, v, \rho, \epsilon)$ is twice differentiable at every (w, b, u, v) . Then, we have the following penalty problems without constraint:

$$(PII_{(\rho, \epsilon)}): \quad \min_{(w, b, u, v)} G(w, b, u, v, \rho, \epsilon).$$

3 Main Results

In this section, we discuss relationship among $(P_{SVM}), (P_\rho)$ and $(PII_{(\rho, \epsilon)})$.

Definition 3.1 Let $\epsilon > 0$. A vector $(w, b, u, v) \in R^n \times R^1 \times R^N \times R^N$ is said to be an ϵ -feasible solution of (P_{SVM}) if

$$\begin{aligned} g_i(w, b, u, v) &\leq \epsilon, & i = 1, 2, \dots, N, \\ h_i(w, b, u, v) &\leq \epsilon, & i = 1, 2, \dots, N, \\ -u &\leq \epsilon, -v &\leq \epsilon. \end{aligned}$$

When $\epsilon > 0$ is very small, ϵ -feasible solution is an approximate solution of (P_{SVM}) , since $\lim_{\epsilon \rightarrow 0} G(w, b, u, v, \rho, \epsilon) = F(w, b, u, v, \rho), \forall \rho$.

Lemma 3.1 For $\epsilon > 0$, we have

$$0 \leq F(w, b, u, v, \rho) - G(w, b, u, v, \rho, \epsilon) \leq \frac{32}{5} N \rho \epsilon^{\frac{1}{2}}, \tag{7}$$

where $\rho > 0$.

Proof. By using the definition of $q_\epsilon(t)$, we have

$$0 \leq p(t) - q_\epsilon(t) \leq \frac{8}{5} \epsilon^{\frac{1}{2}}.$$

As a result,

$$0 \leq p(g_i(w, b, u, v) - q_\epsilon(g_i(w, b, u, v))) \leq \frac{8}{5}\epsilon^{\frac{1}{2}} \quad i = 1, 2, \dots, N.$$

$$0 \leq p(h_i(w, b, u, v) - q_\epsilon(h_i(w, b, u, v))) \leq \frac{8}{5}\epsilon^{\frac{1}{2}} \quad i = 1, 2, \dots, N.$$

$$0 \leq p(u_i) - q_\epsilon(u_i) \leq \frac{8}{5}\epsilon^{\frac{1}{2}} \quad i = 1, 2, \dots, N.$$

$$0 \leq p(v_i) - q_\epsilon(v_i) \leq \frac{8}{5}\epsilon^{\frac{1}{2}} \quad i = 1, 2, \dots, N.$$

Adding up for all i and combing (6), we obtain (7).

By Lemma 3.1, we can easily obtain the following three conclusions.

Theorem 3.1 Let $\{\epsilon_j\} \rightarrow 0$ be a sequence of positive numbers. Assume that for j , (w_j, b_j, u_j, v_j) is a solution to $(\text{PII}_{(\rho, \epsilon_j)})$ for some $\rho > 0$. Let (w^*, b^*, u^*, v^*) be an accumulation point of the sequence $\{(w_j, b_j, u_j, v_j)\}$, then (w^*, b^*, u^*, v^*) is an optimal solution to (P_ρ) .

Theorem 3.2 Let (w^*, b^*, u^*, v^*) be an optimal solution of (P_ρ) and $(\bar{w}, \bar{b}, \bar{u}, \bar{v})$ be an optimal solution of $(\text{PII}_{(\rho, \epsilon)})$. Then

$$0 \leq F(w^*, b^*, u^*, v^*, \rho) - G((\bar{w}, \bar{b}, \bar{u}, \bar{v}, \rho, \epsilon) \leq \frac{32}{5}N\rho\epsilon^{\frac{1}{2}} \quad (8)$$

Proof. From Lemma 3.1 we have

$$F(w, b, u, v, \rho) \leq G(w, b, u, v, \rho, \epsilon) + \frac{32}{5}N\rho\epsilon^{\frac{1}{2}}$$

Consequently,

$$\inf_{(w, b, u, v)} F(w, b, u, v, \rho) \leq \inf_{(w, b, u, v)} G(w, b, u, v, \rho, \epsilon) + \frac{32}{5}N\rho\epsilon^{\frac{1}{2}},$$

which proves the right-hand side inequality of (8). The left-hand side inequality of (8) can be proved by Lemma 3.1.

Theorem 3.3 Let (w^*, b^*, u^*, v^*) be an optimal solution of (P_ρ) and $(\bar{w}, \bar{b}, \bar{u}, \bar{v})$ be an optimal solution of $(\text{PII}_{(\rho, \epsilon)})$. Furthermore, let (w^*, b^*, u^*, v^*) be feasible to (P_{SVM}) and $(\bar{w}, \bar{b}, \bar{u}, \bar{v})$ be ϵ -feasible to (P_{SVM}) . Then

$$0 \leq f(w^*, b^*, u^*, v^*) - f(\bar{w}, \bar{b}, \bar{u}, \bar{v}) \leq \frac{64}{5}N\rho\epsilon^{\frac{1}{2}}. \quad (9)$$

Proof. By Lemma 3.1 and Definition 3.1, we get

$$0 \leq f(w^*, b^*, u^*, v^*) - [f(\bar{w}, \bar{b}, \bar{u}, \bar{v}) + \rho \sum_{i \in I} (q_\epsilon(g_i(\bar{w}, \bar{b}, \bar{u}, \bar{v})) + q_\epsilon(\bar{u}_i) + q_\epsilon(h_i(\bar{w}, \bar{b}, \bar{u}, \bar{v})) + q_\epsilon(\bar{v}_i))] \leq \frac{32}{5}N\rho\epsilon^{\frac{1}{2}}, \quad (10)$$

which implies (9).

Theorems 3.2 and Theorem 3.3 mean that an approximate solution to $(PII_{(\rho,\epsilon)})$ is also an approximate solution to (P_ρ) when $\epsilon > 0$ is sufficiently small. Moreover, an approximate solution to $(PII_{(\rho,\epsilon)})$ also becomes an approximate optimal solution to (P_{VSM}) by Theorem 3.3 if the approximate solution is ϵ -feasible. Therefore, we may obtain an approximate optimal solution to (P_{VSM}) by finding an approximate solution to $(PII_{(\rho,\epsilon)})$.

4 Numerical Results

In this section, we do some numerical experiments by using approximately exact penalty function of $(PII_{(\rho,\epsilon)})$ for the stock price of prediction in the China’s stock data. In the experiment, we choose sample data of all 104 days for SHAHE’s stock from 14 March 2002 to 19 August 2002. The ending price of each day is taken as training data for all 81 days from 14 March 2002 to 22 July 2002. After 22 July 2002, the ending price of each day is define as testing data for all 20 days. By solving problem (P_{SVM}) , we obtained prediction value y_t of the ending price of each day from 23 March 2002 to 19 August 2002, where y'_t is factual value of each day. Then, we get an error between y_t and y'_t : $|y_t - y'_t|$. In order to compare it with precision of prediction, we calculate error $|y'_t - y''_t|$, where prediction value y''_t is obtained by artificial neural network. And we calculate error $|y'_t - y'''_t|$, where prediction value y'''_t is obtained by time sequence.

Let penalty parameter $\rho = 10, 100, 1000$, and $\epsilon = 10^{-8}$. We calculate some results of ending price about error of SVM, error of neural networks and time sequence in Table 1.

Table 1. Compare error of SVM with error of Neural Networks and Time Sequence

Day of Forecast	Error of SVM	Error of Neural Networks	Error of Time Sequence
1	0.006795	0.008599	0.009419
2	0.006955	0.009500	0.010873
3	0.007129	0.010168	0.013198
4	0.006847	0.012237	0.018581
5	0.006531	0.015135	0.033308
8	0.006531	0.013882	0.033308
19	0.006531	0.015653	0.033308

Table 1 shows that the the prediction value of SVM is better precision than neural network and time sequence. Especially, the the prediction value of SVM is better within longer time for short prediction.

5 Conclusion

This paper gives a smoothing SVM problem’s $(PII_{(\rho,\epsilon)})$ based on exact penalty function. The smoothed exact penalty function is twice differential. An approximate solution to $(PII_{(\rho,\epsilon)})$ is proved to be an approximate solution to (P_{SVM})

when $\epsilon > 0$ is sufficiently small. Experiment's results show that the way is efficient for the prediction of regression problems.

Acknowledgements

This research work was partially supported by grant No. NX05LJ07 from Zhejiang Provincial Philosophical & Social Science Foundation.

References

1. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition, *Data Mining and Knowledge Discovery* **2**(1998) 121-167
2. Lee, K. L. and Billings, S. A.: Time series prediction using support vector machines, the orthogonal and the regularized orthogonal least-squares algorithms. *International Journal of Systems Science* **33**(2002) 811-821
3. Yuh-Jye Lee and Mangasarin, O.L.: SSVM: A Smooth Support Vector Machine for Classification. *Computational Optimization and Applications* **20**(2001) 5-22
4. Yong Quan and Jie Yang: An Improved Parameter Tuning Method for Support Vector Machines. *Lecture Notes in Computer Science* **2639**(2003) 607-610
5. Zenios, S.A., Pinar, M.C., Dembo, R.S.: A smooth penalty function algorithm for network-structured problems. *European Journal of Operation Research* **64**(1993) 258-277
6. Meng, Z.Q., Dang, C.Y., Zhou G., Zhu Y., Jiang M.: A New Neural Network for Non-linear Constrained Optimization Problems, *Lecture Notes in Computer Science*, Springer **3173** (2004) 406-411
7. Yang, X.Q., Meng, Z.Q., Huang, X.X., Pong, G.T.Y.: Smoothing nonlinear penalty functions for constrained optimization, *Numerical Functional Analysis Optimization* **24** (2003) 351-364
8. Rubinov, A. M., Glover, B. M. and Yang, X. Q.: Decreasing functions with applications to penalization, *SIAM Journal. Optimization* **10**(1999) 289-313

Speech Acts Tagging System for Korean Using Support Vector Machines*

Songwook Lee¹, Jongmin Eun², and Jungyun Seo²

¹Div. of Computer & Information Engineering, Dongseo University,
San69-1, Jurye2-dong, Sasang-gu, Busan 617-716, South Korea
leesw@dongseo.ac.kr

²Dept. of Computer Science and Interdisciplinary Program of Integrated Biotechnology,
Sinsu-dong 1, Mapo-gu, Seoul 121-742, South Korea
uook@hanafos.com, seojy@sogang.ac.kr

Abstract. We propose a speech-act analysis method for Korean dialogue using Support Vector Machines (SVM). We use a lexical word, its part of speech (POS) tags, and bigrams of POS tags as utterance feature and the contexts of the previous utterance as context feature. We select informative features by χ^2 statistic. After training SVMs with the selected features, SVM classifiers determine the speech-act of each utterance. In experiment, we acquired overall 90.5% of accuracy with dialogue corpus for hotel reservation domain.

1 Introduction

In a natural language dialogue, speech act is an abstraction of a speaker's intention which the speaker wants to represent through utterance. Thus identifying the speech acts of utterances is very important to understand speaker's intention and it is essential to produce relevant responses in a natural language dialogue system [1, 2]. The dialogue system must consider the semantic information and the flow of dialogue to interpret speech act. However, it is difficult to infer the speech act from a surface utterance because the same utterance may represent a different speech act according to the context.

Recently, there have been many works for the speech act analysis utilizing a tagged dialogue corpus with machine learning methods. [4] proposes a statistical method for analyzing both speech act and discourse structure by using Maximum Entropy model and shows that discourse structure is helpful for speech act analysis. [5] utilizes decision trees to compute transition probabilities with previous utterance's speech act and observation probabilities with syntactic patterns and speaker information extracted from a given utterance. Then, they used those probabilities in Hidden Markov model. [6] proposes a neural network model using features such as lexical information, part of speech (POS) tag, and the speech act of the previous utterance for a given utterance. They selected useful features by χ^2 statistic.

* This research was performed for the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Commerce, Industry and Energy of Korea.

We propose a speech act tagging system using Support Vector Machines (SVM). Utterance feature and context feature are used to train SVM. Utterance feature is extracted from a given utterance to represent the meaning of the utterance and context feature is extracted from the previous utterance to represent the flow of dialogue. Useful features are selected by χ^2 statistic which [6, 9] have proposed. The speech act of a given utterance is determined by the SVM classifier which produces the largest weight. We show the type of features which are used for training SVMs and explain feature selection method using χ^2 statistic in Section 2. In Section 3, we explain how we apply SVM in this task. In Section 4, we show the experimental results and conclude in Section 5.

2 Features for Training SVM

2.1 Dialogue Corpus and the Type of Speech Acts

We use a Korean dialogue corpus which was transcribed from recordings in domains such as hotel reservation, airline reservation, and tour reservation [3]. This corpus consists of 528 dialogues with 10,285 utterances. Each utterance in a dialogue is manually annotated with discourse knowledge such as speaker (SP), syntactic pattern (ST), speech act (SA) and discourse structure (DS) information. SP has a value of either ‘User’ or ‘Agent’. Fig. 1 shows a part of the annotated dialogue corpus.

/SP/ User /KS/Sug-so-e gwan-hae-seo mun-ui-hal sa-hang-i iss-eo-seo-yo. (I have some questions about lodgings.) /ST/[decl,paa,present,no,none,none] /SA/ask-ref /DS/[2]	/SP/Agent /KS/Georgia-dae-hak-ui eo-hak-yeon-su course-neun dae-hak-e gi-suk-sa-reul je-gong-ha-go iss-seub-ni-da. (There is a dormitory in University of Georgia for language course students.) /ST/[decl,pvg,present,no,none,none] /SA/response /DS/[2]
--	--

Fig. 1. A part of the annotated dialogue corpus

We will use SA, ST, and SP which are extracted from the annotated dialogue corpus for our experiment and not consider discourse structure. We use 17 types of speech acts in the corpus such as *accept*, *acknowledge*, *ask-confirm*, *ask-if*, *ask-ref*, *closing*, *correct*, *expressive*, *inform*, *introducing-oneself*, *offer*, *opening*, *promise*, *reject*, *request*, *response*, and *suggest*.

2.2 Utterance Feature and Context Feature

The utterance feature consists of ST, lexical words automatically annotated with POS tags by [10], and bi-grams of POS tags of all words in an utterance.

USER : Suk-so-e gwan-hae-seo mun-ui-hal sa-hang-i iss-eo-seo-yo. (I have some questions for lodging.) Word/POS : Suk-so/nc e/jca gwan-ha/pv eo-seo/ecs mun-ui-ha/pv -l/exm sa-hang/nc -i/jc iss/pa eo-seo-yo/ef ./s. POS bigram : [nc,jca] [jca, pv] [pv, ecs] [ecs, pv] [pv, exm] [exm, nc] [nc, jc] [jc, pa] [pa, ef] [ef, s.] ST : decl, paa, present, no, none, none
--

Fig. 2. An example of utterance feature for a user utterance

Fig. 2 represents an example of the utterance feature. ST consists of the selected syntactic features of an utterance. Surface utterances can be different, but can convey the same meaning. ST is used to generalize them. Table 1 shows the syntactic features of the syntactic pattern [2, 5]. An utterance can have different speech acts as the context of dialogue. Thus we take into account the previous utterances as the context feature. The context feature consists of the features of the previous utterance such as utterance feature, speaker, and speech act.

Table 1. Syntactic features used in the syntactic pattern

Syntactic feature	Example	Total	Syntactic feature	Example	Total
Sentence Type	yn_quest, decl, wh_quest, imperative	4	Negativeness	no, yes	2
Main-Verb	be, know, ask, promise, etc.	88	Aux-Verb	want, will, seem, intend, etc.	31
Tense	present, future, past	3	Clue Word	yes, okay, hello, etc.	26

2.3 Feature Selection and Calculation of Feature's Weight

We remove non-informative features by using a well-known χ^2 statistic. The χ^2 statistic measures the lack of independence between a feature value f and a speech act s and can be compared to the χ^2 distribution with one degree of freedom to judge extremeness [9]. Using the two-way contingency table of f and s , where A is the number of times f and s co-occur, B is the number of times the f occurs without s , C is the number of times s occurs without f , D is the number of times neither s nor f occurs, the feature-goodness measure is defined to be:

$$\chi^2(f, s) = \frac{(A+B+C+D) \times (AD-CB)^2}{(A+C) \times (B+D) \times (A+B) \times (C+D)} \quad (1)$$

The χ^2 statistic has a natural value of zero if f and s are independent. We compute the χ^2 statistic between each unique feature in a training dialogue corpus and each speech act, and then we calculate the feature scores associated with each speech act by Equation (2). On the experimental basis, we select n feature values that are highly ranked by the χ^2 statistic.

$$\chi^2_{\max}(f) = \max_{i=1}^m \{ \chi^2(f, s_i) \} \quad (2)$$

There are many methods to give weight to a feature. We have tried to use binary weight, TF-IDF (Term Frequency-Inverse Document Frequency) weight, or TF-ICF (Term Frequency-Inverse Category Frequency) weight. Among them, we choose the method which shows best performance. To calculate TF-IDF and TF-ICF weight for the task of speech act analysis, a feature is regarded as term, utterance as document, and the speech act as category.

3 Training Support Vector Machines

The technique of SVM is a learning approach for solving two-class pattern recognition problems introduced by Vapnik [7]. It is based on the Structural Risk

Minimization principle for which the error-bound analysis has been theoretically motivated. The problem is to find a decision surface that optimally separates the data points into two classes. It can be solved using quadratic programming techniques. The algorithms for solving linearly separable cases can be extended so that they can solve linearly non-separable cases as well by either introducing soft margin hyperplanes, or by mapping the original data vectors to a higher dimensional space where the new features contain interaction terms of the original features, and the data points in the new space become linearly separable.

We use SVM^{light} [8] for our experiment. To train SVM, we use the selected features explained in Section 2. The dimension of a feature vector is determined by the number of features. The weight of each feature is represented on its corresponding domain. That is, an observed data token is indicated as a point in the vector space. The speech act of given utterance is utilized as a positive example for the relevant specific speech act classifier. At the same time, it is utilized as a negative example for speech act classifiers of any other categories.

Because we have seen that the kernel of SVM does not strongly affect the performance of our problem through many experiments, it is concluded that our problem is linearly separable. Thus we will use only the linear kernel. As SVM is a binary class classifier, we construct 17 classifiers so that each class can represent one of the speech acts. Each classifier constructs a hyperplane which demarcates one specific class from all the other classes. We select the classifier which has the maximal distance from the margin for each test data point.

4 Experimental Results

We used the tagged corpus explained in Section 2. We divided the corpus into the training data with 428 dialogues, 8,349 utterances, and the test data with 100 dialogues, 1,936 utterances.

Table 2 represents the performance of our system with the type of features (the system uses binary weight and the number of features is 3,000) as an accuracy on the test data. ① is the result of using utterance features described in Subsection 3.2, ② uses the utterance features of the previous utterance as context feature in addition to ①. ③ uses utterance features of the previous utterance and ④ uses the determined speech act of the previous utterance as the context feature. As seen as in Table 3, using the speech act of the previous utterance as context feature shows better performance than using the previous utterance feature. In addition to ④, we acquired the best results with speaker information.

Table 3 represents the performance of the system using feature set ⑤ with the number of features which are selected by the χ^2 statistic. We sort the features with χ^2 statistic by descending order and select higher ranked features. As showed in Table 4, we acquired best results with 3,000 features.

Table 4 represents the system performance with the kind of weighting methods for the selected features by χ^2 statistic. TF-IDF or TF-ICF does not contribute to the system performance. Since most features occur only once, most features' TFs have the same frequency- 1. As a result, TF does not have a good effect on the system performance and simple binary weight shows the best performance.

Table 2. System performance with the types of features

	Type of features	Accuracy
①	Utterance features	80.0%
②	① + ST	83.8%
③	② + features of the previous utterance	85.1%
④	② + SA of the previous utterance	88.1%
⑤	④ + SP	90.5%

Table 3. The performance of the system with the number of features

# of the features	Accuracy
1000	87.7%
2000	89.8%
2500	90.3%
3000	90.5%
3500	87.5%
4000	87.5%

Table 4. The system performance with the kind of weighting methods

Type of weight	Accuracy
Binary weight	90.54%
TF-IDF weight	88.52%
TF-ICF weight	85.53%

Table 5. The performance of other systems

System	Accuracy
Maximum Entropy model[4]	80.5%
HMM with Decision Trees[5]	81.5%
Neural Network[6]	85.2%
The proposed system	90.5%

Table 5 compares the performance of the proposed system with other machine learning based systems such as maximum entropy model [4], HMM with decision trees [5], and neural network model [6]. Comparing our system with other machine learning based systems with the same test data set, our system using SVMs shows the highest accuracy of 90.5%. Even though it is difficult to compare the proposed system directly with the other systems because the types of feature are slightly different from each other, we can say that our proposed system achieved the-state-of-the-art performance among them in Korean speech act analysis on the same test data.

The errors are mainly caused by insufficient training data. The accuracies of speech acts having low frequencies in the training data such as ‘accept’, ‘suggest’, ‘reject’, and ‘offer’ are lower than other speech acts. Especially, utterances like ‘inform’, ‘suggest’, and ‘offer’ are leading utterances that do not depend on the previous utterances but initiate new sub-dialogue. It is difficult to predict these speech acts by using contexts [11]. Low frequencies of these utterances in the training data make it more difficult than other utterances. However, it is interesting that ‘promise’ is predicted well although this utterance is rare since it has unique features which other speech acts do not have. Therefore, new type of features playing an important role in predicting speech act must be investigated.

5 Conclusion and Future Works

We analyzed speech act of Korean dialogue using SVM classifiers. SVM classifiers are trained with utterance feature and context feature. Utterance feature consists of syntactic features, POS tagged lexical words, and bi-grams of POS tags. Context feature consists of speaker information and the speech act of the previous utterance. We selected 3,000 useful features with χ^2 statistic. The speech act of an input utterance is determined by comparing the weight of SVM classifiers trained with selected features. We acquired an accuracy of 90.5% with the trained SVM classifiers.

The proposed system is developed in hotel reservation domain. To expand to other domains for general applications, it is required to define various types of speech act for general purpose dialogue system because some types of speech act are defined for this domain. It is also required to collect dialogue corpus in various domains.

References

1. Lambert, L., Caberry, S.: A Tripartite Plan-Based Model of Dialogue. *Proceedings of ACL* (1991) 47-54
2. Chu-Carroll, J.: A Plan-Based Model for Response Generation in Collaborative Negotiation Dialogues. Ph.D. thesis, University of Delaware (1996)
3. Lee, J. W., Seo, J., Kim, G. C.: A Dialogue Analysis Model with Statistical Speech Act Processing for Dialogue Machine Translation. *Proceedings of IWSLT* (1997) 10-15
4. Choi, W. S., Cho, J., Seo, J.: Analysis System of Speech Acts and Discourse Structures Using Maximum Entropy Model. *Proceedings of ACL* (1999) 230-237
5. Lee, S., Seo, J.: Korean Speech Act Analysis System Using Hidden Markov Model with Decision Trees. *IJCPOL*, Vol. 15. No. 3 (2002)
6. Kim, K., Seo, J.: Feature Selection in Automatic Speech Act Tagging. *Proceedings of ICCPOL* (2001) 379-383
7. Vapnik, V.: *The Nature of Statistical Learning Theory*. Springer, New York (1995)
8. Joachims, T.: <http://svmlight.joachims.org>
9. Yang, Y., Pedersen, Jan O.: A Comparative Study on Feature Selection in Text Categorization. *Proceedings of ICML* (1997)
10. Lee, S. H.: Korean Part-of-Speech Tagging System with Handling Unknown Words. MS thesis, KAIST, Korea (1995)
11. Nagata, M., Morimoto, T.: First Steps towards Statistical modeling of Dialogue to Predict the Speech Act Type of the Next Utterance. *Speech Communication*, Vol. 15 (1994) 193-203

A New Support Vector Machine for Multi-class Classification*

Zhiquan Qi¹, Yingjie Tian², and Naiyang Deng^{1,**}

¹ College of Science, China Agricultural University 100083, Beijing, China
qizhiquan2003@tom.com, dengnaiyang@vip.163.com

² Chinese Academy of Sciences Research Center on Data Technology & Knowledge
Economy 100080, Beijing, China
tianyijingjie1213@163.com

Abstract. Support Vector Machines (SVMs) for classification – in short SVM – have been shown to be promising classification tools in many real-world problems. How to effectively extend binary SVC to multi-class classification is still an on-going research issue. In this article, instead of solving quadratic programming (QP) in Algorithm K -SVCR and Algorithm ν - K -SVCR, a linear programming (LP) problem is introduced in our algorithm. This leads to a new algorithm for multi-class problem, K -class Linear programming ν -Support Vector Classification-Regression (Algorithm ν - K -LSVCR). Numerical experiments on artificial data sets and benchmark data sets show that the proposed method is comparable to Algorithm K -SVCR and Algorithm ν - K -SVCR in errors, while considerably faster than them.

1 Introduction

Classification refers to constructing a decision function $f(x)$ defined from an input space $\mathcal{X} = R^n$ onto an unordered set of classes $\mathcal{Y} = \{\theta_1, \theta_2, \dots, \theta_K\}$, where $K \geq 2$ is an integer, based on independently and identically distributed (i.i.d) training set

$$T = \{(x_1, y_1), \dots, (x_l, y_l)\} \in (\mathcal{X} \times \mathcal{Y})^l. \quad (1)$$

Binary classification and multi-class classification correspond respectively the case $K = 2$ and the case $K \geq 3$. For both case, Support Vector Classification (SVC) is very promising. This paper is concerned with the latter case, which is more interesting since real world classification problems are usually multi-class. Currently, there are roughly two types of SVC algorithms to solve the multi-class classification problems. One is the ‘decomposition-reconstruction’ architecture approach which mainly makes direct use of binary SVC to tackle the tasks of multi-class classification, such as the ‘one versus the rest’ method[1, 2], and ‘the one versus one’ method [3, 4], and the ‘error-correcting output code’ method[7, 8].

* This work is supported by the National Natural Science Foundation of China (No.10371131).

** Corresponding author.

The other is the ‘all together’ approach[11, 12, 13, 14, 15], where the multi-class classification is solved through only one optimization formulation. It should be pointed out that there is a new variant[5, 6] in the decomposition-reconstruction approach. Instead of using binary SVC, it uses 3-class classification, where the decision function of a 3-class classification is obtained by a combination of SVC and support Vector Regression (SVR). The new variant is not only more robust, but also more efficient.

In this article, following the idea of [5, 6], we propose a new algorithm called as K -class Linear programming Support Vector Classification-Regression (K -LSVCR) .This algorithm can be considered as an improved version of [6]. Their main difference is that our algorithm depends on solving linear programming instead of quadratic programming.

The rest of this article is organized as follows. In Section 2, we describe our algorithm ν - K -LSVCR . Section 3 gives numerical results to show the superiority of our new algorithm over the counterparts in [5, 6]. Section 4 concludes the article.

2 The Fomulation of ν -K-LSVCR

Let us turn to multi-class classification problem with the training set (1). Corresponding to ‘one versus one’ and ‘one versus rest’ in decomposition-reconstruction approach, our algorithm, just like the algorithms in [5] and [6], is based on ‘one versus one versus rest’. More precisely, it is based on (Θ_j, Θ_k) classifier, where $j, k = 1, \dots, K$ with $j < k$. The corresponding decision function $f_{\Theta_{jk}}(x)$ takes 3 values: 1, -1 and 0; these values imply respectively that $x \in \Theta_j, x \in \Theta_k$ and $x \in \mathcal{Y} \setminus \{\Theta_j, \Theta_k\}$. Obviously, in total there are $K(K-1)/2$ decision functions. For the multi-class classification problem, it is not difficult to get its solution by summarizing all values of these decision functions. In this section, we first introduce the (Θ_j, Θ_k) classifier, then describe our algorithm — Algorithm ν - K -LSVCR .

2.1 The (Θ_j, Θ_k) Classifier

Consider the training set T defined by (1). For an arbitrary pair $(\Theta_j, \Theta_k) \in (\mathcal{Y} \times \mathcal{Y})$ of classes with $j < k$, we construct a decision function $f_{\Theta_{jk}}(x)$ which divides the inputs into three classes: the class Θ_j and the class Θ_k as well as the remaining classes $\mathcal{Y} - \{\Theta_j, \Theta_k\}$. The corresponding training set consists of three parts and is denoted as

$$\begin{aligned} \tilde{T} = \{ & (\tilde{x}_1, \tilde{y}_1), \dots, (\tilde{x}_{l_1}, \tilde{y}_{l_1}), (\tilde{x}_{l_1+1}, \tilde{y}_{l_1+1}), \dots, (\tilde{x}_{l_1+l_2}, \tilde{y}_{l_1+l_2}), \\ & (\tilde{x}_{l_1+l_2+1}, \tilde{y}_{l_1+l_2+1}), \dots, (\tilde{x}_l, \tilde{y}_l) \}, \end{aligned} \tag{2}$$

which is obtained from T in (1) by the following way:

$$\begin{aligned} \{\tilde{x}_1, \dots, \tilde{x}_{l_1}\} &= \{x_i | y_i = \Theta_j\}, \quad \{\tilde{x}_{l_1+1}, \dots, \tilde{x}_{l_1+l_2}\} = \{x_i | y_i = \Theta_k\}, \\ \{\tilde{x}_{l_1+l_2+1}, \dots, \tilde{x}_l\} &= \{x_i | y_i \neq \Theta_j \text{ and } y_i \neq \Theta_k\}, \end{aligned} \tag{3}$$

with

$$\tilde{y}_i = \begin{cases} +1, & i = 1, \dots, l_1; \\ -1, & i = l_1 + 1, \dots, l_1 + l_2; \\ 0, & i = l_1 + l_2 + 1, \dots, l. \end{cases} \tag{4}$$

In the following, we denote

$$l_{12} = l_1 + l_2, \quad l_3 = l - l_{12}. \tag{5}$$

For fixed $C, D \geq 0$, $\nu_1, \nu_2 \in (0, 1]$ and $K(\cdot, \cdot)$ chosen prior, following the basic idea in [5, 6] and combining ν -SVC and ν -SVR in linear programming formulations described in [9] and [10] respectively, we thus obtain a LP problem:

$$\begin{aligned} \min_{\alpha, \alpha^*, \xi, \eta, \eta^*, b, \rho, \epsilon} \quad & \frac{1}{l} \sum_{i=1}^l (\alpha_i + \alpha_i^*) + C \left(\frac{1}{l_{12}} \sum_{i=1}^{l_{12}} \xi_i - \nu_1 \rho \right) \\ & + D \left(\frac{1}{l_3} \sum_{i=l_{12}+1}^l (\eta_i + \eta_i^*) + \nu_2 \epsilon \right) \end{aligned} \tag{6}$$

$$\text{s.t.} \quad \frac{1}{l} \sum_{j=1}^l (\alpha_j + \alpha_j^*) = 1, \tag{7}$$

$$\tilde{y}_i \left(\sum_{j=1}^l (\alpha_j - \alpha_j^*) K(\tilde{x}_j, \tilde{x}_i) + b \right) \geq \rho - \xi_i, \quad i = 1, \dots, l_{12}, \tag{8}$$

$$\sum_{j=1}^l (\alpha_j - \alpha_j^*) K(\tilde{x}_j, \tilde{x}_i) + b \leq \epsilon + \eta_i, \quad i = l_{12} + 1, \dots, l, \tag{9}$$

$$\sum_{j=1}^l (\alpha_j - \alpha_j^*) K(\tilde{x}_j, \tilde{x}_i) + b \geq -\epsilon - \eta_i^*, \quad i = l_{12} + 1, \dots, l, \tag{10}$$

$$\alpha, \alpha^*, \xi, \eta, \eta^* \geq 0, \rho \geq \epsilon \geq 0, \tag{11}$$

where $\alpha, \alpha^*, \xi, \eta$ and η^* are the vectors respectively with the components $\alpha_i, \alpha_i^*, \xi_i, \eta_i, \eta_i^*$. Suppose $(\bar{\alpha}, \bar{\alpha}^*, \bar{\xi}, \bar{\eta}, \bar{\eta}^*, \bar{b}, \bar{\rho}, \bar{\epsilon})$ is an optimal solution of problem (6)–(11), the decision function can be expressed as

$$f_{\Theta_{j,k}}(x) = \begin{cases} +1, & \text{if } g(x) \geq \epsilon; \\ -1, & \text{if } g(x) \leq -\epsilon; \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

where

$$g(x) = \sum_{i=1}^l (\bar{\alpha}_i - \bar{\alpha}_i^*) K(\tilde{x}_i, x) + \bar{b}. \tag{13}$$

According to (12) and (13), it is natural to define an input x_i in the training set to be a support vector (SV) if $(\bar{\alpha}_i - \bar{\alpha}_i^*) \neq 0$.

It should be pointed out that the parameter ν in the (Θ_i, Θ_k) classifier has a similar meaning to that in ν -SVM [6].

2.2 The Algorithm ν -K-LSVCR

In order to solve multi-class classification problem with the training set (1), we propose our Algorithm ν -K-LSVCR as follows: for each pair (Θ_j, Θ_k) with $j < k$, we construct a (Θ_j, Θ_k) classifier (12). So we have $K(K-1)/2$ classifiers in total. Hence, for a new input x , $K(K-1)/2$ outputs are obtained. We translate these outputs as follows: When $f_{\Theta_{j,k}}(x) = +1$, a positive vote is added on Θ_j , and no votes are added on the other classes; when $f_{\Theta_{j,k}}(x) = -1$, a positive vote is added on Θ_k , and no votes are added on the other classes; when $f_{\Theta_{j,k}}(x) = 0$ a negative vote is added on both Θ_j and Θ_k , and no votes are added on the other classes. After translating all of the $K(K-1)/2$ outputs, we get the total votes of each class by adding its positive and negative votes. Finally, x will be assigned to the class that gets the most votes.

3 Experiments

In order to compare our Algorithm ν -K-LSVCR with the counterparts in [5, 6], we use the same test problems, including the artificial data sets and the benchmark data sets. Our experiments are carried out using Matlab v7.0 on Intel Pentium III 800MHz PC with 256 MHz of RAM. For the Artificial Data Sets, we obtain similar results to that in [6]. The detail is omitted here.

For the Benchmark Data Sets, our Algorithm ν -K-LSVCR is tested on a collection of three benchmark data sets: ‘Iris’, ‘Wine’ and ‘Glass’, from the UCI machine learning repository [16]. Each data set is first split randomly into ten subsets. Then one of these subsets is reserved as a test set and the others are summarized as the training set; this process is repeated ten times.

For data sets ‘Iris’ and ‘Wine’, the polynomial kernels $K(x, x') = (x \cdot x')^d$ with degree $d = 4$ and $d = 3$ are employed respectively. For data set ‘Glass’, the Gaussian kernel $K(x, x') = \exp(-\frac{\|x - x'\|^2}{2\sigma^2})$ with $\sigma = 0.22$ is employed. We compare the obtained results with that by Algorithm K-SVCR in [5] and Algorithm ν -K-SVCR in [6] in Table 1.

Table 1. Results Comparison

	K-SVCR		ν -K-SVCR		ν -K-LSVCR	
	Error	Time	Error	Time	Error	Time
Iris	[1.97,3.2]	154.23s	[1.42, 2.89]	145.27s	[1.62,3.2]	10.3s
Wine	[2.41,4.54]	178.11s	[2.74,3.98]	189.29s	[2.40,4.50]	14.6s
Glass	[31.23,37.42]	2577.20s	[33.48,37.21]	1765.56s	[31.34,36.51]	17.8s

In Table 1, $[\cdot, \cdot]$ in ‘Error’ columns refers to two kinds of error percentage; The first number is the percentage of error when examples are finally assigned to the wrong classes, while the second number is the percentage of error when examples are assigned to the wrong classes by any decision function $f_{\Theta_{j,k}}(x)$, $j, k = 1, \dots, l$

with $j < k$. The number in the ‘Time’ columns is the number of seconds consumed by the corresponding algorithm. Remarkably, Table 1 shows that the time consumed by our algorithm is much less than the others while their errors are in the same level. Generally speaking, our algorithm is faster than both of them over ten times.

4 Conclusion

In this paper we have proposed a new algorithm, Algorithm ν - K -LSVCR, for the multi-class classification by solving linear programming. Because this new algorithm has the same structure with Algorithm K -SVCR in [5], it can also be proved that it has good robustness. Experiments have shown that our algorithm is considerably faster, usually over ten times, than both Algorithm K -SVCR in [5] and Algorithm ν - K -SVCR in [6] while the same level of errors are kept. Therefore, It is suitable for solving large-scale data sets. Future research includes comprehensive test of the new algorithm and its parameters selection.

References

1. Bottou, L., Cortes, C., Denker, J.S., Drucker, H., Guyon, I., Jackel, L.D., LeCun, Y., Müller, U.A., Sackinger, E., Simard, P., and Vapnik, V.: Comparison of classifier methods: a case study in handwriting digit recognition. in: IAPR (Ed.), Proceedings of the International Conference on Pattern Recognition, IEEE Computer Society Press (1994) 77-82
2. Vapnik, V.: Statistical Learning Theory. John Wiley&Sons (1998)
3. Hastie, T. J., Tibshirani, R. J.: Classification by pairwise coupling. in: M. I. Jordan, M. J. Kearns and S. A. Solla (Eds.), Advances in Neural Information Processing Systems, MIT Press, Cambridge, MA 10 (1998) 507-513
4. Krebel, U.: Pairwise classification and support vector machines. in: B. Schölkopf, C. J. C. Burges and A. J. Smola (Eds.), Advances in Kernel Methods: Support Vector Learning, MIT Press, Cambridge, MA (1999) 255-268
5. Angulo, C., Parr, X., Català, A.: K-SVCR. A support vector machine for multi-class classification. Neurocomputing 55 (2003) 57-77
6. Zhong, P., and Fukushima, M.: A new multi-class support vector algorithm. Optimization Methods and Software, to appear (2004)
7. Allwein, E. L., Schapire, R. E., and Singer, Y.: . Reducing multiclass to binary: A unifying approach for margin classifiers. Journal of Machine Learning Research 1 (2001) 113-141
8. Dietterich, T. G., and Bakiri, G.: Solving multi-class learning problems via error-correcting output codes. Journal of Artificial Intelligence Research 2 (1995) 263-286
9. Graepel, T., Herbrich, R., Schoelkopf, B., and et al. Classification on Proximity Data with LP-Machines ICANN(1999)
10. Smola, A., Scholkopf, B., Rutsch, G.: Linear programs for automatic accuracy control in regression. In Ninth International Conference on Artificial Neural Networks , Conference Publications London 470 (1999) 575-580

11. Bennett, K. P.: Combining support vector and mathematical programming methods for classification. in: B. Schölkopf, C. J. C. Burges and A. J. Smola (Eds.), *Advances in Kernel Methods: Support Vector Learning*, MIT Press, Cambridge, MA (1999) 307-326
12. Crammer, K., Singer, Y.: On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research* 2 (2002) 265-292
13. Lee, Y., Lin, Y., Wahba, G.: . Multicategory support vector machines. *Computing Science and Statistics* 33 (2001) 498-512
14. Weston, J., Watkins, C.: Multi-class support vector machines. CSD-TR-98-04 Royal Holloway, University of London, Egham, UK (1998)
15. Graepel, T., et al.: Classification on proximity data with LP-machines: In: *Ninth International Conference on Artificial Neural Networks IEEE*. London: Conference Publications 470 (1999) 304-309
16. Blake, C.L., Merz, C.J.: *UCI Repository of Machine Learning Databases*. University of California, Irvine (1998) <http://www.ics.uci.edu/mlearn/MLRepository.html>

Support Vector Classification with Nominal Attributes^{*}

Yingjie Tian¹ and Naiyang Deng^{2,**}

¹ Chinese Academy of Sciences Research Center on Data Technology & Knowledge Economy, Beijing 100080, China

tianyijingjie1213@163.com

² College of Science, China Agricultural University, Beijing 100083, China
dengnaiyang@vip.163.com

Abstract. This paper presents a new algorithm to deal with nominal attributes in Support Vector Classification by modifying the most popular approach. For a nominal attribute with M states, we translate it into M points in $M - 1$ dimensional space with flexible and adjustable position. Their final position is decided by minimizing the Leave-one-out error. This strategy overcomes the shortcoming in the most popular approach which assume that any two different attribute values have the same degree of dissimilarities. Preliminary experiments also show the superiority of our new algorithm.

1 Introduction

Classification is one of the fundamental problems of machine learning. It can be described as follows: suppose that two classes of objects are given. We are then faced a new object, and have to assign it to one of the two classes. This problem is formulated mathematically : Given a training set

$$T = \{(x_1, y_1), \dots, (x_l, y_l)\} \in (R^n \times \{-1, 1\})^l, \quad (1)$$

where $x_i = ([x_i]_1, \dots, [x_i]_n)^T$ is called a input with the attributes $[x_i]_j, j = 1, \dots, n$, and $y_i = -1$ or 1 is called a corresponding output, $i = 1, \dots, l$. The question is , for a new input $\bar{x} = ([\bar{x}]_1, \dots, [\bar{x}]_n)^T$, to find its corresponding \bar{y} .

The attributes $[x_i]_j$ and $[\bar{x}]_j, j = 1, \dots, n$, are allowed to take either continuous values or nominal values. The classification with nominal attributes is studied in this paper.

2 Standard SVC with Continuous Attributes

Among the methods to deal with the classification with continuous attributes, Support Vector Classification (SVC) is very interesting because of its efficiency

^{*} This work is supported by the National Natural Science Foundation of China (No.10371131).

^{**} Corresponding author.

and its theoretical background [1]. The standard algorithm of SVC with continuous attributes is given as follows:

Algorithm 2.1. (*C-SVC*)

- (1) Given a training set $T = \{(x_1, y_1), \dots, (x_l, y_l)\} \in (R^n \times \{-1, 1\})^l$;
- (2) Select a kernel $K(\cdot, \cdot)$, and a parameter $C > 0$;
- (3) Solve problem

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l y_i y_j \alpha_i \alpha_j K(x_i, x_j) - \sum_{j=1}^l \alpha_j, \tag{2}$$

$$\text{s.t.} \sum_{i=1}^l y_i \alpha_i = 0, \tag{3}$$

$$0 \leq \alpha_i \leq C \quad i = 1, \dots, l, \tag{4}$$

and get its solution $\alpha^* = (\alpha_1^*, \dots, \alpha_l^*)^T$;

- (4) Compute the threshold b by KKT conditions, e.g. see [1], and construct the decision function as

$$f(x) = \text{sgn}\left(\sum_{i=1}^l \alpha_i^* y_i K(x, x_i) + b\right). \tag{5}$$

3 Most Popular Approach in SVC with Nominal Attributes

Now consider the training set (1) with nominal attributes. Suppose the input $x = ([x]_1, \dots, [x]_n)^T$, where the j -th nominal attribute $[x]_j$ take M_j states, $j = 1, \dots, n$. The most popular approach in SVC is as follows: Let R^{M_j} be the M_j -dimensional space. The j -th nominal attribute $[x]_j$ is represented as one of the M_j unit vectors in R^{M_j} . Thus the input space of the training set (1) can be embedded into a Euclidian space with the dimension $M_1 \times M_2 \times \dots \times M_n$, and every input x is represented by n unit vectors which belongs to the space $R^{M_1}, R^{M_2}, \dots, R^{M_{n-1}}$ and R^{M_n} respectively.

However, the above strategy has a severe shortcoming in distance measure. The reason is that it assumes that all attribute values are of equal distance from each other. The equal distance implies that any two different attribute values have the same degree of dissimilarities. Obviously this is not always to be preferred.

4 New SVC with Nominal Attributes

4.1 From Fixed Points to Flexible Points

Let us improve the above most popular approach by overcoming the shortcoming pointed out in the end of the last section.

We deal with the training set (1) in the following way. Suppose that the j -th nominal attribute $[x]_j$ takes values in M_j states

$$[x]_j \in \{v_{j1}, v_{j2}, \dots, v_{jM_j}\}, j = 1, \dots, n. \tag{6}$$

We embed the j -th nominal attribute $[x]_j$ into a $M_j - 1$ dimensional Euclidian space R^{M_j-1} : the first value v_{j1} corresponds to the point $(0, \dots, 0)^T$, the second value v_{j2} corresponds to the point $(\sigma_1^j, 0, \dots, 0)^T$, the third value v_{j3} correspond to the point $(\sigma_2^j, \sigma_3^j, 0, \dots, 0)^T, \dots$, and the last value v_{jM_j} corresponds to the point $(\sigma_{q_j+1}^j, \dots, \sigma_{q_j+M_j-1}^j)^T$, where $q_j = \frac{(M_j - 1)(M_j - 2)}{2}$. Therefore for the j -th nominal attribute $[x]_j$, there are p_j variables $\{\sigma_1^j, \sigma_2^j, \dots, \sigma_{p_j}^j\}$ to be determined, where

$$p_j = \frac{M_j(M_j - 1)}{2}. \tag{7}$$

In other words, for $j = 1, \dots, n$, the j -th nominal attribute $[x]_j$ corresponds to a matrix

$$\begin{pmatrix} v_{j1} \\ v_{j2} \\ v_{j3} \\ \vdots \\ v_{jM_j} \end{pmatrix} \longrightarrow \begin{pmatrix} 0 & 0 & 0 & \dots & 0 \\ \sigma_1^j & 0 & 0 & \dots & 0 \\ \sigma_2^j & \sigma_3^j & 0 & \dots & 0 \\ \dots & & & & \\ \sigma_{q_j+1}^j & \sigma_{q_j+2}^j & \sigma_{q_j+2}^j & \dots & \sigma_{p_j}^j \end{pmatrix} \triangleq H_j \in R^{M_j \times (M_j-1)}. \tag{8}$$

Suppose an input $x = ([x]_1, \dots, [x]_n)^T$ taking nominal value $(v_{1k_1}, v_{2k_2}, \dots, v_{nk_n})$, where k_j is the k_j -th value in $\{v_{j1}, v_{j2}, \dots, v_{jM_j}\}$. Then x corresponds to a vector

$$x \rightarrow \tilde{x} = ((H_1)_{k_1}, \dots, (H_n)_{k_n})^T, \tag{9}$$

where $(H_j)_{k_j}$ is the k_j -th row of $H_j, j = 1, \dots, n$. Thus the training set (1) turns to be

$$\tilde{T} = \{(\tilde{x}_1, y_1), \dots, (\tilde{x}_l, y_l)\} \tag{10}$$

where \tilde{x}_i is obtained from x_i by the relationship (9) and (8).

Obviously, if we want to construct a decision function based on the training set (10) by SVC, the final decision function depends on the positions of the above embedded points, in other words, depends on the set

$$\Sigma = \{\sigma_1^j, \sigma_2^j, \dots, \sigma_{p_j}^j, j = 1, \dots, n\}, \tag{11}$$

where p_j is given by (7).

4.2 New Algorithm

The values of $\{\sigma_1^j, \sigma_2^j, \dots, \sigma_{p_j}^j, j = 1, \dots, n\}$ in Σ in (11) can be obtained by learning. For example, it is reasonable to select them such that the LOO error for SVC is minimized, see, e.g. [3].

The definition of LOO error for Algorithm 2.1 is given as follows:

Definition 4.1. Consider Algorithm 2.1 with the training set (10). Let $f_{\tilde{T}|t}(x)$ be the decision function obtained by the algorithm from the training set $\tilde{T}|t = \tilde{T} \setminus \{(\tilde{x}_t, y_t)\}$, then the LOO error of the algorithm with respect to the loss function $c(x, y, f(x))$ and the training set \tilde{T} is defined as

$$R_{LOO}(\tilde{T}) = \sum_{i=1}^l c(\tilde{x}_i, y_i, f(\tilde{x}_i)). \tag{12}$$

In the above definition, the loss function is usually taken to be the 0 – 1 loss function

$$c(\tilde{x}_i, y_i, f(\tilde{x}_i)) = \begin{cases} 0, & y_i = f(\tilde{x}_i); \\ 1, & y_i \neq f(\tilde{x}_i). \end{cases} \tag{13}$$

Therefore, we investigate the LOO error with (13) below.

Obviously, the LOO error $R_{LOO}(\tilde{T})$ depends on the set (11)

$$R_{LOO}(\tilde{T}) = R_{LOO}(\tilde{T}; \Sigma). \tag{14}$$

The basic idea of our algorithm is: First, select the values in Σ by minimizing the LOO error, i.e. by solving the optimization problem:

$$\min_{\Sigma} R_{LOO}(\tilde{T}; \Sigma). \tag{15}$$

Then, using the learned values in Σ to train SVC again, and construct the final decision function. This leads to the following algorithm — New SVC with Nominal attributes (New SVCN):

Algorithm 4.2. (New SVCN)

- (1) Given a training set T defined in (1) with nominal attributes, where the j -th nominal attribute $[x]_j$ takes values in M_j states (6));
- (2) Introducing a parameter set $\Sigma = \{\sigma_1^j, \sigma_2^j, \dots, \sigma_{p_j}^j, j = 1, \dots, n\}$ appeared in (8) and turn T (1) to \tilde{T} (10) via (9) and (8);
- (3) Select a kernel $K(\cdot, \cdot)$ and a parameter $C > 0$;
- (4) Solve problem (15) with replacing T by \tilde{T} , and get the learned values $\bar{\Sigma} = \{\bar{\sigma}_1^j, \bar{\sigma}_2^j, \dots, \bar{\sigma}_{p_j}^j, j = 1, \dots, n\}$;
- (5) Using the parameter values in $\bar{\Sigma}$, turn T to $\bar{T} = \{(\bar{x}_1, y_1), \dots, (\bar{x}_l, y_l)\}$ via (10) and (9) with replacing “the wave \sim ” by “the bar $-$ ”.
- (6) Solve problem (2)–(4) with replacing T by \bar{T} and get the solution $\alpha^* = (\alpha_1^*, \dots, \alpha_l^*)^T$;
- (7) Compute the threshold b by KKT conditions, and construct the decision function as

$$f(\bar{x}) = \text{sgn}\left(\sum_{i=1}^l \alpha_i^* y_i K(\bar{x}_i, \bar{x}) + b\right). \tag{16}$$

where \bar{x} is obtained from x by the relationship (9) with replacing \tilde{x} by \bar{x} .

5 Numerical Experiments

In this section, the preliminary experiments on our algorithm are presented. As shown in Table 1, some standard data sets in [4] are tested by executing our algorithm. For every set, we split it into two parts, one is for training, and the other for testing.

Table 1. Data sets

Data set	#Nominal attributes	#Training patterns	#Test patterns
monks-1	6	300	132
monks-3	6	300	132
tic-tac-toe	9	500	458

When we choose RBF kernel function $K(x, x') = \exp(\frac{-\|x - x'\|}{2\delta^2})$, and choose $(\delta, C) = (0.1, 10)$, $(\delta, C) = (0.7, 10)$, $(\delta, C) = (3, 0.1)$, $(\delta, C) = (3, 100)$ respectively, we compare our algorithm (New SVCN) with the most popular approach using unit vectors (Popular SVCN) described in Section 3. The mean classification errors on testing sets are listed in Table 2. It is easy to see from Table 2 that our algorithm leads to smaller classification errors.

Table 2. Classification errors on testing set

Data set	Popular SVCN	New SVCN
monks-1	21.2%	18.9%
monks-3	19.7%	16.7%
tic-tac-toe	22.9%	19.87%

6 Conclusion

In this paper, we propose a new algorithm for SVC with nominal attributes, different from the most popular approach where the set of values of a nominal attribute with M states consists of M points in M dimensional space with fixed position. Our algorithm changes these M fixed points into M points in $M - 1$ dimensional Space with flexible and adjustable position. Following the idea of learning for the case of continuous attributes, we decide their final position by minimizing the LOO error. In this way, more precise results are expected. Preliminary experiments on real-world data sets have supported this conclusion. Another simplified version of dealing with the j -th nominal attribute with M_j states, $j = 1, \dots, n$, is also under our consideration. Here (8) is replaced by

$$\begin{pmatrix} v_{j1} \\ v_{j2} \\ v_{j3} \\ \vdots \\ v_{jM_j} \end{pmatrix} \longrightarrow \begin{pmatrix} \sigma_1^j & 0 & 0 & \dots & 0 \\ 0 & \sigma_2^j & 0 & \dots & 0 \\ \dots & & & & \\ 0 & 0 & 0 & \dots & \sigma_{M_j}^j \end{pmatrix} \triangleq H_j \in R^{M_j \times M_j}. \tag{17}$$

Furthermore, extend our approach to deal with the problem with both nominal and real attributes and the problem of feature selection are also under our consideration.

References

1. Schölkopf, B., Smola, A. J.: Learning with Kernels-Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press (2002)
2. Tsang, I.W., Kwok, J.T.: Distance metric learning with kernels, in: Proceedings of the International Conference on Artificial Neural Networks, Istanbul, Turkey, June (2003)
3. Vapnik, V., Chappelle, O.: Bounds on error expectation for support vector machines. *Neural Computation* 12 (2000)
4. Blake, C.L., Merz, C.J.: UCI Repository of Machine Learning Databases, University of California, Irvine, (1998). <http://www.ics.uci.edu/mlearn/MLRepository.html>

A New Smooth Support Vector Machine*

Yubo Yuan and Chunzhong Li

School of Applied Mathematics, University of Electronic Science and Technology of China,
Chengdu 610054, China
ybyuan@uestc.edu.cn

Abstract. In this paper, a new method that three-power spline function is used to smoothen the model of support vector machine (SVM) is presented. A third-order spline smooth support vector machine(TSSSVM) is obtained. Moreover, by analyzing the function precision, TSSSVM is better than SSVM and PSSVM.

1 Introduction

Finding rules by data observation to dope future data or non-observation data out is an orthodox learning method, but till now it has not an all-accepted theory frame. The solution to it may have three methods: Parameter Statistic Estimate Method, Experience Non-linear Method and Statistical Learning Theory.

Statistic Learning Theory is an expert machine learning theory to investigate small stylebook. It has built a new theory system for small stylebook statistic problem. The statistics inference techniques under the system not only consider the request of gradual capability, but also seek the optimal result under finite given information.

Yuh-Jye Lee and O.L.Mangasarian proposed to use signal function integral to get a smooth SSVM model in [5] in 2001. In [6], the author proposed two polynomial functions and got PSSVM model in January 2005 and it can be proved that PSSVM is more effective than SSVM.

In this paper we introduce a third-order spline function to smooth the object one of primal model and obtain the third-order spline smooth support vector machine model (TSSSVM). Moreover, it can be proved that TSSSVM model is better than ones are given in smooth precision and classification capability.

Now, we briefly outline the contents of the paper. In section 2 we will state the third-order spline function to construct the third-order spline smooth support vector machine (TSSSVM) model. In section 3 we mainly discuss the smooth property of the third-order spline function such as smooth capability and the approach degree to primal function. In section 4 we will prove the convergence of the TSSSVM model. In section 5, a conclusion is presented.

2 Third-Order Spline Smooth Support Vector Machine Model

In [5] and [6], the SVM problem is converted into an unconstrained optimal problem as follows,

* Supported by the Applied Basic Research Foundations of Sichuan (05JY029-068-2) and the Youth Key Foundations of Univ. of Electronic Science and Technology of China (JX04042).

$$\min_{\omega, \gamma} \frac{1}{2} \nu \| (e - D(A\omega - e\gamma))_+ \|_2^2 + \frac{1}{2} (\|\omega\|_2^2 + \gamma^2). \tag{1}$$

Here plus function $(\bullet)_+$ is described as follows: $(\mathbf{x})_+ = ((x_1)_+, (x_2)_+, \dots, (x_n)_+)^T$, in it

$$(x_i)_+ = \begin{cases} x_i & x_i \geq 0 \\ 0 & x_i < 0 \end{cases} \tag{2}$$

The unconstrained optimal model (1) is the object we will discuss. Its optimal objective function is a strongly convex but nonsmooth one, since the function $(\bullet)_+$ is sequential at the point $\mathbf{x} = 0$ but $\lim_{x \rightarrow 0^-} (x_+)' = 0, \lim_{x \rightarrow 0^+} (x_+)' = 1$, and it is nondifferentiable.

This property directly leads to the result that lots of good algorithms can't be used to solve the problem (1). Since those optimal algorithms require the objective function is first or quadratic differentiable, it is a feasible method to smooth the objective function by the use of fast algorithms. The discussion mainly has two aspects: the first is how to smooth the model and the second is to discuss the solution to it.

We propose a third-order spline function:

$$T(x, k) = \begin{cases} 0, & x < -\frac{1}{k} \\ \frac{k^2}{6}x^3 + \frac{k}{2}x^2 + \frac{1}{2}x + \frac{1}{6k}, & -\frac{1}{k} \leq x < 0 \\ -\frac{k^2}{6}x^3 + \frac{k}{2}x^2 + \frac{1}{2}x + \frac{1}{6k}, & 0 \leq x \leq \frac{1}{k} \\ x, & x > \frac{1}{k} \end{cases} \tag{3}$$

This function is used here to replace the plus function of (1) to obtain a third-order spline smooth support vector machine (TSSSVM):

$$\min_{(\omega, \gamma) \in R^{n+1}} \frac{\nu}{2} \| T(e - D(A\omega - e\gamma), k) \|_2^2 + \frac{1}{2} (\omega^T \omega + \gamma^2). \tag{4}$$

3 Analysis Capability of Smoothing Functions

In this section we will compare the smooth capability of the third-order spline function with the old ones.

Lemma 1^[6] $p(x, k)$ is defined as signal function integral in [5], x_+ is the plus function:

- (i) $p(x, k)$ is smooth for arbitrary rank about x .
- (ii) $p(x, k) \geq x_+$;
- (iii) for an arbitrary $\rho > 0$, $p(x, k)^2 - x_+^2 \leq (\frac{\log 2}{k})^2 + \frac{2\rho}{k} \log 2$ for $|x| < \rho$.

Lemma 2^[6] $q(x, k), f(x, k)$ are two polynomials defined in [6], x_+ is the plus function:

- (i) $q(x, k)$ is one rank smooth about x . $f(x, k)$ is twice rank smooth about x .
- (ii) $q(x, k) \geq x_+, f(x, k) \geq x_+$;
- (iii) for arbitrary x, k :

$$q(x, k)^2 - x_+^2 \leq \frac{1}{11k^2}, f(x, k)^2 - x_+^2 \leq \frac{1}{19k^2}.$$

Theorem 3. Let $x \in R, T(x, k)$ is defined as (3). x_+ is the plus function:

- (i) $T(x, k)$ is quadratic smoothness: at the point $x = \pm \frac{1}{k}, x = 0$ satisfy:

$$\left\{ \begin{array}{l} T(-\frac{1}{k}, k) = 0, \lim_{x \rightarrow 0^-} T(x, k) = \lim_{x \rightarrow 0^+} T(x, k), T(\frac{1}{k}, k) = \frac{1}{k}; \\ T'(-\frac{1}{k}, k) = 0, \lim_{x \rightarrow 0^-} T'(x, k) = \lim_{x \rightarrow 0^+} T'(x, k), T'(\frac{1}{k}, k) = 1; \\ T''(-\frac{1}{k}, k) = 0, \lim_{x \rightarrow 0^-} T''(-\frac{1}{k}, k) = \lim_{x \rightarrow 0^+} T''(0+, k), T''(\frac{1}{k}, k) = 0; \end{array} \right.$$

- (ii) $T(x, k) \geq x_+$;

- (iii) for arbitrary $x, k: T(x, k)^2 - x_+^2 \leq \frac{1}{24k^2}$.

The proof is omitted.

We can find from the comparison above that the capability of this approach is the best of all the given functions when k is fixed. The following picture will help to explain it when $k = 10$.

Through the picture we can easily find the smooth capability of the third-order spline function is the best.

4 Convergence Analysis of TSSVM

In this section, we will discuss the convergence of TSSVM model (4). If $T(x, k)$ is (3), we will prove that the model can approach to the primal model (1)'s optimal solution when $k \rightarrow +\infty$. What we need to say is that the smoothness of a vector is entry-wise.

Theorem 4. Let $A \in R^{m \times n}, b \in R^{m \times 1}$, define the real functions $h(x): R^n \rightarrow R$, and $g(x, k): R^n \times N \rightarrow R$ as follows:

$$h(x) = \frac{1}{2} \|(Ax - b)_+\|_2^2 + \frac{1}{2} \|x\|_2^2,$$

$$g(x, k) = \frac{1}{2} \|T(Ax - b, k)\|_2^2 + \frac{1}{2} \|x\|_2^2,$$

here $T(x, k)$ is defined as (3), then we have the following conclusion:

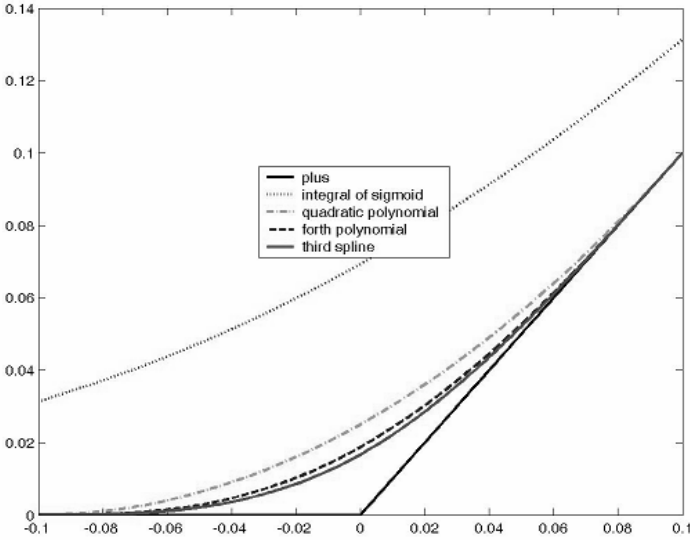


Fig. 1. Compared with Smoothing Functions. From above to below, the figures are integral function, quadratic polynomial function, forth polynomial function and third-spline function.

- (i) $h(x)$, $g(x,k)$ are strongly convex functions;
- (ii) Optimal problem $\min_x h(x)$ has a single solution x^* ; optimal problem $\min_x g(x,k)$ has a single solution $(x^*)^k$;
- (iii) For arbitrary $k \geq 1$, we have :

$$\|(x^*)^k - x^*\|^2 \leq \frac{m}{48k^2} ; \tag{5}$$

- (iv) $\lim_{k \rightarrow \infty} (x^*)^k = x^*$.

Proof: (i) It is clear that $h(x)$, $g(x,k)$ are strongly convex functions because of the strong convexity of $\|\bullet\|_2^2$.

(ii) In order to prove the existence of the optimal solutions to $\min_x h(x)$ and $\min_x g(x,k)$, notice that $T(x,k) \geq x_+$, and level sets $L_v(g(x,k))$ and $L_v(h(x))$ satisfy: for arbitrary $v \geq 0$, $L_v(g(x,k)) \subseteq L_v(h(x)) \subseteq \{x | \|x\|_2^2 \leq 2v\}$, therefore $L_v(g(x,k))$ and $L_v(h(x))$ are strict sets, so the conclusion is proved.

(iii) In order to prove the optimal solution to $\min_x g(x, k)$ converges to the optimal solution to $\min_x h(x)$ when $k \rightarrow +\infty$, according to the first rank optimization condition and the convex property of $h(x)$ and $g(x, k)$, we have :

$$h((x^*)^k) - h(x^*) \geq \nabla h(x^*)((x^*)^k - x^*) + \frac{1}{2} \|(x^*)^k - x^*\|_2^2 = \frac{1}{2} \|(x^*)^k - x^*\|_2^2,$$

$$g(x^*, k) - g((x^*)^k, k) \geq \nabla g((x^*)^k, k)(x^* - (x^*)^k) + \frac{1}{2} \|x^* - (x^*)^k\|_2^2 = \frac{1}{2} \|(x^*)^k - x^*\|_2^2,$$

With the combination of two formulas above and notice $f(x, k) \geq x_+$, we have:

$$\begin{aligned} \|(x^*)^k - x^*\|_2^2 &\leq (g(x^*, k) - h(x^*)) - (g((x^*)^k, k) - h((x^*)^k)) \\ &\leq (g(x^*, k) - h(x^*)) \\ &= \frac{1}{2} \|T(Ax^* - b, k)\|_2^2 - \frac{1}{2} \|(Ax^* - b)_+\|_2^2 \end{aligned}$$

Using theorem 3, the result $\|(x^*)^k - x^*\|_2^2 \leq \frac{m}{48k^2}$ can be obtained.

(iv) From (5) we have $\lim_{k \rightarrow \infty} \|(x^*)^k - x^*\|_2^2 \leq \lim_{k \rightarrow \infty} \frac{m}{48k^2} = 0$, and then $\lim_{k \rightarrow \infty} (x^*)^k = x^*$.

The proof is complete. □

5 Conclusion

This paper presents a new smoothing function for support vector machine. It is a third-order spline function that is more effective than presented methods for solving the SVM before. How to get a good smooth function is a difficult problem. The authors would believe that there must be a better function that can be used to get a good smoothing support vector machine.

References

- [1] Vapnik, V.. The Nature of Statistical Learning Theory. Springer. 1995.
- [2] Vapnik, V.. The support vector method of function estimation NATO ASI Series. Neural Network and Machine Learning, C. Bishop (Ed.), Springer, 1998.
- [3] Vapnik, V.. Three remarks on support vector function estimation. in Advanced in Kernel methods: Support Vector Learning, B. Scholkopf, B. Burges and A. Smola (Eds), The MIT Press, Cambridge, Massachusetts, 1999.
- [4] Vapnik, V.. An overview of statistical learning theory. IEEE transactions on Neural Networks, 1999, 10(5): 988-1000.
- [5] Yuh-Jye, Lee, Mangarasian, O. L.. SSVM: A smooth support vector machine for classification. Computational Optimization and Applications, 2001, 22(1): 5-21.
- [6] Y. Yuan, J. Yan, C. Xu. Polynomial Smooth Support Vector Machine. Chinese Journal of Computers. 2005. 28 (1):9-17.

- [7] Y. Yuan, T. Z. Huang, A polynomial smooth support vector machine for classification, ADMA05 (Proceeding of International Conference for Advance Data Mining and Application 2005). Lecture Notes in Artificial Intelligence. 2005,3584: 157-164.
- [8] Y. Yuan, T. Z. Huang, A matrix algorithm for mining association rules, ICIC05 (Proceeding of International Conference for Intelligent Computation 2005). Lecture Notes in Computer Science. 2005,3644: 370-379.
- [9] Mangasarian, O. L. , David R. Musicant. Successive overrelaxation for support vector machines. IEEE Transactions on Neural Networks, 1999, 10:1032-1037.

The Application of Support Vector Machine in the Potentiality Evaluation for Revegetation of Abandoned Lands from Coal Mining Activities

Chuanli Zhuang¹, Zetian Fu², Ping Yang¹, and Xiaoshuan Zhang^{3,*}

¹ College of Economics & Management, China Agricultural University,
Beijing 100083, China
{zhuangchuanli, yangping1999}@163.com

² Key Laboratory of Modern Precision Agriculture System Integration,
Ministry of Education, Beijing 100083, China
fzt@cau.edu.cn

³ College of engineering, China Agricultural University, Beijing 100083, China
zhxshuan@cau.edu.cn

Abstract. This paper presents the comparableness of SVM method to artificial neural networks in the outlier detection problem of high dimensions. Experiments performed on real dataset show that the performance of this method is mostly superior to that of artificial neural networks. The proposed method, SVM served to exemplify that kernel-based learning algorithms can be employed as an efficient method for evaluating the revegetation potentiality of abandoned lands from coal mining activities.

1 Introduction

The problem of abandoned land from mining activities is aggravated since the coal boom [1,2]. Activities of excavating coal excise natural vegetation and deposit stone on the natural land that modified the natural land contribute to waste farmlands. Additionally, pollutes environment, losses of human life, human settlements and the infrastructure are also rising, which certainly demands urgent attention.

It is not surprising that government agencies and local land-use planners have an interest in reducing the social and economic costs due to the abandoned land from coal mining activity. Therefore, it is desired to have a notion about the potentiality for revegetation of abandoned lands from coal mining activities, so that they can decide which areas should be revegetated firstly with lower cost and shorter time.

Several countries have attempted to discriminate areas potentiality for revegetation [3]. However, until now, systematic abandoned land from mining approaches using innovative technology to estimate potential sites are very scarce, particularly in China.

Although several techniques are available for the potentiality evaluation of abandoned land from mining investigation, a number of issues that affect the performance of these techniques have been identified, including the difficulties of handling continuous (i.e. numerical) and categorical data together. However, in the light of present

* Corresponding author.

knowledge, few research studies have attempted to consider the application of alternative techniques such as artificial neural networks in order to resolve these problems.

The paper aims mainly to develop a model based on support vector machines for evaluating the revegetation potentiality of the abandoned land from mining potentiality. The brief of the support vector machine (SVM) is described in Section 2. Sections 3 discuss the application of potentiality evaluation for abandoned land from coal mining activity. Some conclusions are presented in Section 4.

2 The Brief of Support Vector Machine

The SVM can be considered to create a line or hyperplane between two sets of data for classification [4-9]. SVM, originally designed for binary classification, employs supervised learning to find the optimal separating hyperplane between the two groups of data. Having found such a plane, SVM can then predict the classification of an unlabeled example by asking on which side of the separating plane the example lies. SVM acts as a linear classifier in a high dimensional feature space originated by a projection of the original input space, the resulting classifier is in general non-linear in the input space and it achieves good generalization performances by maximizing the margin between the two classes. In the following we give a short outline of construction of SVM [10].

Consider a set of training examples

$$\{(x_i, y_i)\}; x_i \in R^n, y_i \in \{+1, -1\}, i = 1, \dots, m \tag{1}$$

Where the x_i are real n -dimensional pattern vectors and the y_i are dichotomous labels.

SVM maps the pattern vectors $x \in R^n$ into a possibly higher dimensional feature space ($z = \phi(x)$) and construct an optimal hyperplane $w \cdot z + b = 0$ in feature space to separate examples from the two classes. For SVM with L1 softmargin formulation, this is done by solving the primal optimization problem

$$\min_{w, b, \xi_i} \frac{1}{2} \|w\|^2 + c \sum_{i=1}^m \xi_i \tag{2}$$

$$\text{s.t. } y_i(w \cdot z_i + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, 2, \dots, m \tag{3}$$

where c is a regularization parameter used to decide a tradeoff between the training error and the margin, and $\xi_i, i = 1, 2, \dots, m$, are slack variables.

The above problem is computationally solved using the solution of its dual form

$$\max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j k(x_i, x_j) \tag{4}$$

$$\text{s.t. } \sum_{i=1}^m \alpha_i y_i = 0, 0 \leq \alpha_i \leq C, i = 1, 2, \dots, m \tag{5}$$

where $k(x_i, x_j) = \phi(x_i) \bullet \phi(x_j)$ is the kernel function that implicitly define a mapping ϕ .

The resulting decision function is

$$f(x) = \text{sgn}\{\sum \alpha_i y_i k(x_i, x_j) + b\} \tag{6}$$

Any function that satisfies Mercer’s theorem [9] can be used as a kernel function for computing the dot product in feature space. There are different kernel functions such as linear, polynomial, Gaussian Laplacian RBF, and Sigmoid used in SVM.

3 Application

3.1 Data Description

The benchmark data set contains 46 instances, all of them are provided by the department of forestry, bureau of mining industry, Fuxin City, Liaoning Province. Every single instance consists of two types of Information (see Table 1). One is information regarding attribute values that contain abandoned land conditions such as Slope angle, Elevation, Topographic wetness index, Lineaments, Geological formations, Soil types, condition of traffic. The other information is about the evaluation value of abandoned land, which is target value and can be grouped into two classes:

- (1) Tractable
- (2) Non-tractable

Table 1. Attributes and evaluation value of the benchmark data set

Attributes	✧	Slope angle
	✧	Elevation
	✧	Topographic wetness index
	✧	Lineaments
	✧	Geological formations
	✧	Soil types
	✧	Condition of traffic
	Classes	✧
✧		Non-tractable

The purpose of SVM proposed in this paper is to construct a model that suggests target value of data instances in the testing set using only the given attributes. For training purposes, we random select 32 data instances to construct the SVM model. Testing (or ex post facto prediction) is made by rest data.

3.2 Configuration of SVM

SVM method has a lot of parameters to be set such as selection of kernel and the representation of the data. Moreover, it turns out to be surprisingly sensitive to

specific choices of representation and kernel. In order to produce robust results, it is very important to pay attention to set these parameters. Therefore, we conducted the experiments on these parameters (table 2).

Table 2. The Configuration of different parameters

Kernels	γ	r	d
Linear: $K(x_i \cdot x_j) = x_i^T \cdot x_j$	-	-	-
Polynomial: $K(x_i \cdot x_j) = (\mathcal{X}_i^T \cdot x_j + r)^d$	0.001,0.01, 0.1, 1, 10	0,1	2,3
Radial basis function (RBF): $K(x_i \cdot x_j) = \exp(-\gamma\ x_i - x_j\ ^2) \gamma > 0$	0.001,0.01, 0.1, 1, 10	-	-
Sigmoid: $K(x_i \cdot x_j) = \tan(\mathcal{X}_i^T \cdot x_j + r)$	0.001,0.01, 0.1, 1, 10	0,1	-

Here, γ , r , and d are kernel parameters to be set for a specific problem.

3.3 Results

One result of all experiment with different kernel, as shown in the fig.1, because the experiment with linear kernel without changes of parameter it's evaluation success rate is always 0.72, so it's curve doesn't display on fig.1, the RBF kernel gives the best results with $\gamma=1$,the evaluation success rate is up to 95% on test set.

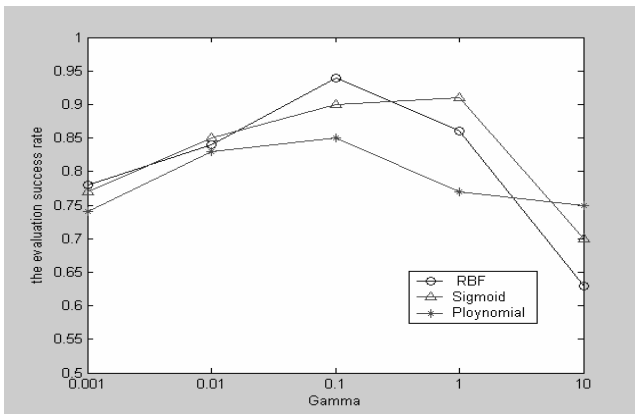


Fig. 1. The evaluation success rate of different kernel

We can also recognize from this validation process that it is important to choose the appropriate parameters. The optimal parameters on different kernels are summarized in Table 3.

We evaluated the performance for the proposed SVM with different types of kernel and compared it with that of Back-propagation model (an artificial neural network) in

Table 3. Optimal parameters for SVM with different kernel

Kernel	γ	r	d
linear			
polynomial	0.1	0	3
RBF	0.1		
Sigmoid	1	1	

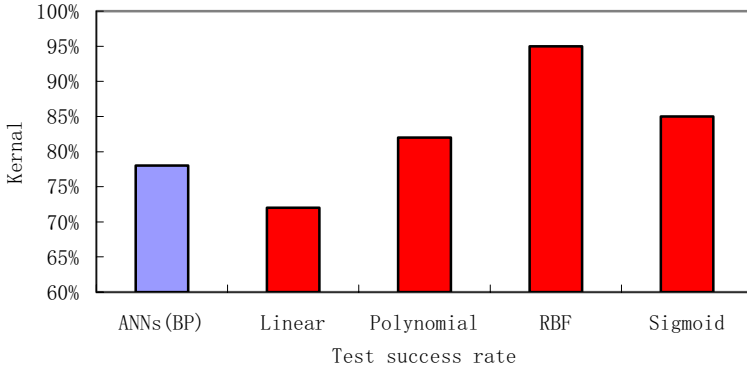


Fig. 2. Performance comparison with different kernels

Fig. 2. The figure shows test success rates that were achieved under each different method. As can be seen, SVM with RBF kernel has consistently given the best performance of all five methods. Examining the results, we can see that ANN manages to achieve 78% success on the test set. Compared to the performance of ANN, SVM gives relatively better results.

4 Conclusion

We have presented SVM that finds maximum margin hyperplanes in a high-dimensional feature space, emulating Vapnik’s SVM. The objective of this paper was to show the comparableness of SVM method to artificial neural networks in the outlier detection problem of high dimensions.

Experiments performed on real dataset show that the performance of this method is mostly superior to that of artificial neural networks.

We can use a variety of methods to evaluate the potentiality for revegetation of abandoned land from coal mining activity. The accuracy and fastness is the barometer of reducing the cost of maintenance and operating of revegetation. The proposed method, SVM served to exemplify that kernel-based learning algorithms can be employed as an efficient method for evaluating the revegetation potentiality of abandoned lands from coal mining activities.

Since SVM was shown to be sensitive to the parameters and the choice of kernel, however, more reinforced consideration should be followed in how to select appropriate parameters and kernel for SVM.

References

1. Wang Ying, Li Daoliang: A potentiality evaluation model for revegetation of abandoned lands from coal mining activities. *Journal of China Agricultural University*, 10 (2005)88 - 92
2. Z.Q. Zhang, M.H. Wong, X.P. Nie, C.Y. Lan: Effects of zinc (zinc sulfate) on rhizobia-earleaf acacia (*acacia auriculaeformis*) symbiotic association. *Bioresource Technology*, 64 (1998) 97-104
3. Catherine Neel, Hubert Bril, Alexandra Courtin-Nomade: Factors affecting natural development of soil on 35-year-old sulphide-rich mine tailings. *Geoderma*, 111(2003) 1-20
4. Bo-Suk Yang, Won-Woo Hwang, Dong-Jo Kim, Andy Chit Tan: Condition classification of small reciprocating compressor for refrigerators using artificial neural networks and support vector machines. *Mechanical Systems and Signal Processing*, 19 (2005) 371–390
5. Hyun Joon Shina, Dong-Hwan Eom, Sung-Shick Kim: One-class support vector machines—an application in machine fault detection and classification. *Computers & Industrial Engineering*, 48 (2005) 395–408
6. Osuna, E., Freund, R., & Girosi, F.: Training support vector machines: An application to face detection. In: 1997 Conference on computer vision and pattern recognition. Puerto Rico: IEEE. (1997) 130–136
7. L.B. Jack, A.K. Nandi: Fault detection using support vector machines and artificial neural networks. augmented by genetic algorithms. *Mechanical Systems and Signal Processing*, 16 (2–3) (2002) 373–390
8. V.N. Vapnik: An overview of statistical learning theory. *IEEE Transactions on Neural Networks* 10 (5) (1999) 988–999.
9. K.R. Muller, S. Mika, G. Ratsch, K. Tsuda, B. Scholkopf: An introduction to kernel-based learning algorithm, *IEEE Transactions on Neural Network* 12 (2) (2001) 181–201.
10. Jie Song, Huanwen Tang: Support vector machines for classification of homo-oligomeric proteins by incorporating subsequence distributions. *Journal of Molecular Structure: THEOCHEM*, 722 (2005) 97–101

Prediction of T-cell Epitopes Using Support Vector Machine and Similarity Kernel

Feng Shi^{1,2} and Jing Huang³

¹ School of Science, Huazhong Agricultural University,
Wuhan, China

² School of Mathematics and Statistics, Wuhan University,
Wuhan, China
shifeng@mail.hzau.edu.cn

³ School of Computer Science, Wuhan University,
Wuhan, China
caroline_hj@sohu.com

Abstract. T-cell activation is a pivotal process in immune response. A precondition for this activation is the recognition of antigenic epitopes by T-cell receptors. This recognition is antigen-specific. Therefore, identifying the pattern of a MHC restricted T-cell epitopes is of great importance for immunotherapies and vaccine design. In this paper, a new kernel is proposed to use together with support vector machine for the direct prediction of T-cell epitope. The experiment was carried on an MHC type I restricted T-cell clone LAU203-1.5. The results suggest that this approach is efficient and promising.

1 Introduction

Antigenic peptides degraded from foreign or host proteins can bind to major histocompatibility complex (MHC) molecules. The major role MHC plays is to present the binding antigenic peptides to T-cell receptors (TCRs). Only when the TCRs recognize the antigen, the T-cell clone will be activated, and the cellular immune will happen. However, not all the MHC-peptide complexes can be recognized by TCRs. Those portions of short binding peptides, which can be recognized, are called T-cell epitopes. The tri-complex of MHC molecule, antigenic peptide and TCR is of high specificity. Therefore, identifying the characteristic patterns of T-cell epitopes will help us understanding disease pathogenesis and etiology, and is helpful for therapeutics such as vaccine development.

T-cell epitopes are certainly MHC binders. So, most early identifications of T-cell epitopes relies on the prediction of peptides binding to MHC molecules. These mathematical approaches were based on binding motifs [1], quantitative matrices [2], artificial neural networks (ANNs) [3], and support vector machines (SVMs) [4], etc..

On the other hand, MHC binders are not always T-cell epitopes. Reseachers have developed methods for direct prediction of epitopes since 1980s. Methods based on machine-learning techniques have been primary methods on this area in recent years. For these machine-learning methods, the T-cell epitope samples inputted into the models must be transformed into reasonable numerical form. The frequently used

technique is to encode them with physical, chemical properties of amino acids. Researchers attempt to improve the prediction performance by combining several properties all along. That is a research focus and a difficult task as well.

In our work, we do not pay much attention to the discussion of peptide coding scheme, but focus on the custom design of kernel. A new kernel based on sequence similarity defined with the cross-correlation coefficient is investigated. Then the support vector machine approach with this new kernel is applied to T-cell epitopes prediction. The new method is evaluated on a data set employed by Zhao [5], and the results demonstrate a comparable performance of our method.

2 Material and Method

2.1 Dataset

The data set used in Zhao *et al.* 2003 is utilized in this experiment. These data can also be downloaded at http://linus.nci.nih.gov/Data/LAU203_Peptide.pdf. These 203 peptides were synthetic. Every peptide consisted of ten amino acid residues. They were tested against the Melan-A-specific CTL clone LAU203-1.5. Those peptides with percentage specific lysis higher than 10% were considered stimulatory, i.e., T-cell epitopes. Among these 203 peptides, 36 were tested stimulatory. The pairwise Pearson correlation coefficients for all stimulatory and non-stimulatory were calculated respectively. Only 5% in the positive group and 1% in the negative group have correlations larger than 0.6. This ensures that the peptides were sufficiently dissimilar for the cross-validation to be valid.

2.2 Support Vector Machine (SVM)

Support vector machine (SVM) is one type of learning machine based on statistical learning theory. It performs a nonlinear mapping of the input vector x from the input space R^d into a higher dimensional Hilbert space by a kernel function. It finds the optimal separating hyperplane corresponding to a non-linear boundary in the input space. A complete description to the theory of SVMs is in Vapnik's book [6].

2.3 Peptides Coding Scheme and New Kernel

In order to design the new kernel, we introduce a measurement of sequence-scale similarity based on a cross-correlation analysis of numerical representations. Cross-correlation is a standard method of estimating the degree to which two series are correlated. The forms of cross-correlation coefficient are various. Suppose $S_1(n)$ and $S_2(n)$ are two signal series with same length N , then a typical and simplest cross-correlation coefficient at delay j of them is described as follows.

$$\rho^{12}(j) = \frac{\gamma^{12}(j)}{\frac{1}{N} [\sum_{n=1}^N S_1^2(n) \sum_{n=1}^N S_2^2(n)]^{\frac{1}{2}}}, \quad j = 0, \pm 1, \pm 2, \dots \quad (1)$$

where j is the number of delay and $\gamma^{12}(j)$ is an estimate of the cross-covariance which is defined as:

$$\gamma^{12}(j) = \frac{1}{N} \sum_{n=1}^N S_2(n)S_1(n-j) \quad (2)$$

The correlation coefficient lies between -1 and +1. +1 means 100% correlation of two series in the same sense and -1 means the 100% correlation in the opposing sense [7]. Biomedical signals are considered strongly correlated with a correlation coefficient exceeds ± 0.7 and weakly correlated with a correlation coefficient between ± 0.7 and ± 0.5 . It has been showed that proteins with common function generally have a strong cross-correlation, while proteins without similar biological function only have a weak correlation. Cross-correlation analysis has ever been used together with Fourier transform or wavelet transform to investigate the sequence similarity in frequency domain [8]. In our experiment, we intend to use the cross- correlation coefficients without delay to define the similarity kernel.

First, we convert the peptides into numerical vectors, which are called representation vectors. Hydrophobicity is one of the most representative and important properties of amino acid influencing sequence structure and function. It is very useful for many researches, especially for T-cell epitope identification. So, in our coding scheme, each amino acid is substituted with its hydrophobicity value (Argos et al., 1982) which is extracted from the AAindex.

Suppose s_i and s_j are two representation vectors, their similarity score is defined as

$$k(s_i, s_j) = \frac{\sum_{n=1}^N s_i(n)s_j(n)}{\sum_{n=1}^N s_i^2(n) \sum_{n=1}^N s_j^2(n)} \quad (3)$$

where N is the uniform length of representation vectors. Then the elements of kernel matrix in SVM are defined as the pairwise score of two peptides.

3 Results and Discussions

We use the matlab and a toolbox of machine learning method called Spider (freely available at <http://www.kyb.tuebingen.mpg.de/bs/people/spider/>) to execute our experiments. Spider is an implementation of SVM using custom design kernel.

Since the sizes of positive and negative sets are quite unbalance, we evaluate the performance of our prediction method with sensitivity and PPV (positive prediction value). Denote tp , tn , fp and fn as the numbers of correctly predicted positive, correctly predicted negative, wrongly predicted positive and wrongly predicted negative, respectively. Then sensitivity is defined as $tp/(tp + fn)$ and PPV is $tp/(tp + fp)$. PPV is an important criterion that reflects the efficiency of a method.

Table 1. Results obtained from the new kernel with hydrophobicity coding

fold	tp	tn	fp	Fn	Sensitivity	PPV
1	2	16	0	3	2/5	2/2
2	4	16	1	0	4/4	4/5
3	2	17	1	1	2/3	2/3
4	5	13	1	1	5/6	5/6
5	1	18	0	1	1/2	1/1
6	2	16	1	1	2/3	2/3
7	1	18	1	0	1/1	1/2
8	4	14	1	1	4/5	4/5
9	1	17	2	0	1/1	1/3
10	4	14	0	2	4/6	4/4

Table 2. Comparison of results to other methods

Method	Sensitivity	PPV
our method	0.722	0.765
SVM (Zhao)	0.763	0.716
ANN	0.555	0.817
Decision tree	0.463	0.867
Score matrix ^a	1.000	0.343
SYFPEITHI	0.863	0.348
SVMHC (a) ^b	0.306	0.458
SVMHC (b) ^c	0.389	0.451

Remark ^aThe analysis was based on an approach using a Z-matrix as described in [2].

^bThe SVM model was trained based on A*0201 restricted MHC binding data from SYFPEITHI database. ^cThe SVM model was trained based on A*0201 restricted MHC binding data from MHCPEP database.

Table 1 shows the 10-fold-cross-validation results of our method using the new similarity kernels. The average cross-validation sensitivity and PPV are 72.2% and 76.5% respectively for 10 test sets. And the average accuracy is 91.1%. Accuracy is the ratio of $tp + tn$ to $tp + tn + fp + fn$.

We also compare our method with several others in table 2. The results for other methods are extracted from Zhao *et al.* 2003. We may see that our method is efficient. First, it is obvious that the performance of our method is more equitable. Its sensitivity and PPV are all higher than 70%. That is a similar performance with SVM(Zhao). Secondly, our method has a higher PPV than most other methods except ANN and Decision tree, while these two methods perform much worse on sensitivity. Last, even considered from the sensitivity, our method is still comparable to others.

To verify the efficiency of our new kernel ulteriorly, we do another experiment besides. As we all know, the pairwise alignment score is sometimes considered a measurement of sequence similarity. Therefore, we define a similarity kernel using the pairwise alignment score based on PAM250. The prediction results of SVM model

with this kernel are listed as follows. The sensitivity averaged over 10 folds is 94.4% while the average PPV is 31.8%. The high sensitivity and excessively low PPV maybe indicate that the alignment give all pairs of peptides similar scores and cannot reflect their nuance in biological function. These results prove that the kernel defined with cross-correlation coefficient is appropriate for representing sequence similarity, and that our method is efficient.

We also tested some other properties of amino acid, but the results are not as good as the Hydrophobicity index.

4 Conclusion

Prediction of T-cell epitopes is of great significance. Kernel based learning methods are advanced machine learning methods. The advantage of them is that they allow researchers to define kernel function according to their expert knowledge, so that the kernel will have more significations from the view of their studing domain. In this paper, we introduced a novel kernel based on cross-correlation analysis. Combining this kernel with support vector machine, we get a good result for T-cell epitopes prediction. The deficiency of our method is its relatively lower sensitivity. That maybe impute to the effect of the unbalance of positive and negative sample size. Given sufficient positive data, it is expected that this problem will be solved in future. Taking one with another, the SVM with our new kernel is efficient and promising.

References

1. Rammensee, H.G., *et al.*: MHC ligands and peptide motifs, first listing. *Immunogenetics*, Vol. 41. (1995) 178-228
2. Zhao, Y., *et al.*: Combinatorial peptide libraries and bio-metric score matrices permit the quantitative analysis of specific and degenerate interactions between clonotypic T-cell receptors and MHC-peptide ligands. *Journal of Immunology*, Vol. 167. (2001) 2130-3141
3. Brusic, V., *et al.*: Prediction ofMHCclass II-binding peptides using an evolutionary algorithm and artificial neural network. *Bioinformatics*, Vol. 14. (1998) 121-130
4. Dönnes, P., Elofsson, A.: Prediction of MHC I binding peptides, using SVMHC. *BMC Bioinformatics*, Vol. 3. (2002) 1-8
5. Zhao, Y., *et al.*: Application of support vector machines for T-cell epitopes prediction. *Bioinformatics*, Vol. 19. (2003) 1978-1984
6. Vapnik, V.: *Statistical Learning Theory*. Wiley-Interscience, New York (1998)
7. Oppenheim AV., Schafer RW. In *Discrete-time Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ (1997)
8. Hejase de Trad, *et al.*: Protein sequence comparison based on the wavelet transform approach. *Protein Eng.*, Vol. 15. (2002) 193-203

Radial Basis Function Support Vector Machine Based Soft-Magnetic Ring Core Inspection

Liangjiang Liu and Yaonan Wang

College of Electrical and Information Engineering, Hunan University,
Changsha 410082, China
Llj_cn@sohu.com, wang_yaonan@hotmail.com

Abstract. A Soft-magnetic ring cores (SMRC) inspection method using radial basis function support vector machine (RBFSVM) was developed. To gain the effective edge character of the SMRC, a sequence of image edge detection algorithms was developed. After edge was detected, feature vector was extracted. Subsequently, principal component analysis (PCA) is applied to reduce the dimension of the feature vector. Finally, RBFSVM is used for classification of SMRC, whose best accuracy in experiments is 97%.

Keywords: soft-magnetic ring core (SMRC), radial basis function support vector machine (RBFSVM), edge detection, principal component analysis (PCA).

1 Introduction

Soft-magnetic ring cores (SMRC) are a vital part of electrical appliances such as chokes, in-line filters, voltage and current transformers [1]. To obtain the desired production quality, its dimension should be inspected carefully because high temperature agglomeration lead to distortion. In the soft-magnetic materials industry, the quality evaluation of SMRC is still performed manually by trained inspectors, which is tedious, laborious, and costly, and is easily influenced by physiological factors, thus inducing subjective and inconsistent evaluation results. Increased demand of SMRC have necessitated the automated quality evaluation of SMRC.

Utilize machine vision technique to substitute human's eyes, to take pictures of SMRC and to inspect the product, which greatly enhance the product quality and the production efficiency. Machine vision is that the equipment with computer is used to realize human's vision function, i.e. computer is used to recognize the object things. With the ceaseless development of machine vision technique, human gradually applied this technique to many aspects of manufacture and life [2], [3], [4].

2 Materials and Methods

The samples of soft-magnetic ring core were often categorised into five quality levels by the qualified inspection personnel in the company, i.e., reject oversize, acceptable oversize, precision size, acceptable undersize and reject undersize. The overall sequence of digital image processing algorithms for classification of SMRC mainly

include five steps: a, image acquisition; b, edge detection; c, feature vector extraction; d, reduction dimension by PCA; e, RBF SVM classification.

2.1 Edge Detection

To get the edge of SMRC, a gradient-based detection approach was developed, which includes three steps, i.e., edge detection, morphological dilation and edge slenderization. For the SMRC image that differs greatly in contrast from the background, the Roberts operator was applied to detect the edge of SMRC. To eliminate the gaps in the edges, morphological dilation was implemented to the edge image. Then the edge was slenderized to one pixel width. The result of edge extraction was shown in Fig.1.

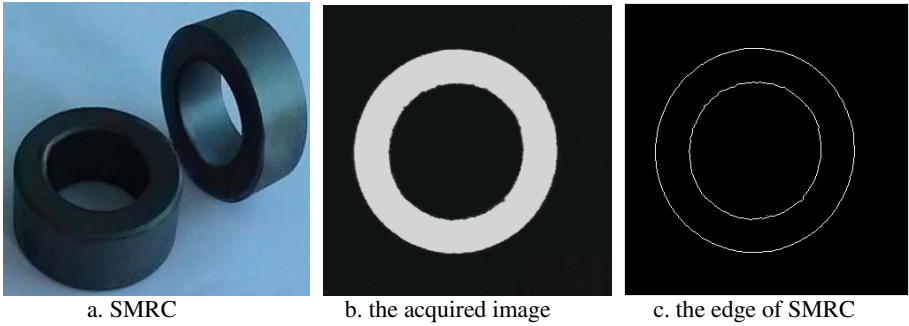


Fig. 1. Edge extraction

2.2 Feature Vector Extraction

To get the centre of SMRC, a supposed center (x_0, y_0) was put forward. The (xa_i, ya_i) was supposed to be the point of the outer edge and m is the point quantity of the outer edge. The (xb_j, yb_j) was supposed to be the point of the inner edge and n is the point quantity of the inner edge. The ra and rb was the desired outer radius and inner radius respectively of SMRC. The da_i and db_j was the distance from the outer edge point and the inner edge point respectively to the supposed centre.

The optimal centre of SMRC is the one that minimizes the following formula,

$$F = \sum_{i=1}^m da_i^2 + \sum_{j=1}^n db_j^2 . \tag{1}$$

The centre can be obtained by solving a quadratic equation extremum problem. Let

$$\begin{cases} \frac{\partial F}{\partial x_0} = \sum_{i=1}^m 2(x_0 - xa_i) + \sum_{j=1}^n 2(x_0 - xb_j) = 0 \\ \frac{\partial F}{\partial y_0} = \sum_{i=1}^m 2(y_0 - ya_i) + \sum_{j=1}^n 2(y_0 - yb_j) = 0 \end{cases} . \tag{2}$$

The centre of SMRC was as follows,

$$\begin{cases} x_0 = \frac{1}{m+n} \left(\sum_{i=1}^m xa_i + \sum_{j=1}^n xb_j \right) \\ y_0 = \frac{1}{m+n} \left(\sum_{i=1}^m ya_i + \sum_{j=1}^n yb_j \right) \end{cases} \quad (3)$$

The error da_i' between da_i and desired radius ra was calculated as below,

$$da_i' = da_i^2 - ra^2, \quad i = 1, 2, \dots, m. \quad (4)$$

The error db_j' between db_j and desired radius rb was calculated as below,

$$db_j' = db_j^2 - rb^2, \quad j = 1, 2, \dots, n. \quad (5)$$

The da_i' and db_j' was arranged descendingly, where the collection $\{da_1', da_2', \dots, da_{10}'\}$ and $\{db_1', db_2', \dots, db_{10}'\}$ represented the ten biggest number of da_i' and db_j' respectively, the collection $\{da_{11}', da_{12}', \dots, da_{20}'\}$ and $\{db_{11}', db_{12}', \dots, db_{20}'\}$ represented the ten smallest number of da_i' and db_j' respectively. The feature vector x_k was composed as follows,

$$x_k = [da_1' \quad da_2' \quad \dots \quad da_{20}' \quad db_1' \quad db_2' \quad \dots \quad db_{20}']^T, k = 1, 2, \dots, l. \quad (6)$$

where l is the images quantity of SMRC.

2.3 Dimensionality Reduction by Principal Component Analysis

In real implementation, the quantified 40-dimensional vectors x_k are still too large to allow fast and accurate classification. The large feature vectors will increase the complexity of the classifier and the classification error. PCA is one of the powerful techniques for dimensionality reduction, which transforms original feature vectors from large space to a small subspace with lower dimensions.

The basic approach of PCA is first to compute the covariance matrix C_x of the quantified 40-dimensional vectors, where $E(x_k)$ is the expectation value of x_k .

$$C_x = E \left\{ [x_k - E(x_k)] [x_k - E(x_k)]^T \right\}, k = 1, 2, \dots, l, x = 1, 2, \dots, l. \quad (7)$$

The covariance matrix C_x can be denoted as follows,

$$C_x = U \Lambda U^T, U = [u_1 u_2 \dots u_{40}], \Lambda = \text{diag} \{ \lambda_1, \lambda_2, \dots, \lambda_{40} \}. \quad (8)$$

where $\{\lambda_1, \lambda_2, \dots, \lambda_{40}\}$ is the eigenvalue of C_x and $\{u_1, u_2, \dots, u_{40}\}$ is the corresponding eigenvector. There are two ways to compute the eigenvalues and eigenvectors: singular value decomposition (SVD) and regular eigen-computation.

The first principal component accounts for the most significant characteristic of the original data with the maximum variance. Each succeeding component accounts for less significant characteristic with as much of the remaining variability as possible. Practically, the last few principal components can be truncated from the back of the transformation matrix. These principal components correspond to useless characteristics that are essentially noise. Taken the first 10 eigenvectors as its rows, a transformation matrix p is formed. The new ring feature vectors x_i are obtained by

$$x_i = p^T x_k, \text{ where } p = [u_1 u_2 \dots u_{10}], \quad i = k = 1, 2, \dots, l. \tag{9}$$

which maps the quantified 40-dimensional vectors to a smaller vectors.

2.4 Support Vector Machine

The classification of SMRC into acceptable and unacceptable quality levels can be considered as a binary categorization problem. Suppose there are l samples of SMRC in the training data, and each sample is denoted by a vector x_i , which represents the ring features of the SMRC. The classification of SMRC can be described as the task of finding a classification function $f : x_i \rightarrow y_i, y_i \in \{-1, +1\}$ using training data. Subsequently, the classification function f is used to classify the unseen test data. If $f(x_i) > 0$, the input vector x_i is assigned to the class $y_i = +1$, i.e., the acceptable quality level, otherwise to the class $y_i = -1$, i.e., the unacceptable quality level.

The classification has the following formula,

$$f(x_i) = \text{sgn} \left\{ \sum_{i=1}^l \alpha_i y_i K(x_i, x) + b \right\}. \tag{10}$$

where α_i is the Lagrange coefficient corresponding to every example, $K(x_i, x)$ is the inner product function., b is the bias term, x_i was input character and y_i was the output result of classification [5], [6], [7].

α_i should satisfy the following conditions:

$$\sum_{i=1}^n y_i \alpha_i = 0, \text{ where } \alpha_i \geq 0, i = 1, \dots, n. \tag{11}$$

α_i can be obtained by solving the maximum of the following function:

$$Q(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) . \tag{12}$$

which is a convex quadratic programming problem subject to linear constraints, and where there is unique answer. It is easy to prove that there are only a few α_i which are nonzero and whose corresponding samples are support vectors.

The $K(x_i, x)$ adopt radius based function, which has the following form:

$$K(x_i, x) = \exp \left\{ - \frac{|x - x_i|^2}{\sigma^2} \right\} . \tag{13}$$

where σ is the constant.

The bias term b can be obtained by solving the following formula:

$$y_j \left\{ \sum_{i=1}^l \alpha_i y_i K(x_i, x_j) + b \right\} - 1 = 0 . \tag{14}$$

where x_j is anyone of support vectors, y_j is the corresponding output result.

3 Results and Discussion

In this study, 200 images of SMRC were captured for classification, including 100 acceptable levels and 100 unacceptable levels. The feature vector extraction of SMRC described above was implemented by visual basic under Windows 2000 on a IBM Graphics Workstation E-PRO. Half of these images were used to train the parameters of support vector machine, the rest were adopted to test the support vector machine.

Radius based function support vector machine was trained using SMRC ring character vector x_i . The SVM classification function figures were chosen to demonstrate the performance of the SVM as shown in Fig.2. The crosses represent class +1, i.e. acceptable SMRC, and squares represent class -1, i.e. unacceptable SMRC. The black line represents classification line. Adopting different constant σ result in different classification curve. With the aggrandizement of constant σ , the classification curve become smoothness.

The following table is the quantity of support vector and the preciseness of classification adopting different constant σ . Adopting different constant result in different quantity of support vector. Then the rest 100 images was used to test the classification preciseness and it lead to different preciseness of classification. On the experiment, the RBFSVM with $\sigma = 144$ result in the best classification rate of 97%.

Table 1. The different quantity of support vector and the corresponding classification preciseness

Serial number	1	2	3	4	5	6	7
constant σ	1	8	36	64	100	144	200
the quantity of support vector	91	51	25	13	7	7	7
the preciseness of classification	63%	78%	85%	92%	96%	97%	96%

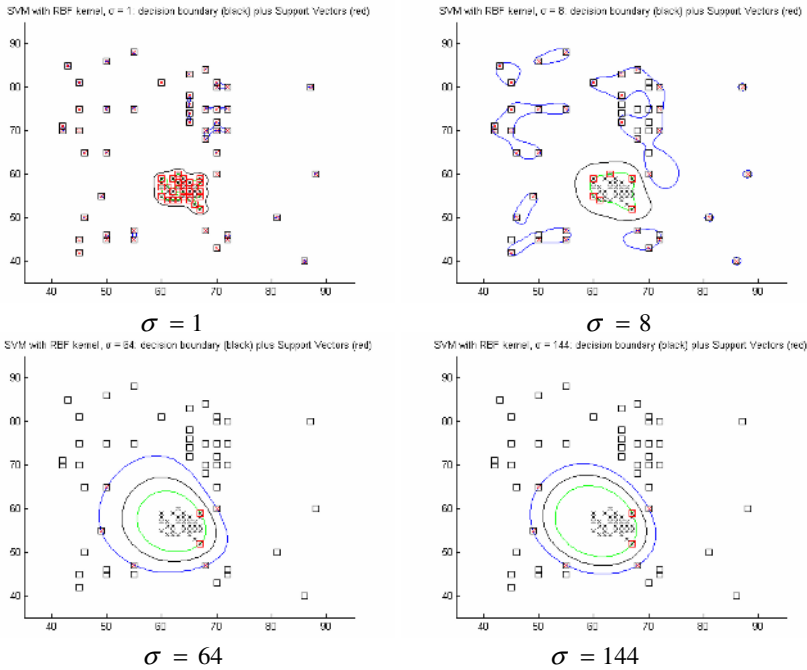


Fig. 2. Classification figure adopting different σ constant

4 Conclusions

The results presented here have demonstrated the ability of the approach based on radial basis function support vector machine to classify SMRC. The PCA techniques successfully reduced the dimensionality of ring features vector . With the first 10 principal components as the input, an overall accuracy of 97% was achieved by the RBFSVM classifiers with $\sigma = 144$.

References

1. Marko Petkovsek, Janez Nastran, Peter Zajec, France Pavlovic, Danijel Voncina. Soft-Magnetic Ring Core Measuring System With a Decreased Number of Primary and Secondary Winding Turns. IEEE Transactions on Instrumentation and Measurement, 2004, VOL. 53, NO. 2:444-447

2. Cheng-Jin Du, Da-Wen Sun. Pizza sauce spread classification using colour vision and support vector machines. *Journal of Food Engineering*, 2005,66:137–145
3. J. Blasco, N. Aleixos, E. Molto. Machine Vision System for Automatic Quality Grading of Fruit. *Biosystems Engineering*, 2003,85(4):415–423
4. D. Earl Kline, Chris Surak, Philip A. Araman. Automated hardwood lumber grading utilizing a multiple sensor machine vision technology. *Computers and Electronics in Agriculture*, 2003,41:139–155
5. Rezaul K. Begg, Marimuthu Palaniswami, Brendan Owen. Support Vector Machines for Automated Gait Classification. *IEEE Transactions on Biomedical Engineering*, 2005, 52(5):828–838
6. Amit David, Boaz Lerner. Support vector machine-based image classification for genetic syndrome diagnosis. *Pattern Recognition Letters*, 2005,26:1029–1038
7. Yiqiang Zhan, Dinggang Shen. Design efficient support vector machine for fast classification. *Pattern Recognition* 2005,38:157–161

Liangjiang Liu received the B.S. degree in industrial automation from Xiangtan University (XTU) in 1999 and the M.S. degrees in metallurgy automation from Changsha Research Institute Of Mining And Metallurgy (CRIMM), Changsha, China, in 2002. From 2002 to 2004, he worked with the CRIMM. Since 2004, he has been a doctor graduate student at Hunan University. His research interests are machine vision, pattern recognition, and intelligent control.

Yaonan Wang received the B.S. degree in computer engineering from East China Technology Institute (ECTI) in 1981 and the M.S. and Ph.D. degrees in electrical engineering from Hunan University, Changsha, China, in 1990 and 1994, respectively. From 1994 to 1995, he was a Postdoctoral Research Fellow with the National University of Defense Technology. From 1981 to 1994, he worked with the ECTI. Since 1995, he has been a Professor at Hunan University. His research interests are image processing, pattern recognition, and industrial process control.

Direct Adaptive NN Control of a Class of Feedforward Systems

Wang Dongliang, Liu Bin, and Zhang Zengke

Tsinghua University, Beijing, China

Abstract. In this paper, direct adaptive neural-network (NN) control is presented for a class of affine nonlinear systems in the strict-feedforward form with unknown nonlinearities. The states of the system can be stabilized. The control performance of the closed-loop system is guaranteed by suitably choosing the design parameters. Simulation results show the effectiveness of direct adaptive NN control.

1 Introduction

IN recent years, adaptive neural control schemes based on Lyapunov's stability theory have been found to be particularly useful for the control of nonlinear systems with unknown nonlinear functions. One main advantage of these schemes is that the parameter adaptation laws are derived based on Lyapunov synthesis and therefore stability of the closed-loop system is guaranteed. However, one limitation of these schemes is that they can only be applied to nonlinear systems where certain types of matching conditions are required to be satisfied. To avoid it, adaptive NN control is mostly used in feedback systems [1][7][2]. There is a constructive design procedure [6], called backstepping. If the feedback system is known, or the basis function of the systems is known [4], the stabilizing controller can be constructed. Therefore, when the system has unknown nonlinearity, NN and backstepping can be used together to get desired effectiveness. Since the feedback systems have the upper-triangular structure, we can design the adaptive NN controller if we know the structural character of the systems.

There is another constructive procedure, called forwarding, which can be used to stabilize the strict feedforward systems. By the procedure of forwarding, the stabilizing controller can be constructed without the coordinate change [6][5][4]. In this paper, adaptive control and adaptive NN control of a class of strict feedforward systems will be given. The class of systems has the formation as

$$\begin{aligned}\dot{x}_1 &= f(x_2) + u \\ \dot{x}_2 &= -x_2 + u\end{aligned}\tag{1}$$

The rest of the paper is organized as follows: The problem formulation is presented in Section II. In Section III, a direct adaptive controller is presented for controlling uncertain nonlinear systems. Simulation results show the effectiveness of the approach. In Section IV, a direct adaptive NN controller is presented for

controlling uncertain nonlinear system .Simulation results performed on an illustrative example are demonstrated to show the effectiveness of the approach. Section IV contains the conclusion.

2 Problem Formulation

We consider a nonlinear system represented globally by the equation (1), where $x_i \in R$ is state, $f(0) = 0$ is an unknown function, and $u \in R$ is input. The objective is to find a stabilizing controller of the systems which make the state asymptotically stable(AS). First we will give the adaptive design when the basis functions of f is known. Then we will use neural networks to design the adaptive controller.

3 Adaptive Forwarding

3.1 Controller Design

Consider the system (1), where $f = \theta^T \phi(x_2)$ is an unknown function, $\phi : R \rightarrow R^n, \theta \in R^n$ is known, and $\theta \in R^n$ is an unknown vector. Then, the objective is to design the stabilizing controller of the feedforward systems using adaptive control. we can constructive the Lyapunov function using forwarding technology [6]. First, we can take the Lyapunov function candidate V_0 as

$$\begin{aligned}
 V_0 &= \frac{1}{2} \left(x_2^2 + \left(x_1 + \int_0^\infty f_1(x_2 e^{-s}) ds \right)^2 \right) \\
 &= \frac{1}{2} \left(x_2^2 + \left(x_1 + \theta^T \int_0^\infty \psi(x_2 e^{-s}) ds \right)^2 \right) \tag{2}
 \end{aligned}$$

When the parameter vector is constant, we can get the derivative of V with respect to time.

$$\begin{aligned}
 \dot{V} &= -x_2^2 + \left(x_2 + x_1 + \theta^T \int_0^\infty \psi(x_2 e^{-s}) ds \right. \\
 &\quad \left. + \left(x_1 + \theta^T \int_0^\infty \psi(x_2 e^{-s}) ds \right) \frac{\partial \int_0^\infty \psi(x_2 e^{-s}) ds}{\partial x_2} \right) u \tag{3}
 \end{aligned}$$

If the parameter vector θ is known, it is easy to design the stabilizing controller by forwarding proposed in [6]. When the parameter vector θ is unknown , we will use the notation $\hat{\theta}$ to denote the estimation of θ and develop an adaptive law to update the parameter .

Theorem 1. *Consider the system (1), the stabilizing controller u is as follows.*

$$u = -L_G V_0(x_1, x_2, \hat{\theta}, \hat{\theta}_1) \tag{4}$$

and the regular law of $\theta, \hat{\theta}_1$ is as follows.

$$\dot{\theta} = \frac{\partial \psi_1}{\partial x} \begin{bmatrix} 1 \\ 1 \end{bmatrix} u; \quad \dot{\hat{\theta}}_1 = \left(\frac{\partial \psi_2}{\partial x_2} \right)^T u \tag{5}$$

Proof. Substitute (4) and (5), we can get

$$\dot{V} = -x_2^2 - u^2 \tag{6}$$

we can obtain that $\dot{V} < 0$ when $[x_1, u] \neq 0$. Therefore, the state are Global stable (AS). If we want to get the GAS property of the systems, it need to suppose that the invariant set of $u(x_1, 0, \hat{\theta}, \hat{\theta}_1) = 0$ is only include the origin.

3.2 Simulation

Here, a simple simulation is presented to show the effectiveness of the approach proposed above. The model of the system is given as

$$\begin{aligned} \dot{x}_1 &= 0.1x_2 + 5x_2^2 + 10 \sin(x_2)^3 + u \\ \dot{x}_2 &= -x_2 + u \end{aligned} \tag{7}$$

where x_1 and x_2 are states. The initial conditions is $[x_1, x_2]^T = [0.3, 0.55]^T$.

Let $\phi(x_2) = [x_2, x_2^2, \sin(x_2)^3]^T$, first we obtain ψ_1, ψ_2 , then we can obtain the control law and the update law according to (4) and (5). Take the initial value of parameter as $\theta(0) = [0.5, 0.5, 0.5]^T, \theta_1(0) = [0.5, 0.5, 0.5, 0.5, 0.5, 0.5]^T$, and get the simulation result as Fig 1.

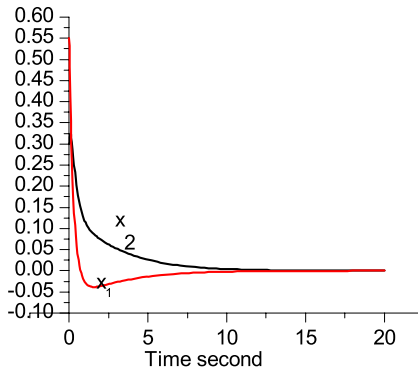


Fig. 1. The state of the system under adaptive controller

Remark 1. If the initial value of above example became large, the state of the systems may be stable but not asymptotically stable.

4 Adaptive Forwarding Using Neural Networks

4.1 Controller Design

Consider the system (1), suppose f_1 is an unknown function, and satisfies $f_1(0) = 0$. The objective is to design the stabilizing controller of the feedforward systems using adaptive NN control. We can construct the Lyapunov function using forwarding technology and neural networks approximation theory. In this paper, we will use RBF networks to approximate the unknown function f_1 . According to the approximation property of the RBF networks, given a continuous real valued function $f : \Omega \rightarrow R$ with a compact set $\Omega \in R^n$, and any δ_m , by appropriately choosing $\zeta_j, \delta_j \quad j = 1, \dots, n$, for some sufficiently large integer N , there exists an ideal weight vector θ^* such that the RBF network can approximate the given function f with the approximation error bounded by δ_m , i.e.,

$$f(x) = \theta^{*T} \xi(x) + \delta^*, X \in \Omega \tag{8}$$

with $\delta^* \leq \delta_m$, where δ^* represents the network reconstruction error δ^* , i.e., $\delta^* = f(x) - \theta^{*T} \xi(x)$. Since θ^* is unknown, we need to estimate it online. We will use the notation $\hat{\theta}$ to denote the estimation of θ^* and develop an adaptive law to update the parameter. To avoid the derivative of uncertain term, we can approximate the derivative of V_0 . From (4), we can get the control

$$\begin{aligned} u_0 &= -L_{G0}V_0 \\ &= -\left(x_2 + \left(x_1 + \int_0^\infty f_1(x_2 e^{-s}) ds\right) \right. \\ &\quad \left. + \left(x_1 + \int_0^\infty f_1(x_2 e^{-s}) ds\right) \frac{\partial \int_0^\infty f_1(x_2 e^{-s}) ds}{\partial x} \right) \end{aligned} \tag{9}$$

We can use the overparametrization proposed in [3] to estimate the unknown term. Take the Lyapunov candidate function V as

$$V = V_0 + \sum_{i=1}^3 \frac{1}{2} \tilde{\theta}_i^T \tilde{\theta}_i$$

where $\tilde{\theta}_i = \theta_i^* - \hat{\theta}_i$. Then its time derivative is

$$\begin{aligned} \dot{V} &= -x_2^2 + (x_1 + x_2 + g_1 + g_2 + g_3)u - \sum_{i=1}^3 \tilde{\theta}_i^T \dot{\hat{\theta}}_i \\ &= -x_2^2 + (x_2 + x_1 + \hat{\theta}_1^T \psi_1 + x_1 \hat{\theta}_2^T \psi_2 + \hat{\theta}_3^T \psi_3)u \\ &\quad + \sum_{i=1}^3 \varepsilon_i u - \sum_{i=1}^3 \tilde{\theta}_i^T \dot{\hat{\theta}}_i + (\tilde{\theta}_1^T \psi_1 + x_1 \tilde{\theta}_2^T \psi_2 + \tilde{\theta}_3^T \psi_3)u \end{aligned} \tag{10}$$

Take the stabilizing controller u as follows.

$$u = -(x_2 + x_1 + \hat{\theta}_1^T \psi_1 + x_1 \hat{\theta}_2^T \psi_2 + \hat{\theta}_3^T \psi_3) \tag{11}$$

and the regular law of $\theta_1, \theta_2, \theta_3$ is as follows.

$$\begin{aligned} \dot{\hat{\theta}}_1 &= \psi_1 u + \hat{\theta}_1 \\ \dot{\hat{\theta}}_2 &= x_1 \psi_2 u + \hat{\theta}_2 \\ \dot{\hat{\theta}}_3 &= \psi_3 u + \hat{\theta}_3 \end{aligned} \tag{12}$$

The following theorem shows the stability and control performance of the closed-loop adaptive system.

Theorem 2. *Consider the closed-loop system consisting of (1), the controller (11), and the NN weight updating laws (12). Assume that there exists sufficiently large compact sets $\Omega_{\theta_1}, \Omega_{\theta_2}, \Omega_{\theta_3}$, such that $\theta_i \in \Omega_i$. Then for bounded initial conditions, we have the following:*

All signals in the closed-loop system remain bounded.

Proof. substituting (11) and (12) into (10), we can get

$$\begin{aligned} \dot{V} &= -x_2^2 - u^2 + \sum_{i=1}^3 \varepsilon_i u - \sum_{i=1}^3 \tilde{\theta}_i^T \hat{\theta}_i \\ &\leq -x_2^2 - \frac{3}{4}u^2 - \sum_{i=1}^3 \tilde{\theta}_i^T \hat{\theta}_i + \sum_{i=1}^3 \varepsilon_i^2 \\ &\leq -x_2^2 - \frac{3}{4}u^2 - \frac{1}{2} \sum_{i=1}^3 \tilde{\theta}_i^T \tilde{\theta}_i + \frac{1}{2} \sum_{i=1}^3 \theta_i^{*T} \theta_i^* + \left(\sum_{i=1}^3 \varepsilon_i\right)^2 \\ &\leq -x_2^2 - \frac{3}{4}u^2 - \frac{1}{2} \sum_{i=1}^3 \tilde{\theta}_i^T \tilde{\theta}_i + \delta \end{aligned} \tag{13}$$

From the above inequality, we know that $x_2, \hat{\theta}, u$ are bounded. under the definition of (11) and (12), we know that x_1 is bounded.

4.2 Simulation

Here, a simple simulation is presented to show the effectiveness of the approach proposed above. The model of the system is given as

$$\begin{aligned} \dot{x}_1 &= 0.1x_2 + 5x_2^2 + 10 \sin(x_2) + u \\ \dot{x}_2 &= -x_2 + u \end{aligned} \tag{14}$$

where x_1 and x_2 are states. The initial conditions is $[x_1, x_2]^T = [1, 2]^T$. In this paper, all the basis functions of the NNs have the form

$$g_{i,j}(x_2) = \exp \left[-\frac{(x_2 - u_{i,j}^2)^2}{v_{i,j}^2} \right] \quad i = 1, 2, 3 \quad j = 1, \dots, 10 \tag{15}$$

where $u_{i,j}$ is the center of the receptive field and is the width of the Gaussian function. The NNs $\hat{\theta}_i \hat{\psi}_i$ contain 11 nodes, with centers evenly spaced in $[-2 \quad 2]$, and widths $v_{i,j} = 2$. The initial weights and are all given arbitrarily in $[0, 1]$. The simulation result as Fig 2.

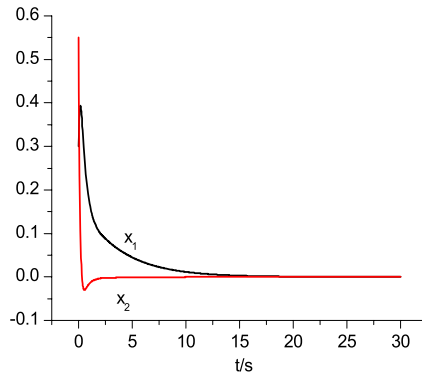


Fig. 2. The state of the system under adaptive NN controller

5 Conclusion

In this paper, the method about adaptive control using forwarding is proposed. The method can stabilize the feedforward system, when people know the structure of the systems and don't know the exact function. The proposed direct adaptive NN scheme can be applied to a large class of nonlinear strict-feedforward systems without repeating the complex controller design procedure for different system nonlinearities.

References

1. Choi, J.Y., Farrell, J.A.: Adaptive observer backstepping control using neural networks. *IEEE Transactions on Neural Networks*. **12** (2001) 1103–1112
2. Ge, S.S., Wang, C.: Direct adaptive nn control of a class of nonlinear systems. *IEEE Trans on Automatic Control*. **13** (2002) 214–221
3. Kanellakopoulos, I., Kokotovic, P.V., Morse A.S.: Systematic design of adaptive controllers for feedback linearizable systems. *IEEE Transaction on Automatic Control*. **36** (1991) 1241–1253
4. Krstic, M.: Feedback linearizability and explicit integrator forwarding controllers for classes of feedforward systems. *IEEE Transaction on Automatic Control*. **49** (2004) 1668–1682
5. Mazen, F., Nijmeijer, H.: Adding integrations, saturated controls, and stabilization of feedforward systems. *IEEE Trans on Automatic Control*. **41** (1996) 1559–1578
6. Sepulchre, R., Jankovic, M. Kokotovic, P.V.: *Constructive Nonlinear Control*. Springer. (1997)
7. Zhang, T., Ge, S.S., Han, C.C.: Adaptive neural network control for strict-feedback nonlinear systems using backstepping design. *Automatica*. **36** (2000) 1835–1846

Performance of an Ant Colony Optimization (ACO) Algorithm on the Dynamic Load-Balanced Clustering Problem in Ad Hoc Networks

Chin Kuan Ho and Hong Tat Ewe

Faculty of Information Technology, Multimedia University,
Persiaran Multimedia, 63100 Cyberjaya, Malaysia
{ckho, htewe}@mmu.edu.my

Abstract. This paper examines the performance of a recently proposed ACO algorithm when applied to the problem of constructing load-balanced clusters in ad hoc networks with node mobility. Performance, in this context, is measured in terms of the magnitude of change in solution quality after nodes move, and reactivity. Reactivity refers to the number of cycles ACO takes to recover from any degradation in solution quality resulting from node movements. Empirical results on 16 problem instances of various sizes revealed a positive correlation

1 Introduction

The ant colony optimization (ACO) metaheuristic, which first appeared in [1], is a probabilistic search method based on the foraging behavior of natural ants. From a broader perspective, it can be classified as a Swarm Intelligence technique [2]. ACO can be looked as an iterative multi-agent approach to problem solving. During each iteration, each individual ant (i.e. agent) attempts to construct a solution. Each ant will then share its experience with the entire colony by updating a data structure called the pheromone matrix or pheromone vector. This data structure mimics the pheromone trails that natural ants produce while finding foraging areas. Each entry in the pheromone matrix gives the desirability of each solution component, and is referred to as its pheromone intensity. At the end of each iteration, the pheromone associated to each solution component is reinforced according to the quality of the solution that comprises the particular solution component.

To put this work into perspective, we differentiate research efforts concerning ACO according to whether the combinatorial optimization (CO) problem being addressed is static or dynamic. A static problem is where the structural definition and problem related remain the same as the problem is being solved. Numerous applications of ACO algorithms to popular static problems have been widely published - traveling salesman problem [3], quadratic assignment problem [4], job-shop scheduling [5], vehicle routing [6], sequential ordering [7], graph coloring [8], and shortest common super-sequence [9].

On the other hand, a dynamic problem is characterized by changes in the problem structure or problem-related data (hereafter referred to as environment change) while the problem is being solved. The earliest applications of ACO to dynamic problems

have focused on problems arising from communication networks, most notably routing [10, 11]. It is to be noted that these ACO algorithms for routing are distributed in nature. However, of late, there has been increasing interest in enhancing ACO algorithms for addressing dynamic CO problems that take a centralized approach by making use of global information.

This paper takes aim at studying how well a recently proposed ACO algorithm [12] performs on the problem of dynamic clustering in ad hoc networks. We quantify its performance using two measures: 1. Change in solution quality after nodes move, and 2. reactivity, which gives an indication of how fast the algorithm can react and recover from any deterioration in solution quality.

2 Problem Description

An ad hoc network with N nodes can be modeled as an undirected graph $G = (V, E)$, where V is the set of vertices representing the nodes of the network and E is the set of edges. Let $\|ij\|$ be the Euclidean distance between nodes i and j . An edge $e_{ij} \in E$ exists between i and j if they can hear each other's transmission, i.e. if $\|ij\| \leq R$, where R is the transmission range of the nodes. Nodes are organized into clusters to enhance manageability

The problem of load-balancing the clusters arises due to the limitation of resources available to CHs. CHs can efficiently support up to a certain number of nodes. It is desirable to have each CH support about the same number of nodes. We define the load of a CH as the number of nodes it has in its cluster. This is based on the fact that in actual ad hoc networks, the work load of a CH is directly proportional to its cluster size. Therefore, the task of constructing load-balanced clusters can be formulated as follows: find a set of dominating nodes V' (the CHs) such that the nodes in $V - V'$ can be distributed as evenly as possible among the members of V' subject to the constraint that the number of dominated nodes per CH is at most *max_load*. The quality of the constructed clusters can then be measured by the variance of the load of the CHs, and is given by the following:

$$LBF = \frac{\sum_i (x_i - \mu)^2}{n_c} \quad (1)$$

where n_c is the number of clusters, x_i is the load of CH i , and $\mu = (N - n_c)/n_c$. A smaller LBF signifies better load distribution.

3 The ACO Approach

This section outlines the ACO approach that was proposed in [18] to construct load balanced clusters based on the ad hoc network topology graph.

3.1 The Visibility Measure

The visibility measure requires each node to be associated with a weight that indicates the number of uncovered neighboring nodes that it can cover. The ACO computes the

visibility measure η associated to node v at t_r (which means at step r of iteration t) using eq. (2). Nodes with weight equal to med will be the most visible. The visibility decreases proportional to the distance of $weight_v(t_r)$ from med . This applies if $weight_v(t_r) \leq max_load$. Nodes whose weight is larger than max_load will be given smaller visibility, hence a smaller chance to be selected. For the experiments, the constants R and S are set to $\{2, 3, 4, 5, 6\}$.

3.2 Node Selection and Pheromone Update Mechanisms

Taking into account both the pheromone and visibility factors, an ant k will select node v to be included in its partially constructed solution $S_k(t_r)$ at step r of iteration t with probability given by eq. (3). In eq. (3), α and β control the relative importance of the pheromone and visibility factors respectively. The term $allowed_k(t_r)$ is the set of nodes that ant k can choose from. Unlike the TSP, the visibility measure of each node is dynamic. This is due to the fact that as neighbors of each nodes are covered, the weight of that particular node decreases.

$$\eta_v(t_r) = \begin{cases} R^{maxPow - med - weight_v(t_r)}, & \text{if } weight_v(t_r) \leq max_load \\ \frac{weight_v(t_r) - max_load}{S^{max_load}}, & \text{otherwise} \end{cases} \tag{2}$$

where R and S are constants, $med = \text{median of } 1, 2, \dots, max_load$

$$maxPow = \begin{cases} max_load - med + 2 & \text{if } med \text{ is even} \\ max_load - med + 1 & \text{if } med \text{ is odd} \end{cases}$$

$$P_v^k(t_r) = \begin{cases} \frac{[\tau_v(t)]^\alpha [\eta_v(t_r)]^\beta}{\sum_{l \in allowed_k(t_r)} [\tau_l(t)]^\alpha [\eta_l(t_r)]^\beta} & \text{if } v \in allowed_k(t_r) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

At the end of iteration t , the pheromone for node i is updated for use in iteration $t+1$ using eq. (4):

$$\tau_i(t+1) = (1 - \rho) \tau_i(t) + \Delta \tau_i \tag{4}$$

where ρ is the pheromone evaporation factor, and $\Delta \tau_i$ is the pheromone deposited after iteration t is given in eq. (5).

$$\Delta \tau_i = \sum_{k=1}^a Q \cdot f(S_k) \quad \text{if } i \in S_k \tag{5}$$

where a is the number of ants, Q is a constant and $f(S_k)$ is the objective function applied to the solution constructed by ant k . This particular ACO algorithm incorporates a heuristic that, given a selected CH, determines which of its neighbors are to be covered. After each cluster has been constructed, the weights and states (covered or uncovered) of the nodes are updated.

4 Computational Experience

The performance of ACO algorithm is evaluated on four graphs representing four ad hoc networks. Each network is generated by randomly (uniform distribution) placing N nodes in an area of size $M \times M$ meters. From each network, we obtain four problem instances by varying the transmission range (R). The ACO algorithm is run 50 times for each problem instance, with each run taking a maximum of 150 iterations. The ACO parameters are set to the following: $\alpha = 1, \beta = 3, \rho = 0.1, Q = 0.5$ and number of ants is 30.

For each run, node mobility is allowed to take place after the 10th but before the 130th iterations. Mobility occurs only once within this period. This restriction is placed so that we can study the net effect of a single mobility event, instead of the cumulative effects of several events. A mobility event is characterized by three parameters: the number of nodes that move (denoted by j), the displacement vector $\mathbf{d} = \{d_1, d_2, \dots, d_j\}$ where d_i is the displacement for node i , and the direction vector $\boldsymbol{\theta} = \{\theta_1, \theta_2, \dots, \theta_j\}$ where θ_i represents the direction node i will move. Possible values for the movement direction are within $[0^\circ, 360^\circ]$. The nodes that will move, the iteration in which mobility will occur, as well as the displacement and direction vectors are determined randomly according to uniform distribution.

The performance of ACO in the presence of node mobility is measured using two metrics:

- *Resilience*. This refers to the change in solution quality immediately after nodes move, and is given by eq. (6), where m is the iteration in which mobility occurred, f_m is the value of the objective function at iteration t , $\overline{f}_{[m-5, m-1]}$ refers to the average objective function value within the five iterations before the mobility event. Positive resilience means the solution quality improved, while negative values indicate that the solution quality has degraded. Values for resilience are divided into four categories - C1: $\Gamma(m) > 0$, C2: $-0.05 \leq \Gamma(m) < 0$, C3: $-0.10 \leq \Gamma(m) < -0.05$ and C4: $\Gamma(m) < -0.10$.

$$\Gamma(m) = \frac{\overline{f}_{[m-5, m-1]} - f_m}{\overline{f}_{[m-5, m-1]}} \quad (6) \quad React(m) = \left\{ m' - m \mid m' \in \mathbb{N}, \frac{f_{m'}}{f_m} \leq (1 + \epsilon) \right\} \quad (7)$$

- *Reactivity*. This refers to the number of iterations that ACO requires to recover from any solution quality degradation, and is given by eq. (7), where ϵ is a parameter that is set to 0.01 for this study. A smaller reactivity value is desirable.

Experiments were performed for $j \in \{5, 6\}$ and $max_load \in \{4, 5, 6\}$, and the results are reported in Table 1 (for $j=5$) and Table 2 (for $j=6$). The columns labeled ‘Count’ gives the number of runs in which resilience under the respective categories was observed. The column labeled ‘Fail’ reports the number of runs in which ACO did not recover (which we will refer to as ‘failure’ in the next section) from the indicated solution degradation. Due to space restrictions, only the results for C3 and C4 are shown.

4.1 Results and Discussion

An obvious observation is that when the resilience is lower, the number of failures is larger. From the two tables, it is clear that for resilience categories C2 and C3, the number of failures is zero in most cases. When moving on to C4, there is a marked increase in the number of failures. We offer two possible explanations for this observation. Firstly, it can be the case that since the solution quality degradation is larger, hence more iterations, beyond the stipulated maximum 150 iterations, is required to recover. Secondly, the pheromone vector may have reached such a state that further reinforcements based on the new problem structure are ineffective. There is a strong negative correlation between the degree of resilience and reactivity. The lower the resilience (i.e. the larger the quality degradation), reactivity becomes larger.

It was also discovered that the number of runs displaying resilience category C4 and corresponding number of failures increase as the network gets smaller. This observation holds for both $j = 5$ and 6. For example, the C4 count for Net1 (all values of R and max_load) is far smaller than 20, while for Net4, much larger than 20 for most cases. The reason is that the sensitivity of the overall load balance of the system as computed by way of eq. (1) depends on the number of nodes. In other words, the effects of mobility are more profound if the ratio of j/N is larger.

The number of nodes that move (represented by j) also affects performance, in particular for category C4. When the total number of failures (category C4) are added up for each network (across all R and max_load), it was found that $j = 6$ gave more failures. To illustrate, under $j = 6$, there is a total of 28 failures, while under $j = 5$, there is a total of 16. Similar behavior can be observed for Net3 and Net4. One exception is Net4, where similar totals were obtained.

The fact that there are numerous runs that produced positive resilience (category C1) has an interesting general implication. Say that the original problem is A , and through some transformation, we obtain B such that $B \in Neighbor(A)$, i.e. B is a neighbor of A through some neighborhood function $Neighbor$. If the collective experience on A can be used to help give a better solution to B , that means that there may exist another problem $C \in Neighbor(A)$ such that we can use the collective search experience on C to better solve A . The count for resilience category C1 ranges from 5 to 25 for all the runs.

Lastly, the empirical results did not reveal any clear correlation between $max_load = \{4, 5, 6\}$ and resilience as well as reactivity.

Table 1. The resilience and reactivity of ACO on the four networks with $j = 5$

Network Description	R	max load	C3: $-0.10 \leq \Gamma(m) < -0.05$			C4: $\Gamma(m) < -0.10$		
			Count	Reactivity	Fail	Count	Reactivity	Fail
Net1 400 nodes Area=3000m x 3000m	210	4	12	4.7	0	10	28.4	3
		5	9	5.9	0	10	16.2	0
		6	10	8.8	0	3	28.0	0
	220	4	16	11.1	1	11	13.8	3
		5	8	4.1	1	10	27.0	3
		6	14	3.7	0	8	12.0	1
	230	4	9	3.5	1	6	25.8	1
		5	8	3.1	0	11	15.4	2
		6	10	8.3	0	8	17.8	0
	240	4	12	8.7	0	7	23.8	1
		5	10	16.9	0	11	17.8	1
		6	7	8.9	0	12	17.3	1
Net2 300 nodes Area=2000m x 2000m	180	4	9	5.0	0	18	13.4	9
		5	7	4.1	0	20	15.3	3
		6	6	4.0	0	21	27.4	3
	190	4	11	7.7	0	23	14.8	3
		5	11	4.6	0	18	12.6	1
		6	9	4.8	0	24	16.5	3
	200	4	6	3.7	0	17	4.4	3
		5	7	7.9	0	19	14.2	3
		6	5	9.8	0	18	10.7	1
	210	4	9	4.0	0	20	7.1	2
		5	11	9.7	0	12	25.9	4
		6	7	4.7	0	22	16.5	1
Net3 200 nodes Area=1000m x 1000m	100	4	13	13.7	1	15	18.1	6
		5	14	8.7	2	16	26.7	3
		6	7	6.6	0	18	13.4	1
	110	4	3	6.3	0	21	12.4	9
		5	8	9.0	0	20	15.3	8
		6	6	2.5	0	25	14.9	9
	120	4	6	5.2	0	22	24.8	10
		5	11	14.5	0	19	15.9	4
		6	3	9.0	0	28	7.8	8
	130	4	9	3.3	1	14	14.1	6
		5	12	8.8	1	14	13.3	3
		6	6	2.7	0	24	10.6	5
Net4 100 nodes Area=600m x600m	80	4	4	8.7	1	30	14.3	18
		5	2	9.0	1	32	16.6	20
		6	2	15.0	0	30	12.0	21
	90	4	5	6.8	1	27	26.0	14
		5	6	7.3	0	30	19.5	18
		6	4	44.5	2	28	16.4	19
	100	4	2	4.0	1	31	13.2	12
		5	3	12.3	0	33	16.1	18
		6	1	4.0	0	40	13.7	22
	110	4	0	N/A	0	28	7.4	12
		5	2	5.0	0	24	13.6	14
		6	4	23.8	0	29	13.2	14

Table 2. The resilience and reactivity of ACO on the four networks with $j = 6$.

Network Description	R	max_load	C3: $-0.10 \leq \Gamma(m) < -0.05$			C4: $\Gamma(m) < -0.10$		
			Count	Reactivity	Fail	Count	Reactivity	Fail
Net1 400 nodes Area=3000m x3000m	210	4	13	6.9	1	11	12.1	4
		5	8	5.6	0	9	19.6	2
		6	8	14.0	0	8	21.1	1
	220	4	7	7.9	0	15	15.9	3
		5	10	8.0	0	14	15.9	2
		6	11	9.1	1	12	8.5	1
	230	4	15	5.9	0	12	23.1	3
		5	13	7.7	0	9	7.7	2
		6	10	9.1	0	12	24.0	2
	240	4	10	14.1	0	16	12.1	3
		5	9	9.2	0	17	13.9	3
		6	8	5.6	0	15	10.6	2
Net2 300 nodes Area=2000m x2000m	180	4	10	4.5	0	19	14.6	4
		5	9	11.7	0	18	7.9	5
		6	3	6.7	0	19	10.1	2
	190	4	6	3.3	0	26	9.0	4
		5	9	3.9	1	22	10.7	6
		6	9	5.4	0	23	9.8	2
	200	4	9	3.2	0	15	9.8	3
		5	9	5.6	0	21	11.6	6
		6	6	3.8	0	22	12.9	3
	210	4	6	2.8	0	25	16.4	3
		5	11	7.7	0	22	14.4	4
		6	8	5.4	0	23	16.4	1
Net3 200 nodes Area=1000m x1000m	100	4	9	8.5	1	17	23.3	7
		5	9	8.7	0	21	24.9	7
		6	5	5.6	0	23	14.7	8
	110	4	6	13.6	1	27	14.3	10
		5	4	2.3	0	25	20.8	11
		6	6	4.7	0	32	20.0	9
	120	4	12	6.2	2	22	22.6	8
		5	6	3.8	0	24	22.9	3
		6	3	4.0	0	29	12.2	6
	130	4	4	3.8	0	19	16.9	5
		5	10	11.0	0	18	14.5	5
		6	6	26.0	1	23	18.1	9
Net4 100 nodes Area=600m x600m	80	4	4	6.5	0	30	23.4	21
		5	2	4.0	0	31	21.8	17
		6	2	18.5	0	30	20.8	12
	90	4	1	2.0	0	33	24.5	20
		5	3	18.0	1	39	14.3	24
		6	3	14.3	0	29	38.6	15
	100	4	5	6.6	0	28	16.7	19
		5	3	6.7	0	33	14.4	16
		6	0	0.0	0	37	26.7	16
	110	4	4	2.8	0	21	10.0	12
		5	2	35.0	0	33	10.9	13
		6	4	4.7	1	34	12.7	17

5 Conclusions

This paper has investigated the ability of an ACO algorithm to respond to environment changes. In particular, we have examined its robustness in terms of resilience and reactivity time. In almost all cases where the solution quality degradation is small, ACO managed to recover with acceptable reactivity times. However, when the degradation is at least 10%, ACO required on average 20 iterations to recover. The results obtained from this study motivate a couple of future work. Firstly, we are currently looking into means to increase the algorithm's resilience. Secondly, we also plan to research mechanisms that will enable ACO to react more quickly to environment changes.

References

1. Dorigo, M., Maniezzo, V., Colomi, A. : Positive Feedback as a Search Strategy. Technical Report 91-016, Politecnico di Milano, Italy (1991).
2. Bonabeau, E., Thérault, G. : Swarm Smarts. *Scientific American*, Vol. 282, No. 3 (2000), 72-79.
3. Dorigo, M., Gambardella, L.M.: Ant Colony System: Optimization by a colony of cooperating agents. *IEEE Trans. Systems, Man, and Cybernetics – Part B*, Vol. 26, No. 1 (1996), 29-41.
4. Maniezzo, V., Colomi, A: The ant system applied to the quadratic assignment problem. *IEEE Trans. Knowledge and Data Engineering*, Vol. 11, No. 5 (1999), 769-778.
5. Colomi, A., Dorigo, M., Maniezzo, V., Trubian, M.: Ant system for job-shop scheduling. *Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL)*, Vol. 34 (1994), 39-53.
6. Gambardella, L.M., Taillard, E., Agazzi, G.: Ant colonies fir vehicle routing problems. In Corne, D., Dorigo, M. and Glover, F., editors, *New Ideas in Optimization*, McGraw-Hill (1999).
7. Gambardella, L.M., Dorigo, M.: HAS-SOP: An hybrid ant system for the sequential ordering problem. Technical Report 11-97, IDSIA, Lugano, CH (1997).
8. Costa, D., Hertz, A.: Ants can color graphs. *Journal of the Operations Research Society*, Vol. 48 (1997), 295-305.
9. Michel, R., Middendorf, M.: An island model based ant system with lookahead for the shortest supersequence problem. In Eiben, A.E., Back, T., Schoenauer, M., and Schwefel, H.P., editors, *Proceedings of PPSN-V, Fifth International Conference on Parallel Problem Solving from Nature*, Springer-Verlag (1998), 692-701.
10. Di Caro, G., Dorigo, M.: AntNet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research (JAIR)*, Vol. 9 (1998), 317-365.
11. Subramanian, D, Druschel, P., Chen, J.: Ants and reinforcement learning: A case study in routing in dynamic networks. In *Proc. of IJCAI 97, International Joint Conference on Artificial Intelligence*, Morgan Kaufmann (1997), 832-838.
12. Ho, C.K., Ewe, H.T.: A hybrid ant colony optimization approach for creating load-balanced clusters. In *Proc. IEEE Congress on Evolutionary Computation 2005 (CEC 2005)*, 2-5 September 2005, Edinburgh, Scotland.

Hybrid Particle Swarm Optimization for Flow Shop Scheduling with Stochastic Processing Time

Bo Liu¹, Ling Wang^{1,2}, and Yi-hui Jin¹

¹ Department of Automation, Tsinghua University, Beijing 100084, China
Liub01@mails.tsinghua.edu.cn

² School of Information Engineering, Shandong University at Weihai,
Weihai 264209, China

Abstract. The stochastic flow shop scheduling with uncertain processing time is a typical NP-hard combinatorial optimization problem and represents an important area in production scheduling, which is difficult because of inaccurate objective estimation, huge search space, and multiple local minima. As a novel evolutionary technique, particle swarm optimization (PSO) has gained much attention and wide applications for both function and combinatorial problems, but there is no research on PSO for stochastic scheduling cases. In this paper, a class of PSO approach with simulated annealing (SA) and hypothesis test (HT), namely PSOSAHT is proposed for stochastic flow shop scheduling with uncertain processing time with respect to the makespan criterion (i.e. minimizing the maximum completion time). Simulation results demonstrate the feasibility, effectiveness and robustness of the proposed hybrid algorithm. Meanwhile, the effects of noise magnitude and number of evaluation on searching performances are also investigated.

1 Introduction

Flow shop scheduling problem (FSSP) is a class of widely studied scheduling problem with a strong engineering background, and has earned a reputation for being difficult to solve [1]. The permutation FSSP (PFSSP) to minimize the maximum completion time (i.e. makespan C_{\max}) is a typical and well-studied scheduling problem, which is well known to be a NP-hard problem when $m > 2$. PFSSP have received considerable theoretical and computational research works, and various optimization techniques have been proposed. But in many real manufacturing environments, uncertainty is so prevalent that it is more important and practical to study stochastic scheduling problems than deterministic ones. In this paper, PFSSP with stochastic processing time is considered, where the processing time $p_{i,j}$ of job i on machine j is assumed to be random. In such case, the expected makespan is often used to evaluate the performance of the solutions.

Recently, a novel evolutionary technique, namely particle swarm optimization (PSO) has been proposed [2], which was developed based on bird flocking, fish schooling, and swarm theory. Due to the simple concept, easy implementation and quick convergence, nowadays PSO has gained much attention and wide applications in a variety of fields [3], [4], [5]. However, most of the published research about PSO

is for continuous optimization problems, while little research can be found for combinatorial optimization problems, especially on scheduling problems.

To the best of our knowledge, there is no any published work for dealing with stochastic PFSSP using PSO. In this paper, a class of PSO approach with simulated annealing (SA) and hypothesis test (HT), namely PSOSAHT is proposed for PFSSP with stochastic processing time with respect to makespan criterion. The proposed PSOSAHT stresses the balance between the global exploration and local exploitation. Firstly, to make PSO suitable for solving flow shop scheduling, a new encoding rule based on random key representation, called ranked-order-value (ROV) rule, is presented to convert the continuous position values of a particle to job permutation. Secondly, the evolutionary searching mechanism of PSO characterized by individual improvement plus population cooperation and competition is utilized to perform the exploration effectively. Thirdly, jumping probability of SA is employed to reduce the probability to be trapped in local minimum and perform the exploitation well. Moreover, HT is incorporated into the hybrid PSO to reasonably estimate the solution performance and reliably identify solution quality so that the repeated search can be reduced to some extent. Simulation results demonstrate the feasibility, effectiveness and robustness of the proposed algorithm.

2 Reviews of PSO and HT

2.1 PSO

PSO is started with the random initialization of a population (*swarm*) of individuals (*particles*) in the search space and works on the social behavior of the particles. The position and the velocity of the *i*-th particle in the *d*-dimensional search space can be represented as $X_i = [x_{i,1}, x_{i,2}, \dots, x_{i,d}]$ and $V_i = [v_{i,1}, v_{i,2}, \dots, v_{i,d}]$ respectively. Each particle has its own best position (*pbest*) $P_i = (p_{i,1}, p_{i,2}, \dots, p_{i,d})$ corresponding to the personal best objective value obtained so far at time *t*. The global best particle (*gbest*) is denoted by P_g , which represents the best particle found so far at time *t* in the entire swarm. The new velocity of each particle is calculated as follows:

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1r_1(p_{i,j} - x_{i,j}(t)) + c_2r_2(p_{g,j} - x_{i,j}(t)), j = 1, 2, \dots, d \tag{1}$$

where c_1 and c_2 are acceleration coefficients, w is called the inertia factor, r_1 and r_2 are two independent random numbers uniformly distributed in the range of [0, 1].

Thus, the position of each particle is updated according to the following equation:

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), j = 1, 2, \dots, d. \tag{2}$$

2.2 HT

Generally, a stochastic optimization problem can be described as follows:

$$\min_x J(X) = E[L(X, \xi)]. \tag{3}$$

where X is a feasible solution of the problem in a finite set and J is the expectation of L , the sample performance as a function of X and ξ (noise or uncertain factors).

Hypothesis test (HT) is an important statistical method that is used to make test for predefined hypothesis based on experiment data [6]. To perform HT for two different solutions when solving uncertain optimization problems, it often needs multiple independent evaluations to provide suitable performance estimation for decision solutions. If n_i independent simulations are carried out for solution X_i , then its unbiased estimated mean value \bar{J}_i and variance s_i^2 can be calculated as follows:

$$\bar{J}_i = \bar{J}(X_i) = \sum_{j=1}^{n_i} L(X_i, \xi) / n_i. \tag{4}$$

$$s_i^2 = \sum_{j=1}^{n_i} [L(X_i, \xi) - \bar{J}_i]^2 / (n_i - 1). \tag{5}$$

Considering two different solutions X_1 and X_2 , whose estimated performances $\hat{J}(X_1)$ and $\hat{J}(X_2)$ are two independent random variables. According to the law of large number and central limit theorem, the estimation $\hat{J}(X_i)$ subjects to $N(\bar{J}_i, s_i^2/n_i)$ when n_i approaches to ∞ . Suppose $\hat{J}(X_1) \sim N(\mu_1, \sigma_1^2)$ and $\hat{J}(X_2) \sim N(\mu_2, \sigma_2^2)$, where the unbiased estimation values of μ_1 , μ_2 and s_1^2 , s_2^2 are given by Eqs.(4) and (5), and let the null hypothesis H_0 be “ $\mu_1 = \mu_2$ ” and the alternative hypothesis H_1 be “ $\mu_1 \neq \mu_2$ ”.

If $\sigma_1^2 = \sigma_2^2 = \sigma^2$ and σ^2 is unknown, below is the critical region of H_0 :

$$|\bar{J}_1 - \bar{J}_2| \geq t_{\alpha/2}(n_1 + n_2 - 2) \cdot Y_1 \cdot Y_2 = \tau. \tag{6}$$

where $Y_1 = \sqrt{[(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2] / (n_1 + n_2 - 2)}$, $Y_2 = \sqrt{(n_1 + n_2) / (n_1 n_2)}$

Thus, if $|\bar{J}_1 - \bar{J}_2| < \tau$, i.e., the null hypothesis holds, then it can be regarded that the performances of those two solutions have no significant difference in statistical sense, otherwise they are significantly different. Furthermore, for uncertain minimization problem it is assumed that X_2 is better than X_1 if $\bar{J}_1 - \bar{J}_2 \geq \tau$, while X_1 is better than X_2 if $\bar{J}_1 - \bar{J}_2 \leq -\tau$. In addition, for specific problem it often supposes that the theoretical performance variances of all solutions are the same [6], so the hypothesis test can be made according to Eq.(6). For multi-modal stochastic optimization problems, a comparison under pure HT can often be trapped into local optima. This motivates us to propose a hybrid approach by incorporating HT.

3 PSOSAHT for Stochastic Flow Shop Scheduling

3.1 Solution Representation

Usually, a job-permutation based encoding scheme has been widely used for PFSSP. However, because of the continuous characters of the position of particles in PSO, standard encoding scheme for PSO can not be directly adopted for PFSSP. In this paper, a ranked-order-value (ROV) rule based on random key representation is presented to convert the continuous positions of PSO $X_i = [x_{i,1}, x_{i,2}, \dots, x_{i,n}]$ to permutations of jobs $\pi = \{j_1, j_2, \dots, j_n\}$.

The position information X_i itself does not represent a sequence, while the rank of each position value of a particle represents a job-index so as to construct a permutation of jobs. In our ROV rule, the smallest position value of a particle is firstly dealt with and assigned a smallest rank value 1. If there are two or more same position values that may be happened in PSO-based searching process, then the position value with the smallest dimension number is assigned a rank value 1. And the rank values of rest position values are added with 1 one by one as the increase of dimension number. Then, the second smallest position value will be dealt with the same way. Note that, if there are k same smallest position values, the rank value of the second smallest position value with the smallest dimension should be $k + 1$. With the same way, all the position values will be dealt with so as to convert the position information of a particle to a job permutation.

In Table 1, we provide a simple example to illustrate the ROV rule. In that instance ($n = 6$), position information is $X_i = [0.06, 2.99, 1.86, 3.73, 1.86, 0.67]$. So, $x_{i,1} = 0.06$ is first dealt with and assigned rank value 1, then $x_{i,6} = 0.67$ is assigned rank value 2. Since there are two 1.86 in position information, i.e., $x_{i,3}$ and $x_{i,5}$, the ROV rule assigns rank value 3 and 4 for $x_{i,3}$ and $x_{i,5}$ respectively. Lastly, $x_{i,2}$ and $x_{i,4}$ are assigned rank value 5 and 6 respectively. Thus, the job permutation is obtained, i.e., $\pi = [1, 5, 3, 6, 4, 2]$.

Table 1 Representation of position information and the corresponding ROV (job permutation)

Dimension j	1	2	3	4	5	6
Position value	0.06	2.99	1.86	3.73	1.86	0.67
Ranked-order-value	1	5	3	6	4	2

In our PSOSAHT, a SA based local search is applied to job permutation directly, not for position information directly. So, when a local search is completed, the particle's position information should be repaired to guarantee the permutation resulted by ROV rule for new position information is the same as the permutation resulted by local search. That is to say, applying local search to job permutation, position information should be adjusted correspondingly. The process to position information is the same as the process to permutation. For example, if SWAP operator

[7] is used as a local search for job permutation in Table 2, obviously the swap of job 5 and job 6 is corresponding to the swap of position value 2.99 and value 3.73.

Table 2. Swap-based local search for job permutation and the adjustment for position

Dimension j	1	2	3	4	5	6
Position value	0.06	<u>2.99</u>	1.86	<u>3.73</u>	1.86	0.67
Job permutation	1	<u>5</u>	3	<u>6</u>	4	2
Dimension j	1	2	3	4	5	6
Position value	0.06	<u>2.99</u>	1.86	<u>3.73</u>	1.86	0.67
Job permutation	1	<u>6</u>	3	<u>5</u>	4	2

3.2 PSO-Based Search Combining HT

In this paper, the position information of particles in current swarm is evolved by PSO-based searching operator, i.e., Eq. (1) and (2), for exploration. So, when evaluating the performance of a particle, the position information can be converted to job permutation using the above ROV rule. On the other hand, aimed at the stochastic PFSSP, multiple independent evaluations will be used to provide reasonable performance estimation for solutions. Meanwhile, hypothesis test is added into the PSO to identify the quality of different solutions when updating $pbest$ and $gbest$.

3.3 SA-Based Local Search Combining HT

SA-based local search is designed to enrich the local searching behavior and to avoid premature convergence. Meanwhile, to achieve good performances for stochastic problems, HT is to identify the quality of solution and to reduce repeated search. The procedure of SA-based local search combining HT is described as follows:

Step 1: Let the permutation of jobs π_1 be the initial solution, and calculate its performance \bar{J}_1 and variance estimation s_1^2 through certain times of independent evaluations. Set π_1 be the best initial solution, i.e., $\pi^* = \pi_1$, denote the performance estimation and variance estimation of π^* as $\bar{J}^* = \bar{J}_1$ and $s^{*2} = s_1^2$ respectively.

Step 2: Generate a neighbor π_2 of π_1 , and estimate its performance \bar{J}_2 and variance s_2^2 by certain times of independent evaluations.

Step 3: Perform hypothesis test for π_1 and π_2 . If the null hypothesis holds, i.e., Eq.(6) does not hold, then go back to step 2; otherwise, continue the following steps.

Step 4: Replace the current solution π_1 by π_2 with probability $\min\{1, \exp[-(\bar{J}_2 - \bar{J}_1)/t]\}$. And update π^* , \bar{J}^* and s^{*2} if possible.

Step 5: if M_1 times neighborhood samplings are performed at current temperature, then output the best solution π^* ; otherwise, go to step 2.

In this paper, the SWAP operator [7] is utilized as the neighborhood generator, which means that two distinct elements of an n -job permutation are randomly selected

and swapped. Moreover, exponential cooling schedule, $t_k = \lambda t_{k-1}$ (where $0 < \lambda < 1$), is applied, which is often believed to be an excellent cooling recipe [8]. To provide a rather good compromise between solution quality and search efficiency, the step of Metropolis sampling process is set to $M_1 = n \cdot (n - 1)$, where n denotes the number of jobs. In addition, above SA-based local search is only applied to the best solution found so far, i.e., g_{best} . After the SA-based local search is completed, g_{best} of the whole swarm should be updated if a solution with better quality is found.

3.4 PSOSAHT

In this sub-section, the PSOSAHT is proposed for the stochastic optimization problem, where PSO and SA are used for exploration and exploitation in search space respectively, and HT is applied to identify the quality of solution and to reduce repeated search. The procedure of PSOSAHT is described as follows.

Step 1: Generate P_s particles with random position values, and determine the corresponding job permutations by ROV.

Step 2: Evaluate initial population in statistical sense, and determine p_{best} and g_{best} , set initial temperature.

Step 3: If g_{best} keeps fixed at consecutive L steps, then output the best solution; otherwise, go to Step 4.

Step 4: Update swarm using PSO-based operation, determine permutation using ROV rule, and evaluate the swarm in statistical sense.

Step 5: Update the p_{best} of each particle and g_{best} using HT.

Step 6: Perform SA-based local search combining HT for g_{best} .

Step 7: Update the temperature and return to Step 3.

4 Numerical Tests

In this paper, eight problems [7] named car1 through car8 by Carrier and two problems named Rec01 and Rec03 by Reeves are employed for test. The processing time is subjected to a uniform distribution $U((1 - \eta)P_{i,j}, (1 + \eta)P_{i,j})$, where $P_{i,j}$ is the expected processing time provided by the benchmarks, η denotes noise magnitude.

Three approaches PSOSA1, PSOSA2 and PSOSAHT are compared. In PSOSA1, HT is removed from PSOSAHT and only one-evaluation is used for solution performance estimation. In PSOSA2, HT is removed from PSOSAHT and 10-evaluation is used. In PSOSAHT, HT is applied and 10-evaluation is used. Setting the swarm size $p_s = 20$, $w = 1.0$, $c_1 = c_2 = 2$, $x_{\min} = 0$, $x_{\max} = 4.0$, $v_{\min} = -4.0$, $v_{\max} = 4.0$, initial temperature $T_0 = 3.0$, annealing rate $\lambda = 0.9$, stopping parameter $L = 5$, and noise magnitude $\eta = 5\%$ for all the problems. Each approach is independently run 20 times for every problem, and the statistical results are listed in Table 3. C^* means the theoretically expected makespan of optimal solution. BEM and AEM respectively denote the relative error of the best and average expected makespan (calculated with

the expected processing time for those solutions obtained by the algorithm with estimated performances) with respect to the theoretically optimal value C^* . From Table 3, it is shown that PSOSAHT is very effective for solving stochastic flow shop scheduling problems. Solutions resulted by PSOSAHT always are the best, while PSOSA1 performs the worst. The effectiveness of the PSOSAHT is due to the statistical comparison based on HT to reduce repeated search, reliable performance estimation based on multiple evaluations and the combination of global search and local search, i.e., the balance of exploration and exploitation.

Table 3. Comparisons between PSOSAHT, PSOSA1, and PSOSA2

Problem	n, m	C^*	PSOSAHT		PSOSA1		PSOSA2	
			BEM	AEM	BEM	AEM	BEM	AEM
Car1	11,5	7038	0	0.0071	0	0.0135	0	0.0213
Car2	13,4	7166	0	0	0	1.1862	0	0.5966
Car3	12,5	7312	0	1.1303	0	1.2712	0	1.1303
Car4	14,4	8003	0	0	0	0.0619	0	0
Car5	10,6	7720	0	0.669	0	1.1224	0	0.7234
Car6	8,9	8505	0	1.2934	0	1.642	0	1.3357
Car7	7,7	6590	0	0.0835	0	0.2777	0	0.1222
Car8	8,8	8366	0	0.3508	0	0.6807	0	0.5026
Rec01	20,5	1247	0.1604	0.2205	0.1604	1.9808	0.1604	0.405
Rec03	20,5	1109	0.1803	0.266	0.1803	0.8702	0.1803	0.4058

Table 4. Effect of estimation accuracy on PSOSAHT

Problem	n, m	C^*	$\eta = 0.05$			$\eta = 0.10$		
			10-eval.	20-eval.	30-eval.	10-eval.	20-eval.	30-eval.
Car1	11,5	7038	0.0071	0.0053	0	0.0284	0.0071	0
Car2	13,4	7166	0	0	0	2.2704	2.0514	1.6264
Car3	12,5	7312	1.1303	1.1078	1.1078	1.1898	1.1898	1.1905
Car4	14,4	8003	0	0	0	0	0	0
Car5	10,6	7720	0.669	0.3627	0.3657	1.3873	1.2111	0.8439
Car6	8,9	8505	1.2934	1.2326	1.0288	2.2252	1.7913	1.067
Car7	7,7	6590	0.0835	0	0	0.8369	0.4431	0.0417
Car8	8,8	8366	0.3508	0.3459	0	1.0585	0.5403	0.7309
Rec01	20,5	1247	0.2205	0.1824	0.1822	4.9679	2.915	2.7907
Rec03	20,5	1109	0.266	0.2882	0.2275	2.0243	1.5464	0.9468

In addition, the estimation accuracy of objective value is relative to two aspects: evaluations times for performance estimation and uncertainty magnitude. So, we carry out simulations on different noise magnitude and different evaluation times. The statistical results are listed in Table 4. From Table 4, it can be concluded that, the more evaluation times are used for performance estimation, the better results can be obtained. Secondly, under the same evaluation times used, PSOSAHT can find better solutions as the uncertainty magnitude decreases. This phenomenon is due to the facts that hypothesis test can work more effectively if estimation is more accurate and performance estimation can be more accurate if uncertainty is smaller. However,

more evaluation times may increase the computational effort of the algorithm. Thus, a trade off should be made between the computational effort and the solution quality.

5 Conclusions

To the best of our knowledge, this is the first report for dealing with stochastic PFSSP using PSO. Simulation results demonstrated the effectiveness and efficiency of the proposed approach.

Acknowledgements. This work is supported by National Natural Science Foundation of China (Grant No. 60204008 and 60374060) and National 973 Program (Grant No. 2002CB312200).

References

1. Wang, L.: Shop Scheduling with Genetic Algorithms. Tsinghua Univ. & Springer, Beijing (2003)
2. Kennedy, J., Eberhart, RC.: Particle Swarm Optimization. Proc. IEEE Int. Conf. on Neural Networks (1995) 1942-1948
3. Liu, B., Wang, L., Jin, Y.H., Huang, D.X.: Advances in Particle Swarm Optimization Algorithm. Control and Instruments in Chemical Industry 32 (2005) 1-6
4. Liu, B., Wang, L., Jin, Y.H., Tang, F., Huang, D.X.: Improved Particle Swarm Optimization Combined with Chaos. Chaos, Solitons and Fractals 25 (2005) 1261-1271
5. Liu, B., Wang, L., Jin, Y.H., Huang, D.X.: Designing Neural Networks Using Hybrid Particle Swarm Optimization. Lecture Notes in Computer Science, Vol. 3496. Springer-Verlag, Berlin Heidelberg (2005) 391-397
6. Wang, L., Zhang, L., Zheng, D.Z.: A Class of Hypothesis-test Based Genetic Algorithm for Flow Shop Scheduling with Stochastic Processing Time. Int J. Adv. Manuf. Technol 25 (2005) 1157-1163
7. Wang, L., Zheng, D.Z.: An Effective Hybrid Heuristic for Flow Shop Scheduling. Int. J. Adv. Manuf. Technol. 21 (2003) 38-44
8. Wang, L., Zheng, D.Z.: An Effective Hybrid Optimization Strategy for Job-shop Scheduling Problems. Comput. Oper. Res. 28 (2001) 585-596

Particle Swarm Optimizer with C-Pg Mutation

Guojiang Fu^{1,2}, Shaomei Wang², Mingjun Chen¹, and Ning Li¹

¹ School of Computer, Wuhan University of Technology,
Wuhan 430070, China
{Fu, dubistbeikun}@eyou.com
{Chen, cmjcmj1}@163.com
{Li, zidong}@tom.com

² School of Logistics, Wuhan University of Technology,
Wuhan 430063, China
{Wang, dubistwang}@eyou.com

Abstract. This paper presents a modified PSO algorithm, called the PSO with *C-Pg* mutation, or PSOWC-Pg, the algorithm adopts *C-Pg* mutation, the idea is to replace global optimal point *gBest* with disturbing point *C* and *gBest* alternately in the original formulae, the probability of using *C* is *R*. There are two methods for selecting *C*: stochastic method and the worst fitness method. The stochastic method selects some particle's current position *x* or *pBest* as *C* stochastically in each iteration loop, the worst fitness method selects the worst particle's *x* or the *pBest* of some particle with the worst fitness value as *C*. So, when *R* is small enough, the distance between *C* and *gBest* will tend towards 0, particle swarm will converge slowly and irregularly. The results of experiments show that PSOWC-Pg exhibit excellent performance for test functions.

1 Introduction

Particle Swarm Optimization is a stochastic optimization algorithm based on swarm intelligent, it is inspired by the conclusions from the social behavior simulation of bird swarm in artificial life study, and proposed by Kennedy and Eberhart^[1]. The following is a brief introduction to PSO.

Assuming a particle swarm containing *m* particles flying (searching) in *D* dimension space, the particle swarm can be expressed by the following parameters: $x_i = (x_{i,1}, x_{i,2}, \dots, x_{i,D})$ is the Position of particle *i*; $v_i = (v_{i,1}, v_{i,2}, \dots, v_{i,D})$ is the Velocity of particle *i*; $P_i = (P_{i,1}, P_{i,2}, \dots, P_{i,D})$ is the historically optimal position of particle *i*, i.e.

pBest; $P_g = (P_{g,1}, P_{g,2}, \dots, P_{g,D})$ is the optimal position among all P_i , i.e. ***gBest***.

The iterative formulae:

$$\begin{cases} v_{i,d} = \omega v_{i,d} + c_1 \cdot rand \cdot (P_{i,d} - x_{i,d}) + c_2 \cdot rand \cdot (P_{g,d} - x_{i,d}) \\ x_{i,d} = x_{i,d} + v_{i,d} \end{cases} \quad (1)$$

Where $i=1,2,\dots,m$; $d=1,2,\dots,D$. *rand* is a random number over [0,1], c_1, c_2 are constants over [0, 2], called acceleration factor, ω is also a constant called inertia

weight. For promoting the algorithm's convergent rate Clerc proposed a PSO convergent factor model^[2], according to the model, $\omega=0.729$, $c_1=c_2=1.49445$. In this paper, all algorithms will use such parameters.

The rest of this paper is structured as follows: Section 2 discusses PSOWC-Pg concretely, section 3 presents the organization of the experiment, the experiment results and the discussion of the results, and section 4 concludes the paper by summarizing our work and future work.

2 Particle Swarm Optimizer with C-Pg Mutation (PSOWC-Pg)

In the original PSO, when the search reaches the later period, particle swarm are liable to converging to local minimum or global minimum, this moment, each particle's $pBest$ and x , $gBest$ tend towards the same point, and each particle's velocity tends towards zero. For improving the quality of results, when particle swarm converge to some bound, the part particles of swarm are reinitialized, that is, each particle's value on each dimension is redistributed evenly in the feasible scope, this is just traditional mutation.

The mutation strategy of PSOWC-Pg is different from that of traditional mutation. With Eq.(1), Pg is replaced with disturbing point C , and we can get iteration formulae Eq.(2):

$$\begin{cases} v_{i,d} = \omega v_{i,d} + c_1 \cdot rand \cdot (P_{i,d} - x_{i,d}) + c_2 \cdot rand \cdot (C_d - x_{i,d}) \\ x_{i,d} = x_{i,d} + v_{i,d} \end{cases} \quad (2)$$

In each loop within the searching process, one of (1) and (2) is selected stochastically as iteration formula in this loop, the essence is that Pg is replaced with C or Pg according to probability R alternately in the iteration formulae. This mutation measure is called "C-Pg mutation", PSO algorithm adopting "C-Pg mutation" is called "PSO with C-Pg mutation (PSOWC-Pg)". Inspecting the movement of the particle with C-Pg mutation, Eq. (1) make a particle with the inertia ($\omega v_{i,d}$) that gets from the previous iteration loop go in the direction of $pBest$, $gBest$, while Eq.(2) make a particle with the inertia ($\omega v_{i,d}$) that gets from the previous iteration loop go in the direction of $pBest$, C (Figure 1), when R reaches to zero, Eq.(2) have no effect, the algorithm degenerates into original PSO, with $pBest$, $gBest$ coming closer, the current positions of particles tend towards $pBest$, $gBest$ rapidly, particle swarm converge very soon. When $R \in (0,1)$, the movement trajectory of the particles are controlled by Eq.(1) one moment and controlled by Eq.(2) the next, when R is small enough, in most of the iteration loop, the movement trajectory of the particles are controlled by iteration Eq.(1), Eq.(2) just disturb occasionally to make the particle swarm's convergent extent decrease temporally so as to achieve mutation effect. But Eq.(1) can make the particle swarm's convergent extent increase in the subsequent iteration.

With the increasing of R , the disturbance of Eq.(2) becomes more and more frequent, the convergent effect of Eq.(1) is counteracted by the disturbance of Eq.(2), the particle swarm's convergent will not increase with the continuation of the iteration,

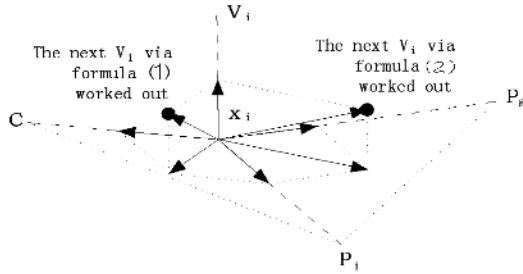


Fig. 1. Next V_i calculated by formulae (1), (2) respectively

the whole particle swarm tend towards divergence, the quality of solution will decrease. So the value of R is very important to PSOWC-Pg, it determines whether the algorithm is convergent or divergent, if the algorithm is convergent, R will determine the convergent speed, when R is bigger, the effect of disturbance is more frequent, convergent speed is slow, the iteration times needed for convergence is more, on the contrary, when R is smaller, the frequency of disturbance is lower, the speed of convergence is fast, the iteration times needed for convergence is less, too small R will make the mutation can't achieve corresponding effect, and the mutation goal can't reach. But it is not easy to determine an appropriate R so as to get the best solution, it is pertinent to the size of swarm, fitness function, x_{max} and v_{max} , it is very difficult to find a formula to determine R that can achieve the best solution, maybe, such formula doesn't exist at all, the feasible method is to treat the R as an empirical value, under the condition that parameters such as the amount of particles in the particle swarm, fitness function, x_{max} , v_{max} are determined, we can run the program multiple times and assign different values to R in each run, through the comparison of superiority and inferiority of the running results, we can accumulate experience and achieve a best value of R . Table 1 presents the best values of R for twelve types of algorithm applying to four benchmark functions, they are all achieved via this method.

In PSOWC-Pg, there are two sources of disturbing point C : the current position x and the historically optimal position $pBest$, with local version of PSOWC-Pg, they can be divided into x or $pBest$ within the neighbor particles and x or $pBest$ within the whole particles. There are two methods for selecting C : stochastic method and the worst fitness method. The stochastic method selects some or other particle's current position x or $pBest$ as C stochastically in each iteration, the worst fitness method selects x or $pBest$ of the particle with the worst fitness value as C . So, we can get twelve different C (or twelve different types of PSOWC-Pg algorithm), the global version of PSOWC-Pg (gPSOWC-Pg) has four types: stochastic x , stochastic $pBest$, the worst fitness x , and the worst fitness $pBest$ which are all within the range of whole particles. The local version of PSOWC-Pg (lPSOWC-Pg) has eight types: stochastic x , stochastic $pBest$, the worst fitness x and the worst fitness $pBest$ which are within the range of the whole particles and are called global disturbing point, stochastic x , stochastic $pBest$, the worst fitness x and the worst fitness $pBest$ which are within the range of the neighbor particles and are called local disturbing point.

Table 1. Best R of four benchmark functions applied by twelve types of PSOWC-Pg

		Function	Stochastic x	Stochastic $pBest$	Worst fitness x	Worst fitness $pBest$
gPSOWC-Pg		Griewank	0.1	0.13	0.075	0.8
		Schaff f6	0.4	0.35	0.15	0.7
		Ackley	0.1	0.1	0.06	0.25
		Rastrigin	0.2	0.1	0.03	0.2
IPSOWC-Pg	Local disturbing point	Griewank	0.1	0.1	0.06	0.08
		Schaff f6	0.18	0.2	0.08	0.2
		Ackley	0.008	0.03	0.004	0.01
		Rastrigin	0.004	0.005	0.009	0.009
	Global disturbing point	Griewank	0.013	0.05	0.016	0.013
		Schaff f6	0.05	0.08	0.09	0.05
		Ackley	0.009	0.028	0.011	0.009
		Rastrigin	0.0015	0.001	0.0012	0.0015

The mutating particles are controlled by Eq.(2), so, they tend to be close to C , but in the iterations between two adjacent mutation, particle swarm are controlled by Eq.(1), each particle tends to be close to Pg (see Fig.1), whether be stochastic method or the worst fitness method, even in the same iteration loop the calculated C maybe comes from $pBest$ or x of different particles, so, different particles will come close to their own different Cs in the mutation, when no mutation, they will come close to the same point Pg from different directions. If particle swarm converge, $pBest$, x and $gBest$ will converge to the same point together, also because C comes from $pBest$ and x , so, C will converge to this point too, this determines that with the proceeding of the iterations, $|CPg|$ will become smaller and smaller, and tends towards zero, where $|CPg|$ is the distance between C and $gBest$. So, on the condition that particle swarm converge, the whole searching scope of particle swarm will be reduced around $gBest$ slowly and irregularly, the variation of the fluctuating range is similar to that of $|CPg|$, and eventually converge to $gBest$; meanwhile, the particle swarm's convergent speed increases slowly and irregularly; but searching density that is the size of population in unit region will increase slowly and irregularly.

Compared with the original PSO, the convergent speed of PSOWC-Pg is slower in the early period of the searching loop, the speed of the reduction of the searching scope is much slower, but the strength of the search is much more bigger, so it is more easier to find global optimum of the target functions. In the later period of the searching process, $|CPg|$ tend towards zero, the PSOWC-Pg approximates to the original PSO, and this is favorable to improving precision of solution. Compared with the traditional PSO, PSOWC-Pg needn't determine the mutation opportunity, mutations may take place in the whole searching process, mutations are uninterrupted, but the mutation measures adopted by traditional PSO are discontinuous, and it takes place when particle swarm converge to some extent. Mutation measures taken by traditional mutation reinitialize the part particles of swarm, that is, mutation takes place on each dimension of some particles, this destroys the structure of the particle swarm, but with PSOWC-Pg, mutation only takes place on one dimension of some particle, mutation

is distributed evenly in the whole searching process, any one of mutation doesn't cause the structure of the particle swarm damaged, but can achieve the goal of mutation, so, PSOWC-Pg is more effective.

Table 2. Other test parameters

Function	$v_{\max} = x_{\max}$	Number of dimensions	Position of the global minimums
Schaffer f6	100	2	(0,0)
Ackley	30	30	(0,0,...,0)
Griewank	600	30	(0,0,...,0)
Rastrigin	5.12	30	(0,0,...,0)

3 Experiment and Discussion of the Results

Three methods which determine the superiority or inferiority of the algorithm have been proposed in [3], in this paper, two of them will be used, one is the speed of searching optimum, it reflects an average iteration times that needed by an algorithm to make each benchmark function reach its own global optimum, the less the times, the faster the speed of the searching optimum, and the better the algorithm. Another one is the success ratio. Computing the sum of the times which an algorithm makes all benchmark functions reach their own global optimum after 10,000 iteration times, and get the ratio of the sum to total running times, obviously, the bigger ratio shows higher ratio of success, and the algorithm is better.

In this paper, the following benchmark functions will be used: Ackley, Schaffer f6, Griewank, Rastrigin, their expressions are in [4], these four functions all are multi-modal functions that have multiple local minimum, their values of R are listed in Table 1, other test parameters are listed in Table 2, in this paper, the contrastive experiments of above mentioned twelve types of PSOWC-Pg, and local version and global version of original PSO and PSO with traditional mutation, will be done. The size of the population for all the algorithms is 10; each algorithm runs fifty times on each benchmark function, the ratio of success and the speed of searching optimum of each algorithm are listed in Table 3 and Table 4.

Experiment results show that different types of PSO of local version are superior to global version no matter for the perspective of success ratio or for convergent speed. The original PSO of local version is superior to original PSO of global version; traditional mutation PSO of local version is superior to PSO with traditional mutation of global version, these are results of many other papers, they are also supported by our experiments results. Eight types of IPSOWC-Pg are all superior to four types of gPSOWC-Pg.

No matter we observe from the perspective of success ratio or from the convergent speed, with the local version and global version, it is PSOWC-Pg that are superior to traditional mutation PSO, and traditional mutation PSO are superior to original PSO. This illustrates that mutation is the effective measure to promote performance of the PSO algorithm, that the efficiency of mutation of PSOWC-Pg is greater than that of

traditional mutation is the result of the above analysis; it is supported by our experiment results.

Whether be gPSOWC-Pg or IPSOWC-Pg, selection of different disturbing point has little effect on success ratio, but has great impact on the convergent speed, with PSOC-Pg of global version, the difference of fastest convergent speed(1081) and slowest convergent speed(1513) comes to 40%, and it has the most difference among the three types of PSOWC-Pg.

Table 3. Experiment results of global version PSO

Algorithm	Original PSO	PSO with traditional mutation	gPSOWC-Pg			
			Stochastic x	Stochastic $pBest$	Worst fitness x	Worst fitness $pBest$
Success ratio(%)	46	90	97.5	97.5	96.5	97.5
Speed of searching optimum	∞	1701	1459	1127	1513	1081

Table 4. Experiment results of local version PSO

Algorithm	Original PSO	PSO with traditional mutation	IPSOWC-Pg(local disturbing point)			
			Stochastic x	Stochastic $pBest$	Worst fitness x	Worst fitness $pBest$
Success ratio(%)	92	93	99	98	98	98.5
Speed of searching optimum	909	720	741	706	713	746
Algorithm	—	—	IPSOWC-Pg(global disturbing point)			
success ratio(%)	—	—	99	98.5	99	98.5
Speed of searching optimum	—	—	557	516	650	525

The IPSOWC-Pg of global disturbing point are all superior to IPSOWC-Pg of local disturbing point, this can be explained with viewpoints from [3], global disturbing point contains information of all the particles, this information can be used by all the particles of IPSOWC-Pg of global disturbing point, but with IPSOWC-Pg of local disturbing point, only particles within the neighbor of particle corresponding to global disturbing position can use them directly, other particles can only use these global

information after some delay, so cause its efficiency lower than that of IPSOWC-Pg of global disturbing point.

Observed from another perspective, IPSOWC-Pg of local disturbing point make the searching scope of each particle be limited by the position that its neighbor particle(containing itself) can search , while, when mutation, IPSOWC-Pg of global disturbing point can make each particle's searching scope reach the searching scope that all the particles can search, when no mutation, it doesn't lose its advantages of local version PSO, so its performance is more superior to that of IPSOWC-Pg of local disturbing point.

4 Conclusion

The ideas of PSOWC-Pg proposed in this paper are very simple, and easy to implement, compare to original PSO and PSO with traditional mutation, PSOWC-Pg promotes performance greatly, but superiority and inferiority of performance of the algorithm depends on the mutation ratio R strongly, but the best R can only be determined by trail computation, this is the common weakness of this type of algorithm, if R can be determined automatically by the algorithm, repeated trail computation can be avoided, or we can settle for second best to achieve the law of the relation between value R and performance of the superiority and inferiority of the algorithm during the process of the trail computation, and exploit the law to reduce the times of the trail computation as much as possible, and it is the direction of future work.

References

1. Kennedy. J, Eberhart. R. C: Particle Swarm Optimization. In: Proceeding of the IEEE International Conference on Neural Networks, Vol IV, Perth, Australia(1995)1942–1948.
2. Clerc. M, Kennedy. J: The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space. In: IEEE TRANSACTION ON EVOLUTIONARY COMPUTER. Vol.6, NO.1, (2002)58-73.
3. Mendes. R, Kennedy. J, José. Neves: The Fully Informed Particle Swarm: Simpler, Maybe Better. In: IEEE Trans on Evolutionary Computation, Vol 8, No. 3, pp. 204–210, Jun. 2004.
4. Eberhart. R.C, Shi. Y: Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization. In: Proceedings of the Congress on Evolutionary Computing, San Diego.(2000)84–89.

Algal Bloom Prediction with Particle Swarm Optimization Algorithm

K.W. Chau

Department of Civil and Structural Engineering, Hong Kong Polytechnic University,
Hungghom, Kowloon, Hong Kong, China
cekwchau@polyu.edu.hk

Abstract. Precise prediction of algal booms is beneficial to fisheries and environmental management since it enables the fish farmers to gain more ample time to take appropriate precautionary measures. Since a variety of existing water quality models involve exogenous input and different assumptions, artificial neural networks have the potential to be a cost-effective solution. However, in order to accomplish this goal successfully, usual problems and drawbacks in the training with gradient algorithms, i.e., slow convergence and easy entrapment in a local minimum, should be overcome first. This paper presents the application of a particle swarm optimization model for training perceptrons to forecast real-time algal bloom dynamics in Tolo Harbour of Hong Kong, with different lead times on the basis of several input hydrodynamic and/or water quality variables. It is shown that, when compared with the benchmark backward propagation algorithm, its results can be attained both more accurately and speedily.

1 Introduction

Owing to the semi-enclosed nature and the nutrient enrichment through municipal and livestock waste discharges in Tolo Harbour, frequent algal blooms with occasional massive fish kills have been recorded over the past two decades. Precise prediction of algal booms is beneficial to fisheries and environmental management since it enables the fish farmers to gain more ample time to take appropriate precautionary measures. However, the causality and dynamics of algal blooms, which are related to various pertinent factors such as time-lagged chlorophyll-a, secchi disc depth, nitrogen, phosphorus, dissolved oxygen, rainfall, water temperature, solar radiation, wind speed, tidal range, and so on, are extremely complicated and not well-understood.

Existing water quality models require exogenous input and embrace different assumptions. In numerical modeling, the physical problem is represented by a highly coupled, non-linear, partial differential equation set. The involving processes are highly complex and uncertain which may demand huge computing cost and time. Process-based models are not completely satisfactory in representing the highly complex inter-relationships. Recently, owing to various advantages (built-in dynamism, data-error tolerance and no need to have exogenous input), artificial neural networks (ANN), and in particular, the feed forward back-propagation (BP)

perceptrons, have been widely applied in water resources engineering [1]. However, slow training convergence speed and easy entrapment in a local minimum are inherent drawbacks of the commonly used BP algorithm.

In this paper, the particle swarm optimization (PSO) algorithm is employed to train multi-layer perceptrons for algal bloom prediction in Tolo Harbour of Hong Kong with different lead times and input variables.

2 Attributes of PSO Algorithm

During its inception, PSO algorithm is developed as a tool for modeling social behavior and is able to optimize hard numerical functions [2-3]. It is currently adapted as a computational intelligence technique intimately related to evolutionary algorithms [4]. It is an optimization paradigm that mimics the ability of human societies to process knowledge. It has roots in two main component methodologies: artificial life on bird swarming; and, evolutionary computation. The PSO technique has been applied in hydrological problems and accomplished satisfactory results [5-7]. Its principle is founded on the assumption that potential solutions will be flown through hyperspace with acceleration towards more optimum solutions. PSO is a populated search method for optimization of continuous nonlinear functions resembling the movement of organisms in a bird flock or fish school. Each particle adjusts its flying according to the flying experiences of both itself and its companions. In doing so, it keeps track of its coordinates in hyperspace which are associated with its previous best fitness solution, and also of its counterpart corresponding to the overall best value acquired thus far by any other particle in the population. Vector, as a convenient form for optimization problems, is used as the variable presentation to represent particles.

The principal advantages of PSO are relatively simple coding and hence computationally inexpensive. A similarity between PSO and a genetic algorithm is the initialization of the system with a population of random solutions and the employment of the fitness concept. However, the evolution of generations of a population of these individuals in such a system is by cooperation and competition among the individuals themselves. The population is responding to the quality factors of the previous best individual values and the previous best group values. The allocation of responses between the individual and group values ensures a diversity of response. The principle of stability is adhered to since the population changes its state if and only if the best group value changes. It is adaptive corresponding to the change of the best group value. The capability of stochastic PSO algorithm to determine the global optimum with high probability and fast convergence rate has been shown in other cases. PSO is adopted to train the multi-layer perceptrons here.

3 Application in Network Training

For a three-layered perceptron, $W^{[1]}$ and $W^{[2]}$ represent the connection weight matrix between the input layer and the hidden layer, and that between the hidden layer and

the output layer, respectively. During training of the multi-layer preceptrons, the i -th particle is denoted by $W_i = \{W^{[1]}, W^{[2]}\}$ whilst the velocity of particle i is denoted by V_i . The position representing the previous best fitness value of any particle is denoted by P_i whilst the best matrix among all the particles in the population is recorded as P_b . Let m and n represent the index of matrix row and column, respectively, the following equation represents the computation of the new velocity of the particle based on its previous velocity and the distances of its current position from the best experiences both in its own and as a group.

$$V_i^{[j]}(m, n) = V_i^{[j]}(m, n) + r\alpha[P_i^{[j]}(m, n) - W_i^{[j]}(m, n)] + s\beta[P_b^{[j]}(m, n) - W_i^{[j]}(m, n)] \tag{1}$$

where $j = 1, 2$; $m = 1, \dots, M_j$; $n = 1, \dots, N_j$; M_j and N_j are the row and column sizes of the matrices W , P , and V ; r and s are positive constants; α and β are random numbers in the range from 0 to 1. In the context of social behavior, the cognition part $r\alpha[P_i^{[j]}(m, n) - W_i^{[j]}(m, n)]$ represents the private thinking of the particle itself whilst the social part $s\beta[P_b^{[j]}(m, n) - W_i^{[j]}(m, n)]$ denotes the collaboration among the particles as a group. The new position is then determined based on the new velocity as follows.

$$W_i^{[j]} = W_i^{[j]} + V_i^{[j]} \tag{2}$$

The following equation is used to determine the fitness of the i -th particle in term of an output mean squared error of the neural networks

$$f(W_i) = \frac{1}{S} \sum_{k=1}^S \left[\sum_{l=1}^O \{t_{kl} - p_{kl}(W_i)\}^2 \right] \tag{3}$$

where f is the fitness value, t_{kl} is the target output; p_{kl} is the predicted output based on W_i ; S is the number of training set samples; and, O is the number of output neurons.

4 Prototype Prediction

A necessary procedure to test the usefulness of a system is to verify its capability to mimic a particular case study with accurate depiction of real phenomena. This system has been verified and validated by applying to predict the algal bloom dynamics in Tolo Harbour of Hong Kong [8-14]. It is observed that the life cycle of algal blooms is in the order of 1 to 2 weeks. Thus, algal biomass, represented as chlorophyll-a, is forecasted with a lead time of 1 and 2 weeks based on comprehensive biweekly water quality data at Tolo Harbour from year 1982 to year 2002. The data of 1982-1995 and those of 1996-2002 are used for training and testing/validation, respectively. They are so chosen such that extreme frequency and intensity are contained in both data sets. Depth-averaged values, derived from the data for surface, mean and bottom of

the water column, are adopted for analysis purpose. Ten input variables are generally considered to be significant on the algal dynamics of Tolo Harbour [15-17].

Hence, different perceptrons are tried which have an input layer with from one to ten neurons, a hidden layer with three to five neurons, and an output layer with one neuron. The input neurons represent the combinations of the time-lagged chlorophyll-a (Chl-a), phosphorus (PO_4), nitrogen (TIN), dissolved oxygen (DO), secchi disc depth (SD), rainfall, water temperature, solar radiation, wind speed, tidal range whilst the output node represents chlorophyll-a. Three scenarios are attempted with 10, 5 and 1 input variables for scenario 1, 2, and 3, respectively. Figure 1 shows the general network for 1 week lead time chlorophyll-a prediction for scenario 1. All source data are normalized into the range between 0 and 1, by using the maximum and minimum values of the variable over the whole data sets. The number of population is set to be 30 whilst the maximum and minimum velocity values are 0.3 and -0.3 respectively.

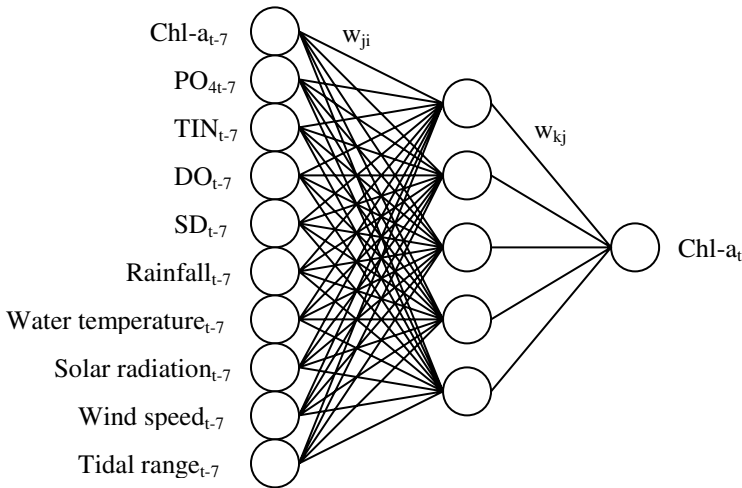


Fig. 1. General network for 1 week lead time chlorophyll-a prediction for scenario 1

5 Analysis of Results

The performance of the PSO-based multi-layer ANN is evaluated in comparison with the benchmarking standard BP-based network. In order to provide a fair and common initial ground for comparison purpose, the training process of the BP-based perceptron commences from the best initial population of the corresponding PSO-based perceptron. Table 1 shows comparison of the results for chlorophyll-a forecasting with both 1 week and 2 weeks lead times for scenarios 1 to 3. It should be noted that the results do not exhibit a significant advantage of using more environmental variables as the network inputs and that 1 week lead time is better than its counterparts of 2 weeks. Moreover, the PSO-based perceptron exhibits much faster convergence and better prediction ability than those by the BP-based perceptron.

Table 1. Results for chlorophyll-a forecasting based on scenarios 1 to 3

Input data	Algorithm	Coefficient of correlation				Convergence time (s)
		Training		Validation		Training
		1 week ahead	2 weeks ahead	1 week ahead	2 weeks ahead	1 week ahead
Scenari 1	BP-based	0.964	0.935	0.947	0.924	213
	PSO-based	0.988	0.977	0.968	0.963	182
Scenari 2	BP-based	0.962	0.934	0.948	0.924	80
	PSO-based	0.986	0.978	0.969	0.959	59
Scenari 3	BP-based	0.961	0.938	0.953	0.923	55
	PSO-based	0.989	0.976	0.966	0.961	38

6 Conclusions

In this paper, a perceptron approach based on the PSO paradigm is employed for real-time prediction of algal blooms at Tolo Harbour in Hong Kong with different lead times and input variables. The results do not exhibit any advantage of using more environmental variables as the network inputs. The chlorophyll-a output from the 1 week time-lagged chlorophyll-a input can be a robust forewarning and decision-support tool. The results also show that the PSO-based perceptron consistently performs better than the benchmarking BP-based perceptron in algal bloom prediction, in term of both accuracy and speed.

Acknowledgement

This research was supported by the Research Grants Council of Hong Kong (PolyU5132/04E).

References

1. Chau, K.W., Cheng, C.T.: Real-time Prediction of Water Stage with Artificial Neural Network Approach. *Lecture Notes in Artificial Intelligence*, **2557** (2002) 715-715
2. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. *Proceedings of the 1995 IEEE International Conference on Neural Networks*. Perth (1995) 1942-1948
3. Kennedy, J.: The Particle Swarm: Social Adaptation of Knowledge. *Proceedings of the 1997 International Conference on Evolutionary Computation*. Indianapolis (1997) 303-308
4. Clerc, M., Kennedy, J.: The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space. *IEEE Transactions on Evolutionary Computation* **6**(1) (2002) 58-73
5. Chau, K.W.: River Stage Forecasting with Particle Swarm Optimization. *Lecture Notes in Computer Science* **3029** (2004) 1166-1173
6. Chau, K.W.: Rainfall-Runoff Correlation with Particle Swarm Optimization Algorithm. *Lecture Notes in Computer Science* **3174** (2004) 970-975

7. Slade, W.H., Ransom, H.W., Musavi, M.T., Miller R.L.: Inversion of Ocean Color Observations using Particle Swarm Optimization. *IEEE Transactions on Geoscience and Remote Sensing* **42(9)** (2004) 1915-1923
8. Chau, K.W.: Manipulation of Numerical Coastal Flow and Water Quality Models. *Environmental Modelling & Software* **18(2)** (2003) 99-108
9. Chau, K.W.: Intelligent Manipulation of Calibration Parameters in Numerical Modeling. *Advances in Environmental Research* **8(3-4)** (2004) 467-476
10. Chau, K.W.: A Three-Dimensional Eutrophication Modeling in Tolo Harbour. *Applied Mathematical Modelling* **28(9)** (2004) 849-861
11. Chau, K.W.: Selection and Calibration of Numerical Modeling in Flow and Water Quality. *Environmental Modeling and Assessment* **9(3)** (2004) 169-178
12. Chau, K.W., Chen, W.: A Fifth Generation Numerical Modelling System in Coastal Zone. *Applied Mathematical Modelling* **25(10)** (2001) 887-900
13. Chau, K.W., Chen, W.: An Example of Expert System on Numerical Modelling System in Coastal Processes. *Advances in Engineering Software* **32(9)** (2001) 695-703
14. Chau, K.W., Cheng, C., Li, C.W.: Knowledge Management System on Flow and Water Quality Modeling. *Expert Systems with Applications* **22(4)** (2002) 321-330
15. Chau, K.W., Jin, H.S.: Eutrophication Model for a Coastal Bay in Hong Kong. *Journal of Environmental Engineering ASCE* **124(7)** (1998) 628-638
16. Chau, K.W., Jin, H.S., Sin, Y.S.: A Finite Difference Model of Two-Dimensional Tidal Flow in Tolo Harbor, Hong Kong. *Applied Mathematical Modelling* **20(4)** (1996) 321-328
17. Jin, H.S., Egashira, S., Chau, K.W.: Carbon to Chlorophyll-a Ratio in Modeling Long-Term Eutrophication Phenomena. *Water Science and Technology* **38(11)** (1998) 227-235

Synthesis of the Antenna Array Using a Modified Particle Swarm Optimization Algorithm^{*}

Tengbo Chen, Yong-Chang Jiao, and Fushun Zhang

National Laboratory of Antennas and Microwave Technology,
Xidian University, Xi'an 710071, China
ctb210@sina.com, ychjiao@xidian.edu.cn

Abstract. The particle swarm optimization algorithm presents a new way for finding an optimal solution of complex optimization problems, where each particle represents a solution to the problem. In this paper a modified particle swarm optimization algorithm is applied to the optimization of the antenna array. Adding an item of integral control and the contractive factor in the modified algorithm can improve its global search ability. Simulation results show that the optimal pattern of the antenna array is able to approach the desired pattern. The results also demonstrate that the modified algorithm is superior to the original algorithm and the nonlinear least-square method.

1 Introduction

The antenna arrays have been widely used in phase array radar, satellite communications and other domains [1]. In the satellite communications, in order to make the antenna array resist interference and noise, the array pattern should possess lower side-lobes, controllable beam width and good azimuthal symmetry. Traditional nonlinear least-square (NLS) method [2] cannot bear the demand of such complex optimization problem. Several evolutionary algorithms, such as genetic algorithms, can be used to solve such kind of problems, and obtain ideal side-lobe envelope. However these methods have heavy computational burden and hence are time-consuming. Particles Swarm Optimization (PSO) is an evolutionary algorithm based on the swarm intelligence (e.g. [3] [4]), which can be used to solve complex global optimization problems. Currently, the algorithm and its variations are applied to many practical problems (e.g. [5] [6]).

A modified PSO (MPSO) algorithm is introduced and adopted to optimize the multi-element hexagon antenna array. The array parameters used to determine the array pattern are chosen as the design variables. A desired pattern is given in advance according to the engineering requirement. The simulation results show that the optimal pattern with very low side-lobes approaches the desired pattern very well. The synthesis result greatly improves the efficiency of the antenna array.

^{*} This work was supported by the National Natural Science Foundations of China (60171045, 60374063 and 60133010).

2 The Particle Swarm Optimization Algorithm

2.1 Analysis of PSO

In PSO algorithms, each potential solution of the D -dimensional optimization problem is a bird in the solution space, which called “particle”. Each particle has a value of fitness determined by the objective function. The particle also has a directional velocity to control its move tracks. The particles chase the optimal solution by searching the solution space. All particles have initial positions and velocities, where the positions and velocities are iterated. At each generation, two “best positions” are chased to update the particle. The first is the optimal solution found by particle, which called personal best position. The other, called global best position, is the optimal solution in the entire population. The i th particle in the solution space is determined by a fitness function value. Denote the position of i th particle by $x_i = (x_{i1}, \dots, x_{id}, \dots, x_{iD})$, and the velocity of the i th particle by $v_i = (v_{i1}, \dots, v_{id}, \dots, v_{iD})$. The positions and velocities of particles update by personal and global best positions at each generation. Let $p_i = (p_{i1}, \dots, p_{id}, \dots, p_{iD})$ be the position vector for an individual particle’s best fitness, which is the personal best position, and $g_i = (g_{i1}, \dots, g_{id}, \dots, g_{iD})$ be the global best position among all the agents. The positions and velocities of particles are updated according to the following equations:

$$v_{id} = w * v_{id} + c_1 * rand * (p_{id} - x_{id}) + c_2 * rand * (g_{id} - x_{id}), \quad (1)$$

$$x_{id} = x_{id} + v_{id}, \quad (2)$$

where w is the inertia weight, c_1 and c_2 are the learn factors, and $rand$ is a random number in the range $[0,1]$. The first part of (1) represents the previous velocity, which provides the necessary momentum for particles to roam across the search space. The second part, known as the “cognitive” component, represents the personal thinking of each particle. The cognitive component encourages the particles to move toward their own best positions found so far. The third part is known as the “social” component, which represents the collaborative effect of the particles, in finding the global optimal solution. The social component always pulls the particles toward the global best particle found so far. We suggest to choose $w = 0.7$, $c_1 = c_2 = 2.0$.

2.2 A Modified PSO Algorithm

The original PSO algorithm has a high convergence speed, but it is easy to fall into local optima. In order to overcome this weakness, some improvements are applied to the origin PSO algorithm. One of such effective strategies is adding an item of integral control into (2). Then, the particle’s position equation (2) is replaced by

$$x_{id} = x_{id} + \chi v_{id} + \tau(I - f_i), \quad (3)$$

where χ is the contractive factor used to control the velocity, $\tau \geq 0$ is the integral factor, I is a constant such that $I \leq f_{min}$, f_{min} stands for the minimum fitness

value. The item of integral control can improve the performance of the original PSO algorithm. When $f_i = I$, the item is zero. In other cases $I - f_i < 0$, and the item of integral control can make x_{id} become smaller and escape from the local minimum. If τ is a fixed value, the item does not work well. Thus at each generation the value of τ is determined according to

$$\tau = \begin{cases} \tau_0 / \max_i(|f_i|), & \text{if } \min_i(|f_i|) > c \\ \tau_0, & \text{otherwise} \end{cases}$$

where τ_0 and c are all positive numbers. We suggest to choose $\chi = 0.729$, $c = 1$. For the least-square problems, we choose $I = 0$.

3 Optimization Model of Antenna Array

One of the features of the hexagon antenna array is that most elements surround a center element [1]. The elements locate in a two-dimensional plane. Being the array element, the conic horn antenna has a good directional characteristic, thus the mutual coupling among the elements is weak, which is neglected here. According to the pattern multiple principle, the far-field pattern can be expressed as the multiple of the array factor with the pattern of the element. Denote the aperture radius of the conic horn by c , the length of the conic horn by R , the spacing between elements by b , and the array element pattern by $f(\theta, \varphi)$. In the optimization of 37-element hexagon antenna array, the parameters, including the amplitude and phase of the elements, b, c as well as R , are chosen as the design parameters. Therefore, the pattern of the antenna array can be expressed as

$$E(x, \theta, \varphi) = \left\{ \sum_{i=-N_1}^0 \sum_{n=-N_1-i}^{N_1} + \sum_{i=1}^{N_1} \sum_{n=-N_1}^{N_1-i} \right\} a_{i,n} e^{j p_{i,n}} f(\theta, \varphi), \quad (4)$$

where $a_{i,n} e^{j p_{i,n}}$ is the normalized complex weight of element (i, n) , and $j = \sqrt{-1}$. In (4), $N_1 = 3$ represents the number of the hexagon loop around the center, and x is expressed as

$$x = (a_{-3,0}, \dots, a_{0,3}, a_{1,-3}, \dots, a_{3,0}, p_{-3,0}, \dots, p_{0,3}, p_{1,-3}, \dots, p_{3,0}, b, c, R)^T. \quad (5)$$

According to the engineering requirements, we choose the desired pattern as $A(\theta) = \cos^2(\theta)$, which is independent of φ . The region $\theta \in [-\pi/2, \pi/2]$, $\varphi \in [0, 2\pi]$ is discretized by $M_1 \times M_2$ discrete points:

$$\theta_i = -\frac{\pi}{2} + \frac{(i-1)\pi}{M_1-1}, \quad i = 1, 2, \dots, M_1,$$

$$\varphi_j = \frac{(j-1)2\pi}{M_2-1}, \quad j = 1, 2, \dots, M_2.$$

Comparing the calculated pattern point-by-point with the desired pattern, we express the fitness function as

$$Fitness(x) = \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} |E(x, \theta_i, \phi_j) - A(\theta_i)|^2. \tag{6}$$

The design parameters should satisfy the following simple constraints:

$$0 \leq a_{i,n} \leq 1, \quad 0 \leq p_{i,n} \leq 2\pi, \quad 2c < b.$$

4 Simulation and Discussion

4.1 Simulation Results

The conic horn pattern $f(\theta, \varphi)$ is calculated according to the formulas given [1]. We choose $M_1 = 181$, $M_2 = 91$. The MPSO algorithm is used to optimize the antenna array. In the MPSO algorithm, we choose $\tau_0 = 0.1$. The population size is sent to 85, and the algorithm stops after 2000 generations. The optimal solution we found is $c = 0.41\lambda$, $R = 2.17\lambda$, and $b = 0.85\lambda$. When the MPSO algorithm stops, most particles have the identical fitness value, thus the solution we found is feasible. E-plane and H-plane array patterns yielded by the optimal solution are shown in Figures 1 and 2, respectively. The side-lobe level (SLL) and the half-power beamwidth (HPBW) of the array pattern in four φ cuts are listed in Table 1. It is clear that the variation of SLL and HPBW with φ is small. Therefore the azimuthal symmetry of the array pattern is very well.

Table 1. Parameters of the array pattern in four φ cuts

φ (degree)	0	90	180	270
HPBW (degree)	11.48	11.41	11.47	11.39
SLL (dB)	-32.5	-31.4	-32.1	-31.7

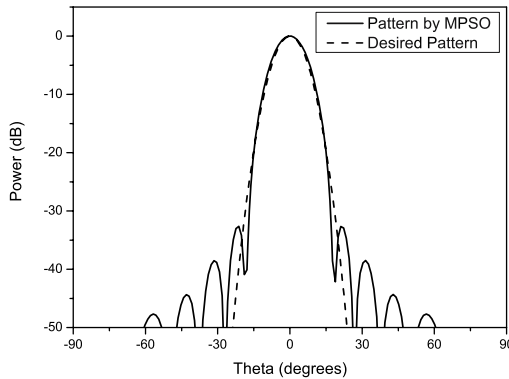


Fig. 1. E-plane pattern of the antenna array

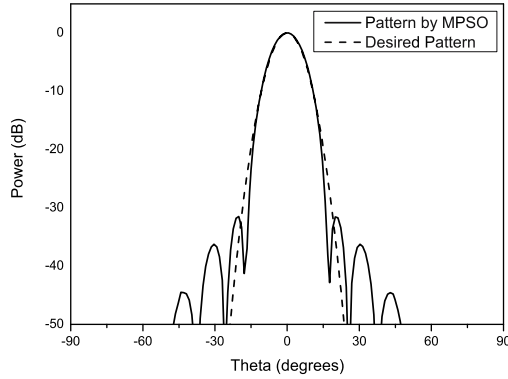


Fig. 2. H-plane pattern of the antenna array

4.2 Verification of the PSO Algorithm

In order to illustrate the efficiency of the PSO algorithm, the nonlinear least-square (NLS) method and the original PSO algorithm are used to optimize the same antenna array. The minimum fitness values found by these three algorithms are presented in Table 2. Variation of the fitness value over 2000 generations for the MPSO algorithm and the original PSO algorithm is compared in Figure 3.

Table 2. Minimum fitness values founded by three algorithms

Algorithm	NLS	PSO	MPSO
Fitness value	2.5484	2.3563	0.5475

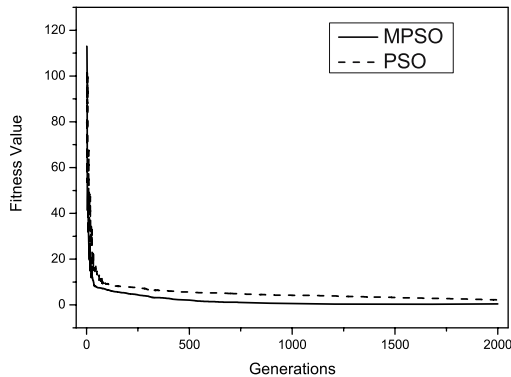


Fig. 3. Variation of the fitness value over 2000 generations for the MPSO algorithm and the original PSO algorithm

The results show that the MPSO algorithm is superior to the NLS method and the original PSO algorithm.

5 Conclusion

The mathematical expression of the MPSO algorithm is very simple, and its concept is intelligible. Such algorithm is feasible for solving complex global optimization problems. The hexagon antenna array pattern is synthesized by using the MPSO algorithm, and the design requirements are satisfied. The simulated results show that the MPSO algorithm converges very fast and is easy to use. The algorithm can also be used to solve other kinds of optimization problems in electromagnetics.

References

1. Balanis, C.A.: *Antenna Theory: Analysis and Design*. Second Edition. John Wiley and Sons, Inc., New York (1997) 309-337
2. Dennis, J.E., Jr.: *Nonlinear Least Squares*. In: Jacobs, D. (Ed.): *State of the Art in Numerical Analysis*. Academic Press, New York (1977) 269-312
3. Clerc, M., Kennedy, J.: *The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space*. *IEEE Transactions on Evolutionary Computation* **6** (2002) 58-73
4. Parsopoulos, K.E., Vrahatis, M.N.: *On the Computation of All Global Minimizers Through Particle Swarm Optimization*. *IEEE Transactions on Evolutionary Computation* **8** (2004) 211-224
5. Boeringer, D.W., Werner, D.H.: *Particle Swarm Optimization Versus Genetic Algorithms for Phased Array Synthesis*. *IEEE Transactions on Antennas and Propagation* **52** (2004) 771-779
6. Robinson, J., Rahmat-Samii, Y.: *Particle Swarm Optimization in Electromagnetics*. *IEEE Transactions on Antennas and Propagation* **52** (2004) 397-407

An Ant Colony Optimization Approach to the Degree-Constrained Minimum Spanning Tree Problem

Y.T. Bau, C.K. Ho, and H.T. Ewe

Multimedia University, Faculty of Information Technology, Jalan Multimedia,
63100 Cyberjaya, Selangor, Malaysia
{ytbau, ckho, htewe}@mmu.edu.my

Abstract. This paper presents the application of an Ant Colony Optimization (ACO) algorithm approach for communications networks design problem. We explore the use of ACO's for solving a network optimization problem, the degree-constrained minimum spanning tree problem (d-MST), which is a NP-Hard problem. The effectiveness of the proposed algorithm is demonstrated through two kinds of data set: structured hard (SHRD) complete graphs and misleading (M-graph) complete graphs. Empirical results show that ACO performs competitively with other approaches based on evolutionary algorithm (EA) on certain instance set problem.

1 Introduction

The ant colony optimization (ACO) metaheuristic algorithm models the behaviour of each artificial ant after the foraging behaviour of real ants [1]. An ant, while going from the colony to the food source deposits a chemical substance, called pheromone. When the ant returns from the food source, it reinforces the pheromones on the path that it had used. Pheromones are used to attract other ants to follow a particular path. When a large number of ants forage for food, the minimum cost path to the food source will eventually contain the highest concentration of pheromones, thereby attracting all the ants to use that minimum cost path, and for higher cost path pheromone trails progressively decrease by evaporation. ACO algorithms are modelled after this behaviour, and have been used to solve minimum cost path problems and problems that can be reduced to a kind of shortest path problems.

This work uses an ACO algorithm to find the degree-constrained minimum spanning tree(d-MST) in communications networks. d-MST problem is obtained by the modification of the MST problem from given a connected, edge weighted, undirected graph G , and the minimum-weight spanning tree such that no vertex of the spanning tree has degree greater than d is NP-Hard problem. It is because when $d=2$, the MST meeting the constraint will take the form of a path. This is the path of least total weight which includes every vertex in the graph. In other words, this is a Hamiltonian path. Hence an algorithm which solves the d-MST problem also solves the Hamiltonian path problem, which is NP-complete. Therefore, the d-MST is NP-hard.

The d-MST problem was first studied by Deo and Hakimi in 1968 [3,5]. Since computing a d-MST is NP-Hard for every d in the range $2 \leq d \leq |V|-1$, a few heuristics had been introduced to solve the d-MST problem such as genetics algorithms[8], problem

space search[4], simulated annealing[4], langrangean relaxation[4], branch and bound[4], parallel algorithms[3] and evolutionary algorithms[2,6]. The d-MST problem has also been studied for the complete graphs of points in a plane where edge costs are the Euclidean distance between these points coordinate. Euclidean problems are relatively simple to solve. Using exact algorithms such as branch and bound and langrangean relaxation as described by Krishnamorrthy et al. [4] can find optimal solutions even for large problem instances including several hundred vertices in polynomial time. This showed that there exist effective polynomial-time heuristics for finding d-MST in the plane. In more general case, the costs associated with the graph's edge are arbitrary and need not satisfy the triangle inequality. In this case, a MST may have degree up to $|V|-1$. Computing a d-MST such a high-degree MST is usually a hard task especially in structured hard complete graphs and misleading complete graphs. Exact approaches and existing heuristics have no guaranteed bounds on the quality of the solutions and very time-consuming and it become ineffective for graphs for large number of vertices. However, in this paper using ACO metaheuristics namely pheromone update and visibility measure we demontrated effectiveness of the our proposed algorithm in solving this two types of our non-Euclidian graph problems.

2 Proposed Solution

This section first describes how communications networks is represented as an undirected complete graph, and defines relevant graph theory concept. Then we details our proposed ACO approach based on d-Prim[2] greedy heuristic.

2.1 Problem Representation

The degree-constrained minimum spanning tree (d-MST) problem can be stated as follows: Let graph $G = (V, E)$ be a connected weighted undirected graph, where $V = \{v_1, v_2, \dots, v_n\}$ is a finite set of vertices, and $E = \{e_1, e_2, \dots, e_m\}$ is a finite set of edges representing connections between these vertices. Let $W = \{w_1, w_2, \dots, w_m\}$ represent the weight or cost of each edge where the weight or cost is restricted to be a nonnegative real number. Any subgraph of G can be described using a vector $x = (x_1, x_2, \dots, x_m)$ where let each element of x_i be a binary decision variable is defined as 1, if edge e_i is part of the subgraph and 0, otherwise.

Let S be a subgraph of G . S is said to be a spanning tree in G if S :

- a) contains all the vertices of G and the vertices can be in non-order form;
- b) is connected, and simple graph contains no cycles.

Now let T be the set of all spanning trees corresponding to the simple graph G . In the MST problem, if we assume that there is a degree constraint on each vertex such that at each vertex v_j , the degree value d_j of vertex j is at most a given constant value d , the number of edges incident to each vertex is constrained. Then the problem is denoted as a d-MST and can be formulated as follows:

$$\min \left\{ z(x) = \sum_{i=1}^{|E|} w_i x_i \mid d_j \leq d, j \in V, x \in T \right\} \tag{1}$$

2.2 The ACO Algorithm for d-MST Problem

The set of edges, $E=\{e_1,e_2,\dots,e_m\}$ where m is the total number of edges or until total number of edges $|V|-1$, will serve as components from which each ant will use to incrementally construct a degree spanning tree(d-ST) during each iteration of the algorithm. We use a d-Prim greedy heuristic for d-MST problem in finding valid d-ST for our graph data set problem. To accommodate our ACO algorithm we need to perturb d-Prim algorithm through the use of an input solution vector so that it does not always choose the minimum weight edge at each step in the spanning tree construction which would not violate the degree constraint on the vertex chosen. The objective function f returns the minimum cost degree-constrained spanning tree path S_k found by ant k .

Let d_{ij} be the distance between vertices i and j and τ_{ij} the amount of pheromone on the edge that connects i and j . τ_{ij} is initially set to a small value as $\tau_0= 10^{-6}$.

The algorithm consists of a series of iterations.

- (1) A set of m artificial ants are initially located at randomly selected vertices as $m = |V|$ in the graph;
- (2) Each ant, denoted by k , constructs a complete valid d-ST, selecting each of vertices exactly once or until $|V|-1$ edges, always maintaining a set list J_k of vertices that remain to be visited;
- (3) At step r of iteration t , an ant located at a set of vertices i hops to a set of another vertices j , selected among the vertices that have not yet been visited to be included in its partially constructed solution till $S_k(t_r)=\{e_1,e_2,\dots e_{|V|-1}\}$, according to probability:

$$p_{ij}^k(t_r) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in J_k} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta}, \forall j \in J_k(i), \eta_{ij} = \frac{1}{d_{ij} * d_{ij}} \\ 0, \text{ otherwise} \end{cases} \tag{2}$$

where α and β are two positive parameters which govern the respective influences of pheromone and distance visibility on ants' decision and η as inverse square distance.

- (4) When every ant has completed a d-ST, pheromone trails are updated:

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \sum_{k=1}^m \Delta \tau_{ij}^k \tag{3}$$

where

$$\Delta \tau_{ij}^k = \rho \cdot \tau_0 \text{ as local update}$$

$$\Delta \tau_{ij}^k = \begin{cases} (Q/L_{gb}), \text{ if } (i,j) \in \text{ edges of length global-best minimum d-ST} \\ 0 \end{cases} \text{ as global update}$$

The ρ is the evaporation rate. Pheromone evaporates at a fixed rate after all ants have constructed their d-ST. $\Delta \tau_{ij}$ is the amount of reinforcement received by edge (i, j) . $\Delta \tau_{ij}$ is proportional to the quality of the solutions in which (i,j) was used by one ant or more. While ant building a solution, d-ST of d-MST, local update rule was applied. The local update rule was needed to yield better performance by encouraging ants

exploration for $\Delta \tau_{ij} = \rho \cdot \tau_0$ where $\tau_0 = 10^{-6}$. Only the globally best minimum d-ST_k(L_{gb}) constructed minimum cost path by ant k is allowed to deposit pheromone. Our parameter values for artificial ant used are $m = |V|$, $Q = 1.0$, $\alpha = 1$, $\beta = 5$, $\rho = 0.1$ and $\tau_0 = 10^{-6}$.

3 Experiments and Results

The proposed approach is tested firstly on Savelsbergh and Volgenant[7] data set and then on structured hard graph and misleading graph data sets for complete, undirected and weighted graph. The ACO algorithm was implemented in RePast Java multi-agent visualisation and simulation framework.

3.1 Savelsbergh and Volgenant Data Set

The numerical example was given by Savelsbergh and Volgenant who solved it using heuristic algorithm denoted as edge exchange[7] and the optimal solution is 2256 for 9-vertex $d=3$. The algorithm was run 30 times for this data set. The iteration or cycle number was 10 for each run. By our ACO approach, the optimal solution 2256 can be reached almost all the time and with as high as 90% of the time. By increasing our ACO iteration from 10 to 20 times for each run, our experiment will always produce the optimal solution 2256.

3.2 Structured Hard (SHRD) Graph and Misleading (M-graph) Graph Data Sets

The SHRD graph are constructed by using non-euclidean distances that hardly to be solved optimally compare to other data sets such as Euclidean data sets of a degree of 3 or more[4]. The misleading graph data set consists if randomly generated weights with a positive sign, greater than or equal to 0.0 and less than 1.0 attached to the edges in such a way that they will mislead greedy algorithms [2,6].

Table 1 show results for the SHRD instance set. The number of vertices ranges from 15, 20, 25 and 30 where the maximum degree was set to 3, 4, and 5. The solution quality is measured by the relative difference between the final objective value C obtained by a specific approach and the objective value C_{d-Prim} of the solution found by the d-Prim heuristic where the first vertex is used as starting point in percent:

$$\text{quality gain} = (C_{d-Prim} - C) / C_{d-Prim} \cdot 100\% \tag{4}$$

In other word, we use d-Prim as a reference algorithm and calculate relative quality improvements for the other approaches; where larger values indicate better results. The results for the MST based on Prim and Kruskal algorithm, two Prüfer-coded EA variants F-EA and P-EA (differ in handling infeasible solutions, crossover method, breeding method and termination criteria), problem search space PSS, simulated annealing SA, branch and bound B&B, weight-coded evolutionary algorithm W-EA and edge-set representation evolutionary algorithm S-EA are adopted from [6] and printed for comparison purposes only.

For ACO approach, 50 independent runs were performed for each problem instance and maximum degree, $d \in \{3,4,5\}$. Each run is terminated after 100 iterations. Besides average gains, the gains of the best runs are reported in Table 1. We can conclude that there are three kinds of observation for SHRD data set problem. ACO did not manage to attain any gain for 3 problem instances, they are SHRD155 $d=5$, SHRD205 $d=5$ and SHRD304 $d=4$ over d-Prim. For all other problem instances, there are positive gains over d-Prim. There are instances ACO gave gains that are better than at least one of other approaches such as in SHRD154 $d=4$, SHRD155 $d=5$, SHRD204 $d=4$, SHRD253 $d=3$, SHRD303 $d=3$ and SHRD305 $d=5$. The best performance can be observed for SHRD253 $d=3$ instance, where the improvement is better than five other approaches, namely F-EA, P-EA, PSS, SA and B&B.

Table 1. Average results (quality gains over d-Prim[2] in %) on hard problem instances Label SHRD153 means problem instance set of SHRD 15-vertex with degree constraint, $d=3$ and so on

Problem	MST avg.	F-EA avg.	P-EA avg.	PSS avg.	SA avg.	B&B avg.	W-EA avg.	S-EA avg.	ACO avg.	ACO best
SHRD153	19.67	13.66	15.07	16.62	14.93	18.03	14.20	18.03	10.16	13.11
SHRD154	5.04	10.83	0.39	12.99	11.61	14.76	11.42	15.35	4.65	4.65
SHRD155	0.00	4.00	-1.07	9.60	9.07	9.60	3.53	9.60	0.00	0.00
SHRD203	6.79	11.32	5.38	10.91	10.43	10.91	12.29	12.43	0.38	1.07
SHRD204	4.74	6.82	0.80	7.05	5.57	7.05	8.50	8.78	3.83	4.38
SHRD205	0.00	6.28	1.46	7.30	7.74	7.30	7.96	8.44	0.00	0.00
SHRD253	22.22	13.07	13.41	15.40	14.73	15.40	16.51	16.75	15.87	15.87
SHRD254	6.00	4.84	1.59	6.79	5.56	6.79	6.83	7.69	0.70	0.72
SHRD255	4.85	5.37	5.92	6.74	5.19	8.29	9.01	9.01	3.16	3.16
SHRD303	12.42	6.51	6.51	11.27	9.53	11.27	12.50	12.17	6.55	6.55
SHRD304	2.27	7.30	3.79	10.58	8.45	10.58	11.76	10.80	0.00	0.00
SHRD305	1.77	2.18	0.19	4.74	2.50	4.74	5.77	4.79	1.27	1.27

Table 2 shows results for misleading problem instances from Günther R. Raidl [6]. Besides average gains, the gains of the best runs are reported in Table 2. The results for the MST based on Prim and Kruskal algorithm, dual-simplex heuristic DS, Knowles and Corne’s evolutionary algorithm K-EA, weight-coded evolutionary algorithm W-EA and edge-set representation evolutionary algorithm S-EA are adopted from [6] and printed for comparison purposes only.

Table 2. Average results (quality gains over d-Prim[2] in %) on misleading problems with degree, $d=5$

Problem	V	d	MST avg	DS avg.	K-EA avg.	W-EA avg.	S-EA avg.	ACO avg.	ACO best
M1d5	50	5	77.01	24.14	27.59	42.76	43.59	21.42	26.91
M2d5	50	5	74.45	14.23	33.22	48.63	50.59	27.42	35.71
M3d5	50	5	66.57	-11.5	26.98	29.25	33.33	16.35	22.08

By our observation, we can conclude that ACO managed to attain positive gain for all the M-graph problem instances over d-Prim. ACO manage to attain higher positive gain over DS except in M1d5 problem instance. Overall our ACO approaches attain positive gain as in K-EA, W-EA and S-EA approaches but does not attains higher positive gain compare to them.

4 Conclusion

We have demonstrated how ACO algorithm can be used to solve d-MST. Our preliminary study shows that there are encouraging results indicating that our ACO is as good as several other metaheuristic approaches. In future, we will consider several ways to improve our ACO. Possible approaches include hybrid models, incorporation of local search or using better visibility measure.

References

1. M.Dorigo and G. Di Caro: The Ant Colony Optimization Metaheuristic. In D. Corne, M. Dorigo and F. Glover (eds.): *New Ideas in Optimization*. McGraw-Hill (1999)
2. Knowles, J. and Corne, D.: A new evolutionary approach to the degree-constrained minimum spanning tree problem. *IEEE Trans. on Evolutionary Computation*, Vol. 4, Issue 2, (2000) 125-134
3. Li-Jen Mao, Narsingh Deo, and Sheau-Dong Lang: A parallel algorithm for the degree-constrained minimum spanning tree problem using nearest-neighbor chains. *4th I-SPAN on Parallel Architectures, Algorithms, and Networks*, (1999) 184-189
4. Mohan Krishnamoorthy , Andreas T. Ernst , Yazid M. Sharaiha: Comparison of Algorithms for the Degree Constrained Minimum Spanning Tree. *Journal of Heuristics*, Vol. 7, No. 6, (2001), 587-611.
5. N. Deo and S.L. Hakimi: The shortest generalized Hamiltonian tree. *Proc. 6th Annual Allerton Conference*, (1968), 879-888
6. Raidl, G. R.: An efficient evolutionary algorithm for the degree-constrained minimum spanning tree problem. *IEEE Trans. on Evolutionary Computation*, Vol 1, (2000) 104-111
7. Savelsbergh, M. and T. Volgenant: Edge exchanges in the degree-constrained spanning tree problem. *Computer and Operation Research*, Vol. 12, No. 4, (1985) 341-348
8. Zhou, G., Gen, M. and Tianzu Wu: A new approach to the degree-constrained minimum spanning tree problem using genetic algorithm. *International Conf., System, Man, and Cybernetics*, Vol. 4 (1996) 2683-2688

Crowd Avoidance Strategy in Particle Swarm Algorithm

Guimin Chen, Qi Han, Jianyuan Jia, and Wenchao Song

School of Electronical and Mechanical Engineering,
Xidian University, Xi'an 710071, China
efoxxx@126.com, hanqi_aa@sina.com, jyjia@xidian.edu.cn

Abstract. To improve the linearly varying inertia weigh particle swarm optimization method (LPSO), a new concept of Crowd Avoidance is introduced in this paper. In this newly developed LPSO (CA-LPSO), particles can avoid entering into a crowded space while collaborate with other particles searching for optimum. Four well-known benchmarks were used to evaluate the performance of CA-LPSO in comparison with LPSO. The simulation results show that, although CA-LPSO falls behind LPSO when optimizing simple unimodal problems, it is more effective than LPSO for most complex functions. The crowd avoidance strategy enables the particles to explore more areas in the search space and thus decreases the chance of premature convergence.

1 Introduction

Particle swarm optimization (PSO) is an evolutionary computation technique first introduced by Kennedy and Eberhart [1] in 1995. For its simplicity of implementation and ability to quickly converge to a reasonably good solution, PSO is becoming very popular with researchers of many fields [2] [3] [4] [5] [6].

The motivation for the development of particle swarm algorithm was based on the simulation of simplified animal social behaviors such as fish schooling, bird flocking [1] [7], etc. It starts with the random initialization of a population (swarm) of individuals (particles) in the search space, which is similar to many other evolutionary algorithms, such as genetic algorithm and ant colony algorithm. In the search space, the trajectory of each particle is adjusted by dynamically altering its velocity, according to its own flying experience and the flying experience of the entire swarm. It is assumed that there are n particles in an m -dimensional search space. The position and the velocity of the j th particle at time t is represented as $X_j(t) = [x_{j,1}(t), x_{j,2}(t), \dots, x_{j,m}(t)]$ and $V_j(t) = [v_{j,1}(t), v_{j,2}(t), \dots, v_{j,m}(t)]$ respectively. According to a user defined fitness function, the j th particle remembers its best position represented as $X_j^*(t) = [x_{j,1}^*(t), x_{j,2}^*(t), \dots, x_{j,m}^*(t)]$ (which corresponds to the best fitness value obtained by that particle so far), and the swarm remembers the fittest position represented as $X^{**}(t) = [x_1^{**}(t), x_2^{**}(t), \dots, x_m^{**}(t)]$ found by all the particles so far. Then, the new velocities and the positions of the particles for the next generation are calculated using the following two equations:

$$\begin{aligned}
 V_j(t+1) = & V_j(t) + c_1 \times \text{Rand}_1 \times [X_j^*(t) - X_j(t)] \\
 & + c_2 \times \text{Rand}_2 \times [X^{**}(t) - X_j(t)]
 \end{aligned}
 \tag{1}$$

$$X_j(t+1) = X_j(t) + V_j(t)
 \tag{2}$$

where c_1 and c_2 are constants known as acceleration coefficients, c_3 is constants known as velocity coefficient, Rand_1 and Rand_2 are two separately generated uniformly distributed random numbers in the range [0, 1]. The first part of (1) represents the inertia, the second part represents the personal thinking of each particle, and the third part represents the collaborative effect of the particles. The social component always pulls the particles toward the global best particle found so far [7].

There has been a considerable amount of work done in improving its performance based on this canonical PSO. One of the most significant developments is the introduction of Inertia Weight by Shi and Eberhart [9] in 1998. It is known that proper control of global exploration and local exploitation is crucial for population-based optimization methods to find the optimum solution efficiently [7] [9] [10]. The Inertia Weight is the very parameter employed to balance the local and global search by controlling the impact of the previous velocities on current velocity during the optimization process. A larger inertia weight facilitates global optimization, while smaller inertia weight facilitates local optimization. Accordingly, Shi and Eberhart found a significant improvement in the performance of the PSO with a linearly varying inertia weight (LPSO) over the generations in 1999. The mathematical representation of LPSO is given by (3) and (4).

$$\begin{aligned}
 V_j(t+1) = & w(t) \times V_j(t) + c_1 \times \text{Rand}_1 \times [X_j^*(t) - X_j(t)] \\
 & + c_2 \times \text{Rand}_2 \times [X^{**}(t) - X_j(t)]
 \end{aligned}
 \tag{3}$$

$$X_j(t+1) = X_j(t) + V_j(t)
 \tag{4}$$

where $w(t)$ is given by

$$w(t) = (w_{\max} - w_{\min}) \times \frac{\text{MAXITER} - t}{\text{MAXITER}} + w_{\min}
 \tag{5}$$

where w_{\max} and w_{\min} are the initial and final values of the inertia weight, respectively, and MAXITER is the maximum number of allowable iterations. Through empirical studies, Shi and Eberhart have observed that the optimal solution can be improved by linearly varying the value of inertia weight from 0.9 at the beginning of the search to 0.4 at the end for most problems.

Although LPSO may outperform other evolutionary algorithms in the early iterations, its performance may not be competitive as the number of generations is increased [7]. In particular, when strongly multi-modal problems are being optimized, PSO usually suffers from premature convergence. The rationale behind this is that the

social behavior drives all the particles into a small space found by the best particle. This small space is not guaranteed to be even a space including a local optimum only.

In order to improve the search ability of LPSO by providing additional diversity, a novel notion of **Crowd Avoidance** is introduced in this paper.

2 Particle Swarm Optimization with Crowd Avoidance Strategy

It is well-known that the fishes usually stay at the place with plentiful food. However, they may escape from the crowded place even there is abound with food. To find a “comfortable” position, a fish has to adjust “his” velocity according to “his” own experience and the experience of one of his companions selected from the swarm according to his criterion of population density. This behavior would lead the particles to congregate at more than one position. Therefore, we introduce a novel notion of “Crowd Avoidance” into the LPSO, in order to enhance the global search capability of the particles by providing additional diversity.

In a m -dimensional linear searching space defined by $P_{\min} = [x_{\min,1}, x_{\min,2}, \dots, x_{\min,m}]$ and $P_{\max} = [x_{\max,1}, x_{\max,2}, \dots, x_{\max,m}]$, the distance between two positions P_k and P_l can be expressed as:

$$d = \|P_l - P_k\|, \tag{6}$$

where $\|\cdot\|$ means norm in m -dimension space. The dominated space of position X_k covers all the positions to which the distances are less than R :

$$R = \frac{1}{c_3} \|P_{\max} - P_{\min}\|, \tag{7}$$

where R is the radius of dominated space, and c_3 is a reserved adjustable parameter for R , which equals 30.0 throughout this paper. The population density at position P_k (PD_k) is defined as the number of particles in the dominated space of P_k . When PD_k reaches the maximum population density (PD_{\max} , which equals $n/2.0$ in this paper), it means that the dominated space of P_k is too crowded, so the particles outside that space will try to keep away from it. Otherwise, we call P_k a **comfortable** position.

For the j th particle at time t ($X_j(t)$), he will observe $X^{**}(t)$ found by the best particle first. If the dominated space of $X^{**}(t)$ is too crowded, he will turn to observe the best position found by the second best particle, then the third best particle, and so on. By representing the best comfortable position as $X_j^{**}(t) = [x_{j,1}^{**}(t), x_{j,2}^{**}(t), \dots, x_{j,m}^{**}(t)]$, the mathematical representation of CA-LPSO is given by (8) and (9).

$$V_j(t+1) = w(t) \times V_j(t) + c_1 \times \text{Rand}_1 \times [X_j^*(t) - X_j(t)] + c_2 \times \text{Rand}_2 \times [X_j^{**}(t) - X_j(t)], \tag{8}$$

$$X_j(t+1) = X_j(t) + V_j(t). \tag{9}$$

3 Experiments

Four of the well-known benchmarks, given in Table 1, were used to evaluate the performance of the newly developed CA-LPSO. The Sphere function is a simple unimodal function, the Rosenbrock function is a deceptively flat function, whereas the other two functions are multimodal functions designed with a considerable amount of local minima. In addition, all the functions have the global optimum value of zero at the origin or very close to the origin [2].

In our implementations for both CA-LPSO and LPSO, the stopping criteria are set to 1×10^{-6} , 40, 0.05 and 2 for these four benchmarks successively, the maximum inertia weight $w_{\max} = 0.9$, and the minimum $w_{\min} = 0.4$. Other parameters are selected as: number of particles $n = 50$, accelerate coefficients $c_1 = c_2 = 2.0$ and the maximum number of allowable iterations MAXITER = 3000.

Table 1. Four benchmarks for experiments

Function name	Mathematical representation	Dimension(d)	Range of search	V_{\max}
Sphere	$f_1 = \sum_{i=1}^n x_i^2$	30	$(-100, 100)^d$	100
Rosenbrock	$f_2 = \sum_{i=1}^n [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	10	$(-100, 100)^d$	100
Griewank	$f_3 = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	10	$(-600, 600)^d$	600
Rastrigrin	$f_4 = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	10	$(-10, 10)^d$	10

Table 2. Mean value, standard deviation and times reaching stopping criteria of the optimal value for 50 trials

	Method	Mean	Standard deviation	Trials reaching stopping criteria
Sphere	LPSO	9.2392×10^{-7}	8.1193×10^{-8}	50
	CA-LPSO	9.4222×10^{-7}	6.5220×10^{-8}	50
Rosenbrock	LPSO	66.5374	130.2780	19
	CA-LPSO	28.1446	71.0008	48
Griewank	LPSO	0.0801	0.0334	17
	CA-LPSO	0.0660	0.0298	36
Rastrigrin	LPSO	2.9063	2.3420	29
	CA-LPSO	2.1294	1.2068	42

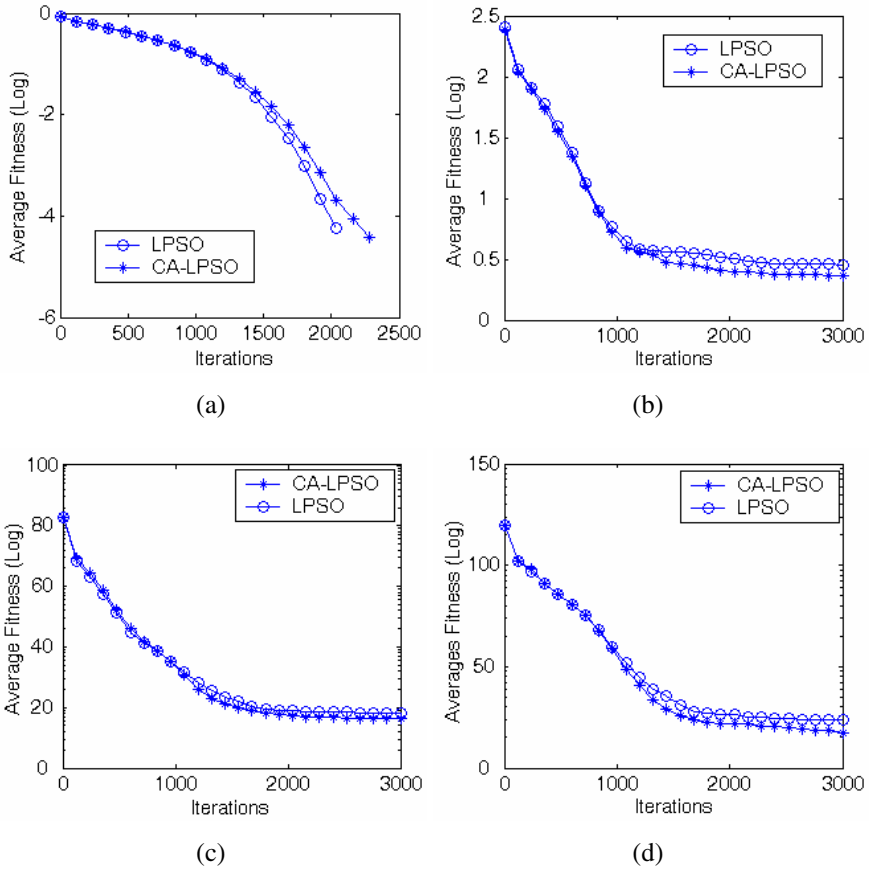


Fig. 1. Variation of the mean best value of the benchmark functions versus the number of iterations. (a) Sphere function; (b) Rosenbrock function; (c) Griewank function; (d) Rastrigrin function.

Fifty trials were carried out for each benchmark to evaluate the performance in terms of the rate of convergence and the quality of the optimum solution of the newly developed CA-LPSO in comparison with the LPSO method. The average optimal values, the standard deviations and the times reaching stopping criteria of the optimal value are summarized in **Table 2**. **Fig. 1** displays the variation of the average best solution over the iterations of the four benchmarks respectively.

From **Fig. 1**, it is clear that the convergence rates of CA-PSO and LPSO are almost the same at the early stage of the optimization process for all the four benchmarks. As to the Sphere function, CA-PSO falls behind LPSO at the latter stage; it takes LPSO only 2100 iterations to reach the stopping criteria of 1×10^{-6} , while CA-PSO 2300 iterations. For the other three functions, CA-PSO always outperforms LPSO at the latter stage, especially after 1000 iterations. From the simulation results listed in **Table 2**, it can be concluded that the CA-LPSO method can find better solutions for most of the complex functions than LPSO.

4 Conclusion

Although LPSO may outperform other evolutionary algorithms in the early iterations, it usually suffers from premature convergence as the number of generations is increased, especially for strongly multi-modal problems. In this paper, we introduced a novel notion of Crowd Avoidance to LPSO (CA-LPSO), in order to improve the search ability of LPSO by providing additional diversity.

The simulation results show that, although CA-LPSO falls behind LPSO when optimizing simple unimodal problems, it is more effective than LPSO for most complex functions. The crowd avoidance strategy enables the particles to explore more areas in the search space and thus decreases the chance of premature convergence.

References

1. Kennedy, J. and Eberhart, R.: Particle Swarm Optimization. Proceeding of IEEE International Conference on Neural Networks, (1995) 1942–1948
2. Ayed, S., Imtiaz, A. and Sabah, A. M.: Particle Swarm optimization for Task Assignment Problem. *Microprocessors and Microsystems*, Vol. 26 (2002) 363–371
3. Elegbede, C.: Structural Reliability Assessment Based on Particles Swarm Optimization. *Structural Safety*. Vol.27 (2005) 171–186
4. Boeringer, D. W. and Werner, D. H.: Particle Swarm Optimization versus Genetic Algorithms for Phased Array Synthesis. *IEEE Transactions on Antennas and Propagation*, Vol. 52 (2004) 771–779
5. Perez, J. R. and Basterrechea, J.: Particle-Swarm Optimization and Its Application to Antenna Far-Field-Pattern Prediction from Planar Scanning. *Microwave and Optical Technology Letters*, Vol. 44 (2005) 398–403
6. Abido, M. A.: Optimal Power Flow Using Particle Swarm Optimization. *Electrical Power and Energy System*. Vol.24 (2002) 563–571
7. Ratnaweera, A., Halgamuge, S. K. and Watson, H. C.: Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients. *IEEE Transactions on Evolutionary Computation*, Vol. 8 (2004) 240–255
8. Angeline, P. J.: Evolutionary Optimization Verses Particle Swarm Optimization: Philosophy and the Performance difference. *Lecture Notes in Computer Science*. Vol.1447 (1998) 600–610
9. Shi, Y. and Eberhart, R. C.: A Modified Particle Swarm Optimization. Proceeding of IEEE International Congress on Evolutionary Computation, (1998) 69–73
10. Shi, Y. and Eberhart, R. C.: Empirical Study of Particle Swarm Optimization. Proceeding of IEEE International Conference on Evolutionary Computation, (1999) 101–106

Particle Swarm Optimization with Multiscale Searching Method

Xiaohui Yuan¹, Jing Peng², and Yasumasa Nishiura¹

¹ Research Institute for Electronic Science,

² Graduate School of Information and Science,

Hokkaido University, Kita-ku, Sapporo 060-0812, Japan

yuan@nsc.es.hokudai.ac.jp

Abstract. This paper presents a new method for effectively searching all global minima of a multimodal function. The method is based on particle swarm optimizer, particles are dynamically divided into several subgroups of different size in order to explore variable space using various step size simultaneously. In each subgroup, a new scheme is proposed to update the the positions of particles, this scheme takes into consideration the effect of all subgroup seeds. Experimental results for one dimensional, two dimensional and thirty dimensional test suites demonstrated that this method can get overall promising performance over a wide range problems.

1 Introduction

Finding all global minima of functions is a tough problem in diverse scientific fields, such as computation of Nash equilibria in game theory and computation of periodic orbits of nonlinear mappings [1]. During engineering design, we may often want to find all the solutions which satisfy our requirements, not only the best solution, but also the second best solutions are desired. However, this problem is proved to be NP-hard and can hardly be solved by traditional methods.

In the past decades years, evolutionary algorithms such as genetic algorithm (GA) and particle swarm optimization(PSO) gained increasing popularity and have been successfully applied to searching global minima of multimodal functions. Gordon[2] proposed an island model parallel GAs (IMGAs) which use parallel subpopulation method to increase the possibility of finding all optima. In this model, subpopulation will exchange their "good solutions" at the end of each iteration, however, this exchanging should be carefully controlled to ensure a good algorithm performance. In [3], Li described a species conservation conserving genetic algorithm(SCGA), comparing with the IMGAs, this method need fewer parameters, only one new control parameter called "species distance" is introduced in addition to those needed to control basic GA. Li[4] applied the species idea for PSO and got a faster convergence speed, but he only tested his method on functions whose global optima are evenly distributed. Konstantinos [1] incorporated PSO, deflection and stretching techniques, experimental results demonstrated that the hybrid PSO algorithm can be used to solve various problems.

In this paper, we introduce a multiscale PSO (MPSO) algorithm. This algorithm is a modified version of Li's SPSO [4] algorithm. Changing the value

of "species distance" adaptively during exploring, we can use small step size in complex area while bigger step size in flat area. Experimental results show that this method can achieve a rapid convergence speed and good performance.

The next section describes the basic idea of particle swarm optimization. Section 3 describes multiscale searching technique and the algorithm to implements it. Section 4 presents the experiment results and compare our result with well-known results. Section 5 concludes our discussion.

2 Simple Particle Swarm Algorithm

The particle swarm optimization(PSO) algorithm was firstly proposed by Eberhart and Kennedy [5]. In the past several years, PSO has grown fast and it has been investigated from various perspectives[6]. Like other stochastic optimization algorithms. PSO begins with a population of randomly distributed particles, then in the following iterations, particles update their positions based on predefined rules. To solve a d -dimensional multimodal function optimization problem, we can use N particles (X_1, X_2, \dots, X_N) , each particle is a d -dimensional vector and can be represented as $X_i = (x_i^1, x_i^2, \dots, x_i^d)$, the velocity of this particle is denoted as $V_i = (v_i^1, v_i^2, \dots, v_i^d)$. The best solution this particle found so far is denoted as $P_i = (p_i^1, p_i^2, \dots, p_i^d)$. We use $P_g = (p_g^1, p_g^2, \dots, p_g^d)$ to represent the best solution of all N particles. At $(K + 1)th$ iteration, each particle updates its position according to following equations:

$$\begin{aligned}
 V_i(K + 1) &= wV_i(K) + c_1r_1(P_i(K) - X_i(K)) \\
 &\quad + c_2r_2(P_g(K) - X_i(K)) \\
 X_i(K + 1) &= X_i(K) + V_i(K + 1)
 \end{aligned}
 \tag{1}$$

where w is called inertia weight, c_1, c_2 are learning factors and r_1, r_2 are random numbers uniformly distributed in $[0, 1]$.

3 Multiscale Searching for Particle Swarm Optimizer

Li[4] introduced a species-based PSO (SPSO) which is very effective in dealing with multimodal optimization functions with lower dimensions. In SPSO algorithm, particles with better fitness are selected as seeds. Each seed can choose

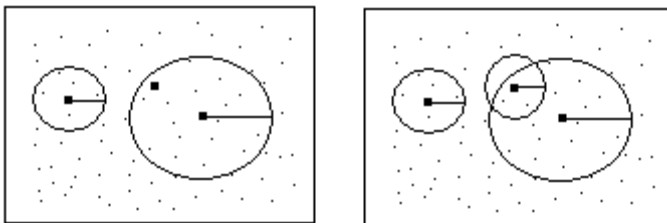


Fig. 1. Create Subgroups in Population

its members based on their similarity to create a subgroup. Euclidean distance r between two particles $X_i = (x_i^1, x_i^2 \dots x_i^d)$ and $X_j = (x_j^1, x_j^2 \dots x_j^d)$ is used to define the similarity, smaller distance means more similarity. Distance r is defined by:

$$r = d(X_i, X_j) = \sqrt{\sum_{k=1}^d (x_i^k - x_j^k)^2} \tag{2}$$

Unfortunately, this method can hardly find all global optima if the optima are unevenly distributed. Since the similarity threshold r_s is fixed during the searching and same to all subgroups, so if there are two optima the distance of which between them less than r_s , obviously, only one can be found. As shown in left picture of Figure 1, black squares are good solutions and black spots are normal solutions, inside the right circle, there are two good solutions, one is the seed located in the center of circle, but the other is among normal solutions, which will not be selected as a seed. Of course, if r_s is small enough, the two good solution will be selected as seeds synchronously. But, as is described in [3], if r_s is too small, many subgroups will be found in every generation, some subgroups may not have any members, on the other hand, if r_s is too big, there are only one group and it is difficult to find all global optima.

If we divide each subgroup into more smaller groups, then the possibility to find the other optima will be greatly increased, and this division has nothing to do with the threshold r_s , as shown in the right picture of Figure 1. The objective of division inside subgroups is to find other possible global optima in the subgroup besides the seed. Firstly, we sort the subgroup in descending order, the seed is the best solution in the subgroup, so it should be in the first place. The rest members whose fitness closed to the seed's are considered has potential to be a global optima. If we assume that there are only one global optima in a narrow space, there should be a minimum distance *MINDIS* between new seeds and the seed of subgroup.

3.1 Adaptive Control of Parameters

In the previous section, we described the method to create multiscale subgroups, and introduced some new parameters: r_s and *MINDIS*. Now, we discuss how to control these parameters. As was mentioned before, r_s should be carefully predefined. In fact, it is not easy to choose a proper value of r_s , however, it is relatively easier to know how many global minima a function has, so we introduce two parameters: max number and min number of subgroups in the population, respectively *MAXSPEC* and *MINSPEC*. Initially we set the r_s by:

$$r_s = \frac{\frac{1}{2} \sqrt{\sum_{k=1}^d (x_k^u - x_k^l)^2}}{\sqrt[d]{MAXSPEC}} \tag{3}$$

where x_k^u and x_k^l are respectively the upper and lower bounds of the d dimensional searching space. Before running the algorithm, we need to set the value

of *MAXSPEC* and *MINSPEC*. The values are decided by the properties of functions, they are generally proportion to the number of global optima of functions. For example, a smooth function have three global optima, so we set *MINSPEC* = 5 and *MAXSPEC* = 10; r_s is changed adaptively during the searching according to the number of seeds *SPECNUM* in the population, note that, the seeds which generated inside the group are not counted into the *SPECNUM*.

In our implementation, $MINDIS = r_s/\alpha$, α is normally set to 10, of course, *MINDIS* should less than the minimum distance between two global optima.

3.2 Update Particle's Position

Basic particle swarm optimization algorithm only consider the affection of best solution in the population and best solution the particle itself found so far. In our algorithm, there are many subgroups, so it is naturally to consider the effect of all the seeds. The longer distance between the particle and seed, the smaller effect between them.

$$\begin{aligned}
 V_i(K+1) &= wV_i(K) + (P_i(K) - X_i(K)) \\
 &\quad + (P_g(K) - X_i(K)) \\
 &\quad + \sum_{k=1}^{allseeds} \left(\frac{MIN(dis(X_i, Seed_k))}{dis(X_i, Seed_k)} \right)^3 \\
 X_i(K+1) &= X_i(K) + V_i(K+1)
 \end{aligned} \tag{4}$$

In this equation, P_g is the seed of the subgroup particle i belongs to, other parameters are defined as in previous section. Experimental results demonstrated that this formula can make the algorithm converge faster than the classical one.

4 Experiments

We test our algorithm for one dimensional, two dimensional and thirty dimensional functions, experimental results are listed in the following sections. For all the experiments, we got 100% success rate of finding all global optima.

4.1 One Dimension

The test functions F_1 to F_4 are binary-coded functions suggested by Beasley [7]. F_5 and F_6 are central two-peak trap and five-uneven-peak trap function respectively which described in [3]. Experiment parameters are set as below: *POPSIZE* = 50, $\alpha = 0.1$, $\beta = 10$, *MAXSPEC* = 10, *MINSPEC* = 5. We test each function 100 times, mean function evaluation numbers and standard deviation are calculated. Results are listed in Table 1, from the results, we can see that, compare with SPSO, our algorithm has a faster convergence speed. When test on function F_5 and F_6 , we changed the *MAXSPEC* = 5, *MINSPEC* = 3, other parameters are the same as before.

Table 1. Comparison of results for one dimensional functions

Func	Num.	Num. of evals (mean and std dev)	
		MPSO	SPSO
F_1	5	625± 195.12	1383± 242.95
F_2	1	399± 203.71	351± 202.35
F_3	5	653± 187.65	1248± 318.80
F_4	1	421± 221.493	503.33± 280.107
F_5	2	840± 328.23	935 (SCGA)
F_6	2	1380± 451.22	2811 (SCGA)

Table 2. Comparison of results for two dimensional functions

Func.	Num.	Num. of evals (mean and std dev)	
		MPSO	SCGA
F_7	4	3792± 487.24	3155 (SPSO)
F_8	2	2450± 452.24	1836
F_9	3	4849± 1149.37	8529

4.2 Two Dimension

Test function F_7 is two dimensional Himmelbalu’s function, F_8 is six-hump camel function, F_9 is Brannin RCOS function and F_{10} is two dimensional Shubert function which described in [3]. When test on function F_7 F_8 and F_9 , we set parameters: $POPSIZE = 100$, $\alpha = 10$, $\beta = 10$, $MAXSPEC = 20$, $MINSPEC = 15$. Results are listed in Table 2. We compare our result on F_7 with the results reported in [4] by SPSO, the convergence speed is slower than SPSO, however, the SPSO algorithm can not get 100% success rate on F_8 and F_9 .

Function F_{10} have 18 global optima and 736 local optima. We set parameters: $POPSIZE = 2000$, $\alpha = 150$, $\beta = 15$, $MAXSPEC = 25$, $MINSPEC = 10$. In [3] reported that it is cost SCGA almost 100000 function evaluation number to find all 18 global optima. In our experiment, we test the function 20 times, each time we found all 18 global optima, the mean evaluation number is 801518.5, standard deviation is 27689.2.

4.3 Higher Dimensions

We also test our algorithm on five common high dimensional test functions: Sphere, Dejong, Rosenbrock, Griewank and Rastrigin, all the functions have one minimum point. Experiment results was compared to results in [8] and [9]. In all experiments, the dimensionality $d = 30$.

In Table 3, the data of algorithm PSO and KSwarm are from paper [8], while data of algorithm Exp Ver and ES are from paper [9], MPSO is the method which described in this paper. From the experimental results we can see that our method got a overall good performance and it is much better than basic PSO.

Table 3. Experiment results on high dimensional functions

Function	Range	PSO	KSwarm	Exp Ver	ES	MPSO
Sphere	(-50,50)	370.041	4.723	0	0	0.29
DeJong	(-20,20)	4346.714	4.609	0	0	0.03
Rosenbrock	(-100,100)	2.61e7	3.28e3	47.75	50.8	29.15
Rastrigin	(-5.12,5.12)	106.550	53.293	63.22	57.19	13.34
Griewank	(-600,600)	13.865	0.996	0.038	0.002	0.31

5 Conclusion

In this paper, we propose a new method to find all global minima of multimodal functions. Several previous research indicated that PSO can locate the region of optimum faster than different EC algorithm, but once in this region PSO progressed slowly[1], our experimental results demonstrated that PSO can also achieve a rapid convergence speed in the region of optimum. Through partitioning searching space into several regions of different size, we increased the ability for multiscale searching without losing its rapid convergence.

References

1. Parsopoulos, K., Vrahatis, M.: On the computation of all global minimizers through particle swarm optimization. *IEEE Transactions on Evolutionary Computation* **8** (2004) 211–224
2. D., G.V.S.W., A., B.: Dataflow parallelism in genetic algorithm. In R., M., B., M., eds.: *Parallel Problem Solving from Nature 2*. Elsevier Science, Amsterdam, Netherlands (1992) 533–542
3. Li, J.P, B.M.P.G., Clarkson, P.: A species conversing genetic algorithm for multimodal function optimization. *Evolutionary Computation* **10** (2002) 207–234
4. Li, X.D.: Adaptively choosing neighbourhood bests using species in a particle swarm optimizer for multimodal function optimization. In: *Proceeding of Genetic and Evolutionary Computation Conference 2004 (GECCO'04)*. (2004)
5. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: *Proceeding of 6th symposium of micro machine and human science, Nagoya, Japan* (1995) 39–43
6. Hu, X., Shi, Y.: Recent advances in particle swarm. In: *Proceeding of IEEE Congress on Evolutionary Computation*. (2004) 90–97
7. Beasley, D., Bull, D., Martin, R.: A sequential niche technique for multimodal function optimization computation. *Evolutionary Computation* **1** (1993) 101–125
8. M., C.K., S., K.D.: The kalman swarm a nwe approach to particle motion in swarm optimization. In: *Proceeding of Genetic and Evolutionary Computation Conference 2004 (GECCO'04)*. (2004)
9. C., M., Kennedy, J.: The particle swarm - explosion stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation* **6** (2002) 58–73

Outcome-Space Branch and Bound Algorithm for Solving Linear Multiplicative Programming*

Yuelin Gao^{1,2}, Chengxian Xu¹, and Yueting Yang³

¹ School of Finance and Economics, Xi'an Jiaotong University, Xi'an 710049, China
mxxu@mail.xjtu.edu.cn

² Department of Information and Computation Science,
Northwest Second National College, Yinchuan 750021, China
gaoyuelin@263.net

³ Department of Mathematics, Beihua University, Jilin 132013, China
yyt@163.com

Abstract. This paper presents an outcome-space branch and bound algorithm for globally solving the linear multiplicative programming problems, in which we use a new two-level partition technique on rectangles and solve a simple relaxed quasiconcave programming problem at each branch node over outcome-space. We prove that the proposed algorithm is convergent. It can be shown by the numerical results that the proposed algorithm is effective.

1 Introduction

Consider the nonconvex programming problem below:

$$(\text{LMP}) \quad \begin{cases} \min \phi(x) = \prod_{j=1}^p (c_j^T x + d_j) \\ \text{s.t. } x \in D = \{x \in R^n | Ax \leq b\} \end{cases}$$

where $2 \leq p \in Z^+$, $c_j \in R^n$, $d_j \in R$ for $j = 1, 2, \dots, p$; $A \in R^{m \times n}$, $b \in R^m$; $c_j^T x + d_j \geq 0$ on D for $j = 1, 2, \dots, p$; the feasible field D is a bounded convex polyhedron. This problem is known as linear multiplicative programming problem. Let x^* be a global optimal solution of the problem (LMP).

The problem (LMP) come from many fields, for example, multiple-objective decision [2], bond portfolio selection [3], optimal packing and layout [6], VLISI chip design [11], etc.. The problem (LMP) is muliextremal and may possess several local minima [9]. It is an NP-hard Problem [6] too. In the last decades, many solution algorithm have been proposed for globally solving the problem (LMP), for

* The work is supported by the Foundation of National Natural Science in China (No. 10231060), and by the Science Research Project of National Committee in China in 2005, and by the Science Research Project of Ningxia's Colleges and Universities in 2005.

example, parameterization-based methods[7], outer-approximation and branch-and-bound methods[12], a primal and dual simplex method [8], outcome-space cutting plane method[9], heuristic methods [10], etc..

In the paper, we present an outcome-space branch and bound algorithm for globally solving the linear multiplicative problems. To accomplish this, we propose a new two-level partition technique on rectangles and give a simple relaxed quasiconcave programming problem of it in outcome-space to determine a lower bound of its global optimal value as well as gives a convergent property of the proposed algorithm. It can be shown by the numerical results that the proposed algorithm is effective. And the proposed algorithm is not the same as the one in the reference [12] in the partitioning of rectangles and relaxed bounding of the problem (LMP).

Section 2 gives a simple relaxed quasiconcave programming problem in outcome-space. Section 3 presents a new two-level partition technique on rectangles. Section 4 describes the outcome-space branch and bound algorithm and gives the convergent theorem of the proposed algorithm. Section 5 gives several numerical computational examples in order to show that the proposed algorithm is effective.

2 A Simple Relaxed Quasiconcave over Outcome-Space

In this section, we will give a simple relaxed quasiconcave problem over outcome space to determine a lower bound of the global optimal value of Problem (LMP).

Let $f_j(x) = c_j^T x + d_j$ for $j = 1, 2, \dots, p$, $f(x) = (f_1(x), f_2(x), \dots, f_p(x))^T$. So the outcome space of the problem (LMP) is the p -dimensional space. Solve $2p$ linear programs ($j = 1, 2, \dots, p$) below:

$$\min\{f_j(x) : x \in D\} \quad \text{and} \quad \max\{f_j(x) : x \in D\},$$

where their optimal solutions are noted as $\bar{x}^{j0}, \underline{x}^{j0}$ and their optional values are noted as l_j^0, u_j^0 for $j = 1, 2, \dots, p$, respectively. Obviously, $\bar{x}^{j0}, \underline{x}^{j0}$ for $j = 1, 2, \dots, p$ are all feasible points of the problem (LMP).

Let $l = (l_1^0, l_2^0, \dots, l_p^0)^T$, $u = (u_1^0, u_2^0, \dots, u_p^0)^T$, $Y = \{y \in R^p | l \leq y \leq u\}$. Now, we construct the nonconvex programming problem below:

$$\text{(CCLMP)} \quad \begin{cases} \min \bar{\phi}(y) = \prod_{j=1}^p y_j \\ \text{s.t.} \quad y \in Y. \end{cases}$$

The problem (CCLMP) is a quasiconcave programming problem in the outcome space [8]. The point l is unique global optimal solution of it and $\bar{\phi}(l)$ is its global optimal value. It can be seen that $f(x) \geq f(l)$ for any $x \in D$, i.e. $f(x) \in Y$ for any $x \in D$, so $\phi(x) \geq \bar{\phi}(l)$ for any point $x \in D$, thus the problem (CCLMP) is a relaxed quasiconcave program of the problem (LMP) over outcome space, and $\bar{\phi}(l)$ is a lower bound of the global optimal value $\phi(x^*)$ of the problem (LMP).

3 Two-Level Rectangle-Partition in Outcome-Space

In the section, we partition the feasible field of the problem (CCLMP) into $p + 2$ subrectangles at most. Let $\bar{y} = f(\bar{x}) \in [\bar{l}, \bar{u}] (\subseteq R^p)$ be a point in \bar{Y} . At first, we will give the procedure below to partition the rectangle \bar{Y} into $p + 1$ subrectangles at most.

The first partitioning procedure:

For $j = 1, 2, \dots, p$ **do**

If $\bar{y}_j \neq \bar{l}_j$ and $\bar{y}_j \neq \bar{u}_j$, **then** \bar{Y} is partitioned into two subrectangles along the point \bar{y}_j on the edge $[\bar{l}_j, \bar{u}_j]$, i.e.

$$\begin{aligned} Y^j &:= \{y \in R^p | l^j = (\bar{l}_1, \dots, \bar{l}_{j-1}, \bar{y}_j, \bar{l}_{j+1}, \dots, \bar{l}_p) \leq y \\ &\leq u^j = (\bar{u}_1, \bar{u}_2, \dots, \bar{u}_p)\}, \\ \bar{Y} &:= \{y \in R^p | \bar{l} = (\bar{l}_1, \bar{l}_2, \dots, \bar{l}_p) \leq y \leq \bar{u} \\ &= (\bar{u}_1, \dots, \bar{u}_{j-1}, \bar{y}_j, \bar{u}_{j+1}, \dots, \bar{u}_p)\}, \end{aligned}$$

else

\bar{Y} is not changed

endif

enddo

Let $Y^{p+1} := \bar{Y}$.

So, the procedure makes \bar{Y} be partitioned into $p + 1$ subrectangles at most. The set to consist of all the subrectangles is noted as Ω , and the index set to consist of the indexes of the subrectangles in Ω is noted as J . Obviously, if $|\Omega| = 1$ where $|\Omega|$ is element number in the set Ω , then $f(\bar{x})$ is a extreme point of the rectangle \bar{Y} for the procedure to start, and the rectangle \bar{Y} for the procedure to start is deleted in the set Ψ to consist of the some subrectangles of Y , which will be given by the algorithm in Section 4.

Let $|\Omega| > 1$. The second partitioning process will be done below.

The second partitioning procedure:

The rectangle which has the longest edge in Ω is partitioned into two subrectangles. To the rectangle Y^j for each $j \in J$, we compute

$$V_j = \max\{(u_i^j - l_i^j) | i = 1, 2, \dots, p\},$$

and the corresponding edge's index λ_j , and determine index η such that

$$V_\eta = \max\{V_j | j \in J\}.$$

So the rectangle Y^η is the one that has the longest edge in Ω , the index λ_η is the one in correspondence with the longest edge of the rectangle Y^η .

Let $\alpha_{\lambda_\eta}^\eta = \frac{u_{\lambda_\eta}^\eta + l_{\lambda_\eta}^\eta}{2}$. Thus the rectangle Y^η is partitioned along the midpoint of the edge $[l_{\lambda_\eta}^\eta, u_{\lambda_\eta}^\eta]$ into two volume-equal subrectangles Z^1, Z^2 , i.e.

$$\begin{aligned} Z^1 &= \{y \in R^p | l^{\eta,1} = (l_1^\eta, \dots, l_{\mu-1}^\eta, l_\mu^\eta, l_{\mu+1}^\eta, \dots, l_p^\eta) \\ &\leq y \leq u^{\eta,1} = (u_1^\eta, \dots, u_{\mu-1}^\eta, \alpha_{\lambda_\eta}^\eta, u_{\mu+1}^\eta, \dots, u_p^\eta)\}, \\ Z^2 &= \{y \in R^p | l^{\eta,2} = (l_1^\eta, \dots, l_{\mu-1}^\eta, \alpha_{\lambda_\eta}^\eta, l_{\mu+1}^\eta, \dots, l_p^\eta) \\ &\leq y \leq u^{\eta,2} = (u_1^\eta, \dots, u_{\mu-1}^\eta, u_\mu^\eta, u_{\mu+1}^\eta, \dots, u_p^\eta)\}, \end{aligned}$$

where $\mu = \lambda_\eta$.

If $f(\bar{x}) \in Z^1$, set $Y^n := Z^1, Y^0 := Z^2$; otherwise, set $Y^n := Z^2, Y^0 := Z^1$. So the rectangle \bar{Y} is partitioned into $p + 2$ subrectangles at most. It can be shown that the subrectangle Y^{p+1} can be deleted and the subrectangle Y^0 can be deleted under some conditions. Now, the set to consist of the remainder rectangles in Ω is yet noted as Ω , and the set to consist of the indexes of the rectangles in Ω is yet noted as J .

We see easily that the partition procedure is exhaust, i.e. the each infinite rectangle-embedded array to be produced in the procedure converges to a point. The first partitioning procedure produces $p + 1$ subrectangles at most; the second partitioning procedure makes the two-level partitioning procedure exhaust.

4 Branch and Bound Algorithm in Outcome-Space

In the iteration procedure of the next algorithm, the sets corresponding with the ones given in Section 3 have the subrectangle set Ψ_k , the subrectangle set Ω_k , the index set I_k to consist of the indexes of the subrectangles of Ψ_k , the index set J_k to consist of the indexes of the subrectangles of Ω_k , and the subrectangles in Ψ_k and Ω_k .

Branch and Bound Algorithm in Outcome-Space (BBAOS)

Step0. Let $\varepsilon > 0$. Find a rectangle $Y = \{y \in R^p | l \leq y \leq u\}$ by the method in

Section 2. Let $\Psi_0 = \{Y\}, \alpha_0 = \prod_{j=1}^p l_j, \beta_0 = \min\{f(\bar{x}^{j0}), f(\bar{x}^{j0}) : j = 1, 2, \dots, p\},$

$x^0 = arg\beta_0$. If $\beta_0 - \alpha_0 < \varepsilon$, let $x^* = x^0, \phi^* = \beta_0$, outcome x^*, ϕ^* , stop; otherwise, go to Step 1.

Step k ($k = 1, 2, 3, \dots$)

k0. When Step k starts, we have known the information below:

- the best feasible point x^{k-1} at present;
- the set Ψ_{k-1} to consist of the some partitioning subrectangles Y ;
- the index set I_{k-1} to consist of the indexes of the rectangles in Ψ_{k-1} ;
- the set L_{k-1} to consist of the left-below extreme points of the rectangles in Ψ_{k-1} ;
- a supper bound β_{k-1} of the global optimal value of the problem (LMP);
- a lower bound α_{k-1} of the global optimal value of the problem (LMP);
- $X = \Phi$ (empty set).

k1. Select a rectangle Z in Ψ_{k-1} which has the longest edge, and gain the index γ in correspondence with Z .

k2. Determine a feasible points \bar{x} such that $f(\bar{x})$ is the nearest-distance point from $f(l^Z)$ to $Z \cap f(D)$ (l^Z is the left-below extreme point of Z), and let $X = X \cup \{\bar{x}\}$. If $f(\bar{x}) = l^Z$, let $\Psi_{k-1} = \Psi_{k-1} - \{Z\}$, go to k1; otherwise, go to k3.

k3. By the partitioning technique in Section 3, obtain the set Ω_k after partitioning Z , and J_k to consist of the indexes of the subrectangles in Ω_k . If $|\Omega_k| = 1$, then $\Psi_{k-1} = \Psi_{k-1} - \{Z\}$, go to k1; otherwise, if $f_j(\bar{x}) \neq l_j^{k,p+1}, j = 1, 2, \dots, p,$

then delete $Y^{k,p+1}$ in Ω_k ; if $f_j(\bar{x}) \neq u_j^{k,0}$, $f_j(\bar{x}) \leq l_j^{k,0}$, $j = 1, 2, \dots, p$, then delete $Y^{k,0}$ in Ω_k , the set to consist of the remainder rectangles in Ω_k is yet noted as Ω_k ; go to k4.

k4. Let $\bar{L}_k = \{l^\xi | \xi \in J_k\}$ (l^Z is the left-below extreme point of $Z \in \Omega_k$);

$$\begin{aligned} L_k &= (L_{k-1} - \{l^\gamma\}) \cup \bar{L}_k; \\ I_k &= (I_{k-1} - \{\gamma\}) \cup J_k; \\ \Psi_k &= (\Psi_{k-1} - \{Z\}) \cup \Omega_k. \end{aligned}$$

k5. Compute the new lower bound α_k and the new supper bound β_k and the best feasible point x^k at present:

$$\begin{aligned} \alpha_k &= \min\{\bar{\phi}(l) : l \in L_k\}; \\ \beta_k &= \min\{\phi(x^{k-1}), \min\{\phi(x) | x \in X\}\}; \\ x^k &= \arg \alpha_k = \arg \min\{\phi(x^{k-1}), \min\{\phi(x) | x \in X\}\}. \end{aligned}$$

k6. If $\beta_k - \alpha_k \leq \varepsilon$, let $x^* := x^k$, $\phi^* := \beta_k$, outcome x^* , ϕ^* , stop; otherwise; go to k7.

k7. Delete subrectangle $Z = \{Y \in R^p | l^Z \leq y \leq u^Z\}$ to satisfy $\bar{\phi}(l^Z) \geq \beta_k$ in Ψ_k . The set to consist of the remainder rectangles in Ψ_k is yet noted as Ψ_k . Go to Step $k + 1$.

Remark 4.1. In $k2$, we can obtain \bar{x} by solving a linear program or a convex quadratic program.

Now, we give a convergent theorem of the algorithm and prove it.

Theorem 4.1. Let $\varepsilon = 0$. If the algorithm stops after finite step iterations, then the feasible point x^k to be produced when stopping is a globally optimal solution of the problem (LMP). If the algorithm can not be stop after finite step iterations, then each accumulation point \hat{x} of the array $\{x^k\}$ to be produced by the iteration process of the algorithm is a global optimal solution of the problem (LMP).

Proof. In the iteration process of the algorithm, there are that $\alpha_k \leq \phi(x^*) \leq \beta_k$, $k = l, 2, \dots$

If the algorithm stops after finite step iterations, then $\alpha_k = \phi(x^*) = \beta_k$, i.e. x^k is a global optional solution of the problem (LMP).

Suppose that the algorithm can not be stop after finite step iterations, and \hat{x} is an accumulation point of the array $\{x^k\}$ to be produced by the iteration process of the algorithm. Because the partitioning rule in the algorithm is exhaustive, there is a point x^{k_s} of the array $\{x^k\}$ and the rectangle $Z = Z(\delta)$ in correspondence with x^{k_s} are belong to the neighbour field $O(\hat{x}, \delta)$ for any $\delta > 0$.

Without loss of generality, suppose that $\delta = \frac{1}{10\nu}$ ($\nu \in Nk$) $\rightarrow 0$, as $\nu \rightarrow +\infty$ and the rectangle array $\{Z(\delta)\}$ is monotone decreasing and embedded as $\nu \rightarrow +\infty$. Let the left-below extreme point of $Z = Z(\delta)$ be $l^Z = l^{Z(\delta)}$, so

$$\alpha_{k_s} = \bar{\phi}(l^{Z(\delta)}) \leq \phi(x^*) \leq \phi(x^{k_s}) = \beta_{k_s}, \text{ for } \delta > 0;$$

let $\delta \rightarrow 0$, so $k_\delta \rightarrow +\infty$, then

$$\lim_{k_\delta \rightarrow +\infty} \alpha_{k_\delta} = \lim_{k_\delta \rightarrow +\infty} Z(\delta) = \lim_{k_\delta \rightarrow +\infty} x^{k_\delta} = \hat{x}$$

Thus $\phi(x^*) = \phi(\hat{x})$, i.e. each accumulation point l of the array $\{x^k\}$ to be produced by the iteration process of the algorithm is a global optimal solution of the problem (LMP).

5 Numerical Computation

This section will demonstrate the effectiveness of the proposed algorithm with an example from Ref.12 and some randomly produced problems (LMPs).

The proposed algorithm is compiled with FORTRAN 90, and we use the simplex method to solve linear programming problems, and two-level partition technique. All computations are carried out on personal computer for CPU: C4-1.7G and RAM: 256M.

Example 5.1. The following example on the problem (LMP) comes from Ref.12

$$\left\{ \begin{array}{l} \min (x_1 + x_2)(x_1 - x_2 + 7) \\ \text{s.t.} \left\{ \begin{array}{l} 2x_1 + x_2 \leq 14 \\ x_1 + x_2 \leq 10 \\ -4x_1 + x_2 \leq 0 \\ 2x_1 + x_2 \geq 6 \\ x_1 + 2x_2 \geq 6 \\ x_1 - x_2 \leq 3 \\ x_1 \leq 5 \\ x_1 + x_2 \geq 0 \\ x_1 - x_2 + 7 \geq 0 \end{array} \right. \end{array} \right.$$

where $p = 2, n = 2, m = 9$ for the problem (LMP).

Let $\varepsilon = 10e - 5$ and computational precision be double-precision. The times of iteration in solving Example 1 with the proposed algorithm is 53 and computational time is 0.3 second, and eighty linear programs are solved, and at last global optimal solution of example 1 is (2,8) which is the same as the one from Ref.12.

Now, we randomly produce the problem (LMPs), in order to further demonstrate that the effectiveness of the proposed algorithm.

We use the quasi-random function and random seed in FORTRAN 90 to produce $c_j \in R^n$ and $d_j \in R$ for $j = 1, 2, \dots, p$ and $A \in R^{m \times n}$ and $b \in R^m$ in the problem (LMP). The random range is (-50,50) and (10,10). N_T is noted as total number of branch and bound iterations.

We test results to solve the problem (LMP) below:

(1) When $p = 2$, we randomly produce 10 examples. As m, n are increasing, the iteration time of the proposed algorithm is increasing too. To the examples randomly produced for $m \leq 180, n \leq 160$, the iteration time of the proposed

algorithm is less than 65 seconds, and for $m = 220$, $n = 200$ the iteration time is 305 seconds and N_T is 1234617.

(2) When $p = 3, 4, 5$, we randomly produce 12, 10, 9 examples, respectively, and make m, n equal to 220, 200 at most respectively. The iteration time to two examples for $p = 5$, $m = 220$, $n = 200$ are 1824 seconds and 324 seconds respectively, and N_T 's are 803 and 570 respectively.

(3) When $p = 6, 7, 8$, we randomly produce 4, 1, 2 examples, respectively, and make m, n equal to 160, 150 at most, respectively. The iteration time to an example randomly produced for $p = 8$, $m = 160$, $n = 150$ is 312 seconds and N_T is 3457.

Thus it is shown by above numerical computation that the proposed algorithm is effective.

References

1. Bennen K.P.: Global tree optimization: A non-greedy decision tree algorithm, *Computing Sciences and Statistics*. **26** (1994) 156-160.
2. Benson, H.P.: Vector maximization with two objective functions, *Journal of Optimization Theory and Applications*. **28** (1979) 253-257.
3. Konno, H., Watanabe, H.: Bond portfolio optimization problems and their application to index tracking: a partial optimization approach, *Journal of the Operations Research Society of Japan*. **39** (1996) 285-306.
4. Konno, H. and Kuno, T.: Generalized linear multiplicative and fractional programming, *Annals of Operations Research*. **25** (1990) 147-161.
5. Falk, J.E. and Palocsa, S.W. (1994), Image space analysis of generalized fractional programs, *Journal of Global Optimization* 4(1), 63-88.
6. Matsui, T.: NP-Hardness of linear multiplicative programming and related problems *Journal of Global Optimization*. **9(2)** (1996) 113-119.
7. Konno, H., Kuno, T. and Yajima, Y.: Global optimization of a generalized convex multiplicative function, *Journal of Global Optimization*. **4** (1994) 47-62.
8. Schaible, S. and Sadini, C.: Finite algorithm for generalized linear multiplicative programming, *Journal of Optimization Theory and Applications*. **87(2)** (1995) 41-55.
9. Benson, H.P., Boger, G.M.: Outcome-space cutting-plane algorithm for linear multiplicative programming, *Journal of Optimization Theory and Applications*. **104(2)** (2000) 301-332.
10. Benson, H.P. and Boger, G. M.: Multiplicative programming problems: analysis and efficient point search heuristic, *Journal of Optimization Theory and Applications*. **94(2)** (1997) 487-510.
11. Domeich, M.C. and Sahinidis, N.V.: Global optimization algorithms for chip design and compaction, *Engineering Optimization*. **25(2)** (1995) 131-154.
12. Ryoo H.S., Sahinidis N.V.: Global optimization of multiplicative programs *Journal of Global Optimization*. **26** (2003) 387-418.

A Binary Ant Colony Optimization for the Unconstrained Function Optimization Problem

Min Kong and Peng Tian

Shanghai Jiaotong University, Shanghai 200052, China
kongmin@sjtu.edu.cn
ptian@sjtu.edu.cn

Abstract. This paper proposes a Binary Ant System (BAS), a binary version of the hyper-cube frame for Ant Colony Optimization applied to unconstrained function optimization problem. In BAS, artificial ants construct the solutions by selecting either 0 or 1 at every bit stochastically biased by the pheromone level. For ease of implementation, the pheromone value is designed specially to directly represent the probability of selection. Principal settings of the parameters are analyzed and some methods to escape local optima, such as local search and pheromone re-initialization are incorporated into the proposed algorithm. Experimental results show that the BAS is able to find very good results for the unconstrained function optimization problems of different characteristics.

1 Introduction

Ant Colony Optimization (ACO) is a stochastic meta-heuristic for solutions to combinatorial optimization problems. Since its first introduction by M. Dorigo and his colleagues [5] in 1991, ACO has been successfully applied to a wide set of different hard combinatorial optimization problems [4, 6], such as traveling salesman problems, quadratic assignment problems, and vehicle routing problems. The main idea of ACO is the cooperation of a number of artificial ants via pheromone laid on the path. Each ant contributes a little effort to the solution, while the final result is an emergent result of the ants' interactions.

Although ACO has been proved to be one of the best meta-heuristics in some combinatorial optimization problems [4, 6], the application to the function optimization problem appears a little difficult, since the pheromone laying method is not straightforward in function optimization problems, specially in the continuous domain. The first attempt of the ACO to the function optimization problem is the Continuous ACO (CACO), which is proposed by Bilchev and Parmee [1] in 1995, and was extended by some other research [2, 11, 13]. Other attempts include the API algorithm by Monmarchè [12], Continuous Interacting Ant Colony (CIAC) by Dréo and Siarry [7, 8], and the extended ACO application, proposed by Krzysztof [10].

This paper proposes BAS, a modification of the hyper-cube framework (HCF) for ACO [3] to deal with the unconstrained function optimization problems.

2 The Binary Ant System

The function optimization problem can be described as a triple (D, F, f) , where D is the set of solutions, F is the set of feasible solutions, and f is the objective function. A solution x^* is called a global optimal solution if $f(x^*) = \min\{f(x)|x \in F\}$. Here we only discuss the minimization problems, since any maximization problem can be easily transferred to the relative minimization problem. This paper concerns with the binary function optimization problem, in which a solutions is represented by a binary bit string: $x = \{x_1, \dots, x_n\}, x_j \in \{0, 1\}$.

In BAS, pheromone are associated to every bit of the solution string for the selection of 0 and 1. τ_{js} represents the pheromone level for node x_j to select $s \in \{0, 1\}$.

An initial pheromone level τ_0 is distributed on all the selections. A number m of ants construct their solutions sequentially at every iteration. During the solution construction, ant k at bit j makes selection of either 0 or 1 according to the following probability distribution:

$$p_{js}^k(t) = \frac{\tau_{js}(t)}{\sum_{l \in \{0,1\}} \tau_{jl}(t)}, s \in \{0, 1\} \tag{1}$$

Every ant makes a complete selection of all the bits to generate a solution. After all the ants have completed their tours, all the solutions generated during the current iteration are evaluated and compared by the objective function $f(x)$.

BAS keeps record of the best solution found so far, which is called the global best solution S^{gb} . If the current iteration best solution S^{ib} is better than S^{gb} , S^{gb} is updated to S^{ib} . At the end of each iteration cycle, a global pheromone update process is performed, which includes two phases, the first phase is called the evaporation phase, in which all the pheromone evaporate a little according to the following evaporation rule:

$$\tau_{js}(t + 1) \leftarrow (1 - \rho)\tau_{js}(t) \tag{2}$$

Where $\rho \in [0, 1]$ is the evaporation parameter. Then, the second phase is performed to intensify the pheromone level of those selections belonging to S^{gb} :

$$\tau_{js}(t + 1) \leftarrow \tau_{js}(t + 1) + \rho\Delta\tau, (j, s) \in S^{gb} \tag{3}$$

Where $\Delta\tau$ is a parameter representing the amount of pheromone intensified.

The iteration loop is repeated until some end conditions are met, such as a maximum number of function evaluations has been performed or a satisfied solution is found.

3 Theoretical Analysis

As described in the previous section, BAS is a quite simple algorithm which only has 4 parameters to be set, which are $m, \tau_0, \Delta\tau,$ and ρ .

For parameter m and ρ , we set them according to the experience from other previous study of ACO. In particular, we set $m = n$, $\rho = 0.1$ for all the test executed in this paper.

To discuss the setting of τ_0 and $\Delta\tau$, we first give a proposition to indicate that the pheromone level on every selection are bounded by a top limit:

Proposition 1. *For any pheromone value τ_{js} in BAS, the following holds:*

$$\lim_{t \rightarrow \infty} \tau_{js}(t) \leq \Delta\tau \tag{4}$$

Proof. The maximum possible pheromone value for any selection (j, s) is that for any iteration, this selection belongs to S^{gb} , thus gets pheromone intensification at every iteration. Therefore, the pheromone update of the selection (j, s) is bounded by:

$$\begin{aligned} \tau_{js}^{max}(t) &= (1 - \rho)\tau_{js}^{max}(t - 1) + \rho\Delta\tau \\ &= (1 - \rho)^t\tau_0 + \sum_{i=1}^t (1 - \rho)^{t-i}\rho\Delta\tau \end{aligned} \tag{5}$$

Asymptotically, because $0 < \rho < 1$, when $t \rightarrow \infty$, this sum converges to $\Delta\tau$.

It would be much simple and easy if the pheromone value of each selection could be directly used as the probability distribution. At the initial stage, it is easy to set $\tau_0 = 0.5$ on every selection to achieve this target with an evenly probability distributed. At the following, we will point out that this characteristic of probability representing holds if we set reasonable parameters.

Proposition 2. *If $\Delta\tau = 1$, and the initial pheromone values on all the selections represents some probability distribution in BAS, at any iteration steps followed, the pheromone values still directly represent a probability distribution.*

Proof. To prove this proposition, what we need to do is to prove $\tau_{j0}(t) + \tau_{j1}(t) = 1$, $0 \leq \tau_{js}(t) \leq 1$ holds for every node j under the condition that $\tau_{j0}(t - 1) + \tau_{j1}(t - 1) = 1$.

From Proposition 1, we can see that for any selection (j, s) , the following holds:

$$0 \leq \tau_{js}(t) \leq \Delta\tau = 1 \tag{6}$$

After the pheromone update procedure, all the pheromone values are evaporated, and there must be one and only one of τ_{j0} and τ_{j1} belongs to S^{gb} that receives pheromone intensification, therefore, for any note j , we have:

$$\begin{aligned} \tau_{j0}(t) + \tau_{j1}(t) &= (1 - \rho)\tau_{j0}(t - 1) + (1 - \rho)\tau_{j1}(t - 1) + \rho\Delta\tau \\ &= (1 - \rho)(\tau_{j0}(t - 1) + \tau_{j1}(t - 1)) + \rho \\ &= 1 - \rho + \rho = 1 \end{aligned} \tag{7}$$

Therefore, we set $\Delta\tau = 1$, and $\tau_0 = 0.5$ Thus the probability distribution described in (1) is replaced by:

$$p_{js}^k(t) = \tau_{js}(t), s \in \{0, 1\} \tag{8}$$

To speed the convergence, we utilize some additional feature to help BAS to jump out of local optima.

One feature we used is the local search method, which has been verified effective in quality improvement by most of the previous study [4, 6]. In the binary function optimization problem, we use a simple one flip method as the local search, that is, checking every bit of the solution by flipping 1 to 0 or 0 to 1, to see if the objective function value gets improved. The local search method is performed at the end of every iteration, and only for S^{ib} and S^{gb} .

Another feature we used is the pheromone re-initialization. Intuitively, as the iteration goes on in BAS, the pheromone value will go to two extremes, with $\tau_{max} \rightarrow 1$ and $\tau_{min} \rightarrow 0$. Obviously, this will lead to premature. To avoid this, a pheromone re-initialization procedure is performed once BAS gets near to premature. We judge the premature by a convergence factor defined as:

$$cf = \frac{\sum_{j=1}^n |\tau_{j0} - \tau_{j1}|}{n} \tag{9}$$

Under this definition, when the algorithm is initialized with $\tau_0 = 0.5$ for all the selection, $cf = 0$, while when the algorithm gets into convergence or premature, $cf = 1$. In BAS, once $cf \geq 0.9$, the pheromone re-initialization procedure is performed with all the pheromone value set to τ_0 . But the global best solution S^{gb} is recorded and is intensified directly after the re-initialization procedure, in order to make the algorithm keep search around S^{gb} .

4 Experimental Results

Two sets of function optimization problem are tested in this paper. The first set use the De Jong’s suite of test functions to verify that BAS is able to deal with different kinds of problems. The second test set is to compare BAS with other Ant System. Due to the space limitation, we can not list these test functions in this paper, please refer to [9] and [8] for detailed description. A test is considered to be successful if it observes the following condition:

$$|f^* - f_{known_best}| < \epsilon_1 \cdot |f_{known_best}| + \epsilon_2 \tag{10}$$

Where f^* is the optimum found by BAS, f_{known_best} is the global optimum of the test function. ϵ_1 and ϵ_2 are accuracy parameters, which is set to be: $\epsilon_1 = \epsilon_2 = 10^{-4}$.

The De Jong suite of test functions are tested with the parameter setting described in the previous section. The maximum number of function evaluation is set to 100,000 for all the 5 De Jong test functions.

Tab. 1 reports the test results. For all the test functions except $f4$, BAS is able to find the minimum solution in every trail. Meanwhile, BAS is also able to find very good solution around the minimum value for $f4$, which includes the Gaussian noise.

There are several previous ant system dealing with function optimization problems [1, 2, 7, 8, 10, 11, 12, 13]. But it is difficult to compare all of them because

Table 1. Test Results on DeJong Suite of tests. For each instance, the table reports the known minimum fitness value from DeJong[9], the best and average solutions over 100 runs, followed by the best and average number of function evaluations until the algorithm stop.

Problem	Known_Best	Best_Fitness	Avg_Fitness	Best_Evals	Avg_Evals
<i>f1</i>	0	0	0	103	6687.97
<i>f2</i>	0	0	0	54	580.53
<i>f3</i>	-30	-30	-30	106	1208.32
<i>f4</i>	0	0.20	0.85		[100,000]
<i>f5</i>	1	1	1	260	5953.21

they are using different test functions in the experiments. We select 7 mostly used test functions as the test bed for comparison. For all the tests, we use general parameter settings as described in the previous section, that is $m = n$, $\tau_0 = 0.5$, $\Delta\tau = 1$, $\rho = 0.1$, and the maximum number of function evaluation is set to 150,000.

Tab. 2 reports the test results. It is clear from the comparison in all the benchmarks, that BAS is at least one of the best algorithms when only considering the success rate. Meanwhile, BAS is also comparable in average number of function evaluations. However, more detailed analysis on a wider sample of test functions, and other measures of the performance, such as variance, stability, accuracy, or even the ease of the implementation, will have to be performed to indicate particular advantages or disadvantages of BAS.

Table 2. Comparison of success rate and average number of function evaluations on different benchmark problems. Results for some of the algorithms were not available, hence some entries are missing. The brackets indicate that the results are based on the runs with a fixed number of evaluations.

<i>f</i>	CACO		API		CIAC		ACO		BAS	
	% ok	evals	% ok	evals	% ok	evals	% ok	evals	% ok	evals
<i>R2</i>	100	6842		[10000]	100	11797		2905	100	2580.53
<i>SM</i>	100	22050		[10000]	100	50000		695	100	8887.66
<i>GP</i>	100	5330			56	23391		364	100	2317.54
<i>MG</i>	100	1688			20	11751			100	2159.62
<i>St</i>		[6000]			94	28201			100	1208.32
<i>Gr₅</i>				[10000]	63	48402			100	41963.95
<i>Gr₁₀</i>	100	50000			52	50121			100	89054.62

5 Conclusions

This paper presented BAS, a binary version of hyper-cube frame of ACO to handle binary optimization problems. In the proposed version of the system, pheromone trails are put on the selections of 0 and 1 for each bit of the solution

string, and they directly represent the probability of selection. Principal settings of the parameters are discussed, and some methods to escape from local optima are also presented. Experimental results show that BAS performs very well on the De Jong suite of test functions, and as a whole, BAS outperforms other previous ACO related algorithms in success rate and with a comparable number of function evaluation. The results reported in the previous section demonstrate that BAS is capable of solving these various problems very rapidly and effectively.

References

1. Bilchev G., Parmee I.C.: The ant colony metaphor for searching continuous design spaces. In: Fogarty T.C. (eds.): Proceedings of the AISB Workshop on Evolutionary Computation. LNCS Vol. 993. Springer-Verlag, Berlin, Germany (1995) 25–39
2. Bilchev G., Parmee I.C.: Constrained optimization with an ant colony search model. Proceedings of 2nd International Conference on Adaptive Computing in Engineering Design and Control, PEDC (1996) University of Plymouth 26–28
3. Blum C., Dorigo M.: The hyper-cube framework for ant colony optimization. IEEE Tran. on Man, Systems and Cybernetics(B) (2004) Vol 34(2) 1161–1172
4. Bonabeau E., Dorigo M., Theraulaz G.: Swarm intelligence: from natural to artificial systems. Oxford University Press (1999)
5. Dorigo M., Maniezzo V., Colnani A.: Positive feedback as a search strategy. Technical Report 91-016, DIPARTIMENTO DI ELETTRONICAQ – POLITECNICO DI MILANO (1991)
6. Dorigo M., Stützle T.: Ant colony optimization. The MIT Press, Cambridge, Massachusetts, London, England (2004)
7. Dreoj J., Siarry P.: A new ant colony algorithm using the heterarchical concept aimed at optimization of multim minima continuous functions. In: Dorigo M., Caro G.D., Sampels M. (eds.): Proceedings of the Third International Workshop on Ant Algorithms LNCS Vol. 2463 Springer-Verlag, Berlin, Germany (2002) 216–221
8. Dreoj J., Siarry P.: Continuous interacting ant colony algorithm based on dense heterarchy. Future Generation Computer Systems (2004) Vol. 20, 841–856
9. De Jong K.A.: An analysis of the behavior of a class of genetic adaptive systems. Doctoral dissertation, University of Michigan (1975)
10. Krzysztof Socha: ACO for continuous and mixed-variable optimization. In: Dorigo M. et al. (Eds.): ANTS 2004, LNCS 3172, (2004) 25–36
11. Mathur M., Karale S.B., Priye S., Jyaraman V.K., and Kulkarni B.D.: Ant colony approach to continuous function optimization. Ind. Eng. Chem. Res. (2000) Vol. 39 3814–3822
12. Monmarchè N., Venturini G., and Slimane M.: On how *Pachycondyla apicalis* ants suggest a new search algorithm. Future Generation Computer Systems (2000) Vol. 16 937–946
13. Wodrich M., Bilchev G.: Cooperative distributed search: the ant’s way. Control & Cybernetics (1997) Vol. 3 413–446

Mining Dynamic Association Rules in Databases

Jinfeng Liu and Gang Rong

National Key Laboratory of Industrial Control Technology,
Zhejiang University, China
{jinfeng, grong}@iipc.zju.edu.cn

Abstract. We put forward a new conception, *dynamic association rule*, which can describe the regularities of changes over time in association rules. The dynamic association rule is different in that it contains not only a support and a confidence but also a *support vector* and a *confidence vector*. During the mining process, the data used for mining is divided into several parts according to certain time indicators, such as years, seasons and months, and a support vector and a confidence vector for each rule are generated which show the support and the confidence of the rule in each subsets of the data. By using the two vectors, we can not only find the information about the rules' changes with time but also predict the tendencies of the rules, which ordinary association rules can not offer.

1 Introduction

Association rules mining is a widely used method in data mining field. And many algorithms have been developed for mining association rules, but almost all of them assume that the rules having been found in the datasets are valid all the time and don't change. These rules are considered as static ones.

In fact, it is possible that a rule won't be valid all the time. For example, while analyzing one year's transaction data of a supermarket, we may find a rule that 90% of the customers who buy Christmas trees also buy Christmas gifts, but if we learn it more carefully, we may discover that most of the data supporting the rule is collected in December and the rule would be not applicable in other months. This example indicates that rules are changing over time.

R.Agrawal and R.Srikant introduced the sequential patterns mining in 1994 and this pattern takes the time sequence into consideration, but it still assumes that the patterns are effective anytime and is also static [1]; G.Dong and J.Li put forward the emerging patterns which takes the changes of rules into consideration in 1999, but it is limited in rules whose support increase significantly from one dataset to another [2]; Wai-Ho Au and Keith C.C.Chan employed the fuzzy sets and residual analysis to discover the regularities about the changes in association rules over time in 2002, but they were limited in a small aspect of prediction [3].

In this study, we take the changes of rules into consideration, and present a new conception, *dynamic association rule*. In order to find such rules, we divide the datasets into several parts according to certain time indicators, such as years,

seasons and months. Correspondingly, we use the *support vectors* and *confidence vectors* together with the support and the confidence values to assess the rules.

2 Dynamic Association Rule

2.1 Definition of Dynamic Association Rule

The *dynamic association rule* is a type of association rule which can describe the changes of itself. It can be defined as follows: Let $I = \{i_1, \dots, i_m\}$ be a set of items. Let D be the task-relevant data, which is collected during a time period t . And t can be divided into n disjoint time segments which make a time series $t = \{t_1, \dots, t_n\}$. According to t , D can be divided into n subsets, i.e., $D = \{D_1, \dots, D_n\}$, where $D_i, i \in \{1, \dots, n\}$ is collected in the time segment $t_i, i \in \{1, \dots, n\}$. Let D be a set of database transactions where each transaction T is a set of items such that $T \subseteq I$. Each transaction is associated with an identifier, called *TID*. Let A be a set of items. A transaction T is said to contain A if and only if $A \subseteq T$. A dynamic association rule is an implication of the form $A \Rightarrow B$, where $A \subset I$, $B \subset I$, and $A \cap B = \emptyset$. The rule $A \Rightarrow B$ holds in the transaction set D with support s , where s is the percentage of transactions in D that contain $A \cup B$ (i.e., both A and B). This is taken to be the probability, $P_D(A \cup B)$. Let $P_D[(A \cup B)_i]$ be the ratio of the number of transactions in D_i that contain $A \cup B$ to the total records number of D . Then support s of $A \Rightarrow B$ can also be put as $\sum_{i=1}^n P_D[(A \cup B)_i]$. The rule $A \Rightarrow B$ has confidence c in the transaction set D if c is the percentage of transactions in D containing A that also contain B . This is taken to be the conditional probability, $P_D(B|A)$. Let $P_D(B_i|A)$ be the ratio of the number of transactions in D_i that contain $A \cup B$ to the number of transactions in D that contain A . Then confidence c can also be put as $\sum_{i=1}^n P_D(B_i|A)$.

2.2 Support Vector

We introduce the support vector SV and the confidence vector CV together with the support s and the confidence c to assess a rule.

The support vector of an itemset is defined as: $SV = [s_1, \dots, s_n]$, where $s_i, i \in \{1, \dots, n\}$ is the ratio of $f_i, i \in \{1, \dots, n\}$, which is the occurrence frequency of itemset A in $D_i, i \in \{1, \dots, n\}$, to M which is the number of records contained in D . That is,

$$s_i = \frac{f_i}{M}, \quad i \in \{1, \dots, n\} \quad (1)$$

Let the support of itemset A be s , then,

$$s = \sum_{i=1}^n s_i \quad (2)$$

Let the minimum support be min_sup , so if $s \geq min_sup$ is true, then the itemset A is a frequent itemset.

Sometimes, it may be more suitable to use the occurrence frequency of one itemset in the support vector. Then a support vector can be put as $SV = [f_1, \dots, f_n]$, and the corresponding support should be $s = \sum_{i=1}^n f_i$.

2.3 Confidence Vector

Since there is no difference in the generation of association rules from frequent itemsets between dynamic rules and ordinary ones except for the calculating of confidence vectors, thus we only focus on the generation of confidence vectors.

The confidence vector of a dynamic association rule $A \Rightarrow B$ is defined as: $CV = [c_1, \dots, c_n]$, where $c_i, i \in \{1, \dots, n\}$ is a value between 0%~100%. Let the support vector of itemset $A \cup B$ be $SV_{A \cup B} = [s_{A \cup B1}, \dots, s_{A \cup Bn}]$, the support vector of A be $SV_A = [s_{A1}, \dots, s_{An}]$, and the support vector of itemset B be $SV_B = [s_{B1}, \dots, s_{Bn}]$. And the support of itemset A be s_A . Then,

$$c_i = \frac{s_{A \cup Bi}}{\sum_{i=1}^n s_{Ai}} = \frac{s_{A \cup Bi}}{s_A}, \quad i \in \{1, \dots, n\} \tag{3}$$

Let the support of itemset $A \cup B$ be $s_{A \cup B}$, the support of itemset B be s_B , and let the confidence of rule $A \Rightarrow B$ be c . Then,

$$c = \frac{\sum_{i=1}^n s_{A \cup Bi}}{\sum_{i=1}^n s_{Ai}} = \frac{s_{A \cup B}}{s_A} = \sum_{i=1}^n c_i \tag{4}$$

Let the minimum confidence be min_conf , so if $c \geq min_conf$ is true, then the rule $A \Rightarrow B$ is a strong association rule.

2.4 A Whole Dynamic Association Rule

A whole dynamic association rule can be described as:

$$A \Rightarrow B (SV = [s_1, \dots, s_n], CV = [c_1, \dots, c_n], s, c) \tag{5}$$

where SV, SC, s and c together describe the characters of the rule.

As shown in (5), the support and the confidence of a rule are also offered in a dynamic association rule, it can characterize a rule with supplying all the information that an ordinary one could do.

Moreover, because the support and the confidence vectors are related to time, they could offer some information about the changes of the rules over time, which ordinary ones could not do.

3 Two Mining Algorithms

The preparation of the data for mining dynamic association rules needs to consider the time factor. And each record of the data should contains a time indicator. We may call it *Time_id* which is used to divide the data into different datasets.

The two algorithms introduced below concentrate on the generation of frequent itemsets and their support vectors. The generation of dynamic association rules is as the same as ordinary ones and their confidence vectors and confidences can be obtained by (3) and (4).

3.1 The First Algorithm

This algorithm may be time-consuming but is very simple. It bases on mature association rule mining algorithms and adopts some very simple improvements.

In order to describe the algorithm clearly, let the whole database be D which is divided into n subsets $D_1 \sim D_n$; Let the whole frequent itemsets be L and l_i be an itemset in L ; Let f_{ij} be the frequency of itemset l_i in subset D_j , $j \in \{1, \dots, n\}$. In the first place, it uses an association rule mining algorithm, such as Apriori and FP-growth, to find L in D . In the second place, it scans $D_1 \sim D_n$ to find f_{ij} , $j \in \{1, \dots, n\}$ of l_i . Finally, it calculates the support vector and the support for each itemset using (1) and (2). The outline of this algorithm is described as below:

Input: Database D and its subsets $D_1 \sim D_n$, min_sup
Output: L , frequent itemsets in D with support vectors

```

1:  L=Association-mining-algorithm
2:  for(j=1;j≤n;j++) {
3:    for each itemset  $l_i \in L$  {
4:      scan  $D_j$  for frequency  $f_{ij}$ 
5:       $s_{ij} = f_{ij}/M$ 
6:    }
7:  }
8:  for each itemset  $l_i \in L$  {
9:     $SV_i = \{s_{i1}, \dots, s_{in}\}$ 
10:    $s_i = \sum_{j=1}^n s_{ij}$ 
11:  }
12:  return  $L$  with support vectors

```

The function of the procedure Association-mining-algorithm is to call an ordinary association mining algorithm, such as Apriori and FP-growth, to find the frequent itemsets L . And M is the number of records in the database D .

3.2 The Second Algorithm

This algorithm bases on the classic Apriori [4] and adopts two improvements in order to generate support vectors. In this algorithm, a support vector uses occurrence frequencies as its elements. And the main character of this algorithm is that it can calculate the support vector while finding the frequent itemsets. The improvements are described as follows.

First, in the first step of finding 1-itemset candidates which is the foundation of the whole algorithm, the algorithm scans every subset of the database and records the occurrence frequencies of the candidates in every subset, so a support vector can be obtained for each candidate. And the support vectors are used to calculate the occurrence frequencies of candidates in order to find the frequent 1-itemsets. Therefore, the frequent 1-itemsets with support vectors can be generated.

Secondly, in the recursion process whose function is to find the frequent k -itemsets, the algorithm also scans every subset of the database to count the frequencies of candidates generated from frequent $(k - 1)$ -itemsets in different subsets, then the support vectors of k -itemset candidates can be acquired. And these support vectors are also used to calculate the frequencies of candidates, thus, the frequent k -itemsets can be found as well as their support vectors. The details of the algorithm are described as below.

Input: Subsets, $D_1 \sim D_n$, of database D , min_sup

Output: L , frequent itemsets in D with support vectors

```

1:  for each  $D_i, i \in \{1, \dots, n\}$  //scan  $D_i$  for frequencies
2:     $C_{1i} = \text{Scan-support-1-itemset}(D_i)$ 
3:     $C_1 = \text{Join-support-vector}(D_1 \text{ to } D_n)$  //find candidate  $C_1$  with SV
4:     $L_1 = \{c \in C_1 \mid \sum_{i=1}^n c.frequency_i \geq min\_sup\}$ 
5:    for ( $k=2; L_{k-1} \neq \emptyset; k++$ ) {
6:       $C_k = \text{Apriori-gen}(L_{k-1}, min\_sup)$ 
7:      for each  $D_i$  {
8:        for each transaction  $t \in D_i$  {
9:           $C_t = \text{subset}(C_k, t)$ 
10:         for each candidate  $c \in C_t$ 
11:            $c.frequency_i++$ 
12:         }
13:       }
14:       $L_k = \{c \in C_k \mid \sum_{i=1}^n c.frequency_i \geq min\_sup\}$ 
15:    }
16:  return  $L = \cup_k L_k$  with support vectors

```

The procedure Apriori-gen is not presented here, it could be found in [5], and its function is to generate C_k candidates from L_{k-1} . The function of the procedure Scan-support-1-itemset is to count frequency for each 1-itemset, and the function of the procedure Join-support-vector is to merge the frequencies of each 1-itemset in all subsets together to form support vectors.

4 Two Ways to Use Dynamic Association Rules

The dynamic association rules contain support vectors and confidence vectors. They could be used in many ways to get comprehensive information about the

rules. Histograms and time series analysis are two of these ways. The details of the two approaches are discussed below.

4.1 Histograms

Histograms of support vectors or confidence vectors can describe the distribution of the supports or the confidences of rules clearly. Moreover, the variations of supports and confidences with time could be shown qualitatively. According to the definition of support vector and confidence vector, we could find that the variation tendency of the support and confidence of one rule keeps the same. Thus, we only need to draw a histogram of the support vector or the confidence vector.

If the confidence vector CV of one dynamic association rule is:

$$CV = [3.7\%, 4.44\%, 7.4\%, 22.2\%, 21.46\%, 22.94\%],$$

then, we can draw a histogram of the CV as Fig.1. As Fig.1 shown, the confidence

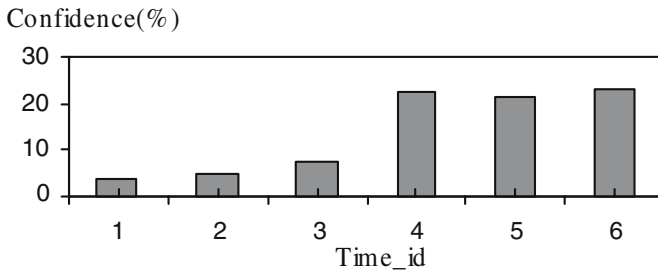


Fig. 1. The histogram of the confidence vector

of the rule has a large raise in the period of *Time_id* 4, and it maintains a high level in the following time segments. This is an ascending trend, and this character of the confidence suggests that the application of the rule could be effective.

From histograms of confidence vectors we may also find downtrends, periodic trends, etc. A downtrend may indicate that the rule's applicability is not very well. And a periodic trend may imply that the validity of the rule is not stable.

4.2 Time Series Analysis

Time series analysis is a wide used method in describing the characters of change and predicting the potential trend of data. If a support vector in the form of occurrence frequency of a rule contains enough elements, it may be suitable for the time series analysis. For instance, if the support vector of a dynamic association rule found in a 12-months-data of a supermarket is:

$$SV = [568, 574, 581, 582, 584, 586, 590, 594, 600, 606, 612, 613],$$

then, we could generate an auto-regression model for the frequency as below:

$$f_i - 1.4f_{i-1} = 8\varepsilon_t, \varepsilon_t \sim N(0, 1) \tag{6}$$

According to the model shown in equation (6), we could find the ascending trend of the frequencies and could predict the frequencies for the following months. A 3-months-prediction of the frequencies is: 617, 622, 624.

Through the time series analysis, we could find quantitative models for the support or confidence vectors of rules. It can give a more profound understand of the rules than histograms. And the most important is that we could use these models to predict the trends and values of the rules.

5 Application

The title of a paper reflects the main content of the paper. Thus, we hope to be able to find some useful information from the paper titles through mining dynamic association rules.

We collect 11077 paper titles concerning “data mining” which was published from 1990 to 2004 from Internet. And the titles are preprocessed in the following steps: 1) Some words, such as “a”, “an”, “of” and “for”, in the titles are removed. 2) Similar words in the titles are found and unified. For example, “mining” and “mine” are unified to “mining”. 3) Distinct words in the titles are found and assigned a “ID” which is an integer and the words in titles are converted to corresponding IDs in order to facilitate the mining algorithms. 4) The converted titles are divided into 8 subsets according to the publication time, and data of some successive years are merged into one subset in order to maintain the total number of titles in each subset approximately the same. 5) The data is stored into one database.

We use the support vectors in the form of occurrence frequency and the confidence vectors to assess the rules, and let the *min_sup* be 200 and the *min_conf* be 10%. Considering that there are so many research directions in data mining area, the 10% for *min_conf* is not a low value. And we do find some interesting rules. For example, a rule is:

$$\begin{aligned} data \wedge mining &\Rightarrow application \ (SV = [4, 18, 33, 22, 26, 34, 32, 37], \\ CV &= [0.2\%, 0.9\%, 1.7\%, 1.1\%, 1.3\%, 1.7\%, 1.6\%, 1.9\%], 206, 10.4\%) \end{aligned} \tag{rule 1}$$

The histogram of the support vector of rule 1 is shown in Fig.2., and it may show us that the application of data mining is a stable ascending trend though there is a fluctuation around *Time_id* 3 which represents the years 1998 and 1999. And this character of the rule may suggest that there are more and more researches which focus on the application of data mining.

For another interesting rule is:

$$\begin{aligned} discover &\Rightarrow knowledge \ (SV = [4, 42, 43, 46, 20, 40, 30, 25], \\ CV &= [0.8\%, 8.4\%, 8.6\%, 9.2\%, 4.0\%, 8.0\%, 6.0\%, 5.0\%], 250, 50.2\%) \end{aligned} \tag{rule 2}$$

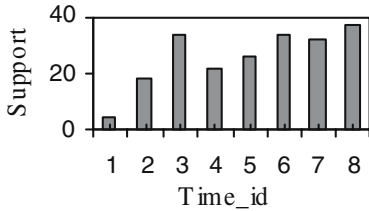


Fig. 2. The histogram of the support vector of rule 1

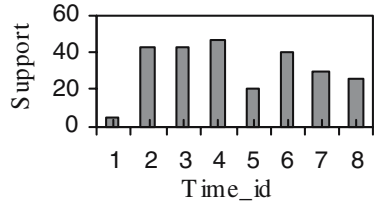


Fig. 3. The histogram of the support vector of rule 2

The histogram of the support vector of rule 2 is shown in Fig.3., and it shows us that the item of “discover knowledge” has a burgeoning increase in the first 4 segments which refer to years from 1990 to 2000 and follows a short fluctuation. However, it displays a continuous downtrend during the last 3 segments which represent years 2002, 2003 and 2004.

This character of the rule may suggest that researches in data mining become more and more profound, in other words, researches focus more on the details of some small aspects than on the whole framework.

6 Conclusion

We have proposed an original idea of dividing databases for mining into several subsets, and mine association rules contain not only supports and confidences but also support vectors and confidence vectors. Such a rule we call a *dynamic association rule* for it could provide information of the changes of the rule over time.

References

1. Agrawal, R., Srikant, R.: Mining sequential patterns. In *Proc. of the 11th Int'l Conference on Data Engineering*, March (1995) 3–14
2. Dong, G., Li, J.: Efficient mining of emerging patterns: Discovering trends and differences. In *Proc. of the Fifth Int'l Conference on Knowledge Discovery and Data Mining*, (1999) 43–52
3. Au, W.H., Chan, K.C.C.: Fuzzy data mining for discovering changes in association rules over time. In *Proc. of 2002 IEEE Int'l Conference on Fuzzy Systems*, May (2002) 890–895
4. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In *Proc of 20th Int'l Conference on Very Large Data Bases*, September (1994) 487–499
5. Han, J., Kamber, M.: *Data Mining Concepts and Techniques*. Morgan Kaufmann Publishers, San Fransisco, USA, (2000)

A Novel Typical-Sample-Weighted Clustering Algorithm for Large Data Sets*

Jie Li, Xinbo Gao, and Licheng Jiao

School of Electronic Engineering, Xidian University, Xi'an 710071, China
leejie@mail.xidian.edu.cn

Abstract. In the field of cluster analysis, most of existing algorithms are developed for small data sets, which cannot effectively process the large data sets encountered in data mining. Moreover, most clustering algorithms consider the contribution of each sample for classification uniformly. In fact, different samples should be of different contribution for clustering result. For this purpose, a novel typical-sample-weighted clustering algorithm is proposed for large data sets. By the atom clustering, the new algorithm extracts the typical samples to reduce the data amount. Then the extracted samples are weighted by their corresponding typicality and then clustered by the classical fuzzy c -means (FCM) algorithm. Finally, the Mahalanobis distance is employed to classify each original sample into obtained clusters. It is obvious that the novel algorithm can improve the speed and robustness of the traditional FCM algorithm. The experimental results with various test data sets illustrate the effectiveness of the proposed clustering algorithm.

1 Introduction

Cluster analysis is one of multivariate statistical analysis methods and one of important branches of unsupervised pattern recognition [1]. In the traditional cluster methods, the objective function based clustering algorithms became more and more popular for converting the cluster analysis into an optimization problem. Fuzzy c -means (FCM) is one of the most typical clustering algorithms [2], which has been widely applied to various fields. As well known, the computational complexity of the FCM algorithm is $O(ncl)$, where n is the sample number of the data set, c is the category number and l is the iteration generation. Compared to other clustering algorithms, the FCM has a high efficiency. Whereas, with the increasing of data amount or clusters number, the complexity of the FCM algorithm will be also increased, which will be difficult to perform data analysis in real-time.

With the rapid development of the computer technology, data mining techniques arose and obtained quick development. As one of effective analysis tools,

* This work was supported by National Natural Science Foundation of China (No.60202004), the Key project of Chinese Ministry of Education (No.104173) and the program for New Century Excellent Talents in University of China.

clustering algorithm attracts more and more attention. In the field of data mining, the clustering algorithm is needed processing large amount of high-dimensional data set [3]. To this end, some fast clustering algorithms are presented, such as CURE, CLARA etc [4]. Those algorithms improve the efficiency by randomly selecting a small part of sample for cluster analysis, which cannot guarantee to obtain the optimal result.

For this purpose, a novel typical-sample-weighted clustering algorithm is proposed for large data sets. The new algorithm first pre-processes the data set with atom clustering and partitions the whole data set into a series of atom clusters. Then the atom centers, called nucleuses, are served as typical samples for further clustering. Finally the extracted typical samples are weighted by their typicality and then classified into different clusters with the traditional FCM algorithm. It has approximate linear time complexity and fits for analyzing large data sets.

2 Fuzzy c -Means Clustering Algorithm

Let $X = \{x_1, x_2, \dots, x_n\}$ be a set of n observation samples in feature space, in which $x_i = [x_{i1}, x_{i2}, \dots, x_{im}]^T$ denotes a feature vector of the sample x_i , and x_{ij} is the j -th attribute value of vector x_i . For a given integer $c(2 \leq c \leq n)$, the fuzzy c -means (FCM) clustering of the set X can be represented as the following mathematical programming problem.

$$\min \left\{ J_2(U, P) = \sum_{i=1}^c \sum_{j=1}^n (\mu_{ij})^2 \cdot d^2(x_j, p_j) \right\} \quad \text{s.t. } U \in M_{fc} \quad (1)$$

Where, $U = [\mu_{ij}]_{c \times n}$ is fuzzy partition matrix [5]. We have

$$M_{fc} = \left\{ U \in R^{c \times n} \mid \mu_{ij} \in [0, 1]; \sum_{i=1}^c \mu_{ij} = 1, \forall j; \quad 0 < \sum_{j=1}^n \mu_{ij} < n, \forall i \right\} \quad (2)$$

which is the fuzzy c -partition space of sample set X . $P = \{p_1, p_2, \dots, p_c\}$ denotes the clustering prototype set with c clustering center vectors. $d(\cdot)$ is defined as Euclidean distance for measuring the dissimilarity between sample and clustering prototype. For $X \subset R^m$, we have

$$d^2(x_j, p_i) = (x_j - p_i)^T \cdot (x_j - p_i) \quad (3)$$

In 1981, Bezdek presented the iterative formulas of FCM algorithm as follows.

$$\mu_{ij} = \left(\sum_{l=1}^c (d(x_j, p_l))^{-2} \right)^{-1} / (d(x_j, p_i))^2, \quad \forall i, j \quad (4)$$

$$p_i = \sum_{j=1}^n \mu_{ij}^2 x_j / \sum_{j=1}^n \mu_{ij}^2 \quad (5)$$

Bezdek pointed out that FCM algorithm has the tendency of partitioning the data set into clusters with approximate size. For the data set with different sized clusters, FCM algorithm cannot obtain the optimal partition, which leads to unsatisfied classification and inaccurate clustering prototypes. The main reason lies in that the traditional FCM algorithm does not consider the influence of the typicality of samples on the clustering result. For this purpose, a sample-weighted FCM algorithm is proposed in next section.

3 The Sample-Weighted FCM Algorithm

Corresponding to the FCM algorithm, the sample-weighted FCM (SWFCM) algorithm also represents the cluster analysis as the following mathematical programming problem.

$$\min \left\{ J_2(U, P) = \sum_{i=1}^c \sum_{j=1}^n w_j (\mu_{ij})^2 \cdot d^2(x_j, p_i) \right\} \quad s.t. U \in M_{fc} \quad (6)$$

where w_j denotes the weight for sample x_j with the constraint of $\sum_{j=1}^n w_j = 1$, and U is the fuzzy weighted partition matrix meeting the condition (2).

With the Lagrange multiplier method, the iterative formula can be derived for optimizing the objective function (6) of the SWFCM algorithm.

$$\mu_{ij} = \left(\sum_{l=1}^c (d(x_j, p_l))^{-2} \right)^{-1} / (d(x_j, p_i))^2, \quad \forall i, j \quad (7)$$

$$p_i = \sum_{j=1}^n w_j \mu_{ij}^2 x_j / \sum_{j=1}^n w_j \mu_{ij}^2 \quad (8)$$

Obviously, the weight w_j plays an important role for adjusting the clustering prototypes. In the case of $w_j = 1/n$, that is, the typicality of each sample is uniform on the clustering result, the SWFCM algorithm degrades to the traditional FCM algorithm.

In SWFCM algorithm, some priori information of data set is used to supervise the clustering process, which leads to satisfied classification result for some specified applications. We have applied the SWFCM algorithm to image segmentation with good results [6].

4 The Typical-Sample-Weighted FCM Algorithm for Large Data Sets

As an effective analysis tool of data mining, cluster analysis often encounter large data set. Like the traditional FCM algorithm, the SWFCM also involves the real-time problem for processing the large data set. For this end, a typical-sample-weighted FCM (TSWFCM) algorithm is proposed for fast clustering of large data set.

4.1 Atom Clustering Algorithm

To reduce the data amount, the atom clustering is first employed to pre-processing the large data set. Then the SWFCM algorithm is performed on the atom clusters for further clustering. In what follows, the atom clustering algorithm is briefly introduced.

Definition 1: For a given arbitrary point x in data space and a distance e , all the points within the circle region with center at x and radii of e form an atom with nucleus x . The other points within an atom is called as electrons around nucleus x .

The atom clustering algorithm first selects one of samples as nucleus, and then computes distance between the rest samples and the nucleus. If the distance is less than the threshold e , these samples are defined as electrons of the nucleus. Otherwise, the sample will be looked as a new nucleus. Till all the samples are processed, the atom clustering algorithm stops and outputs the atom clusters, $C = \{c_1, c_2, \dots, c_k\}$, where k is the total number of nucleus.

The atom clustering algorithm is a partition-based method, which classifies the n samples into some atom clusters. This algorithm only needs pre-setting a threshold, i.e., the radii of atom e , and performs once iteration for obtaining the data partition. The number of atoms will be much less than that of samples.

4.2 The TSWFCM Algorithm

The TSWFCM algorithm performs the SWFCM algorithm on the obtained atom clusters $C = \{c_1, c_2, \dots, c_k\}$. Each nucleus of atom cluster is looked as a typical sample, whose weight is determined by the number of corresponding electrons within this atom. Assume that M_j is the number of electrons around the j -th nucleus, then its weight w_j is defined as

$$w_j = \frac{M_j}{n} \quad (9)$$

where n is the total number of samples in data set, we have, $\sum M_j = n$. It is obvious that the more the number of electrons around a nucleus, the larger the typicality of the nucleus is.

4.3 The Sample Labeling Algorithm for Large Data Set

After performing the SWFCM algorithm on the atom clusters, the clustering prototypes and partition matrix of atoms can be obtained. To further partition each samples into clusters, a sample labeling algorithm is proposed based on the distance between each sample and the obtained clustering prototype. To avoid the wrong labeling resulted from the different size of clusters, the Mahalanobis distance is employed to measure the similarity between the samples and clustering prototypes.

Let $P = \{p_1, p_2, \dots, p_c\}$ denote the obtained c clustering prototypes, and \sum_i be the covariance of the the i -th cluster. The distance between nucleus c_j and prototype p_i is computed as follows.

$$d_M^2(x_j, p_i) = (x_j - p_i)^T \cdot (\sum)^{-1} \cdot (x_j - p_i) \quad (10)$$

If $d_M(x_j, p_l) = \min\{d_M(x_j, p_i), i = 1, 2, \dots, c\}$, then the nucleus x_j belong to the l -th cluster, and the electrons around x_j also belong to the l -th cluster.

As discussed above, the large data set is first reduced by the atom clustering algorithm. Then the SWFCM algorithm is used to cluster the obtain nucleuses. Finally, each sample is labeled and classified into clusters by computing the Mahalanobis distance between the nucleus and clustering prototypes. This is the kernel idea of the proposed TSWFCM algorithm.

5 Experimental Results

To verify the effectiveness of the proposed TSWFCM algorithm, some preliminary experiments are conducted for comparing the proposed algorithm and the traditional FCM algorithm. The experimental results illustrate the good performance of the new algorithm.

5.1 Experiments on the Data Set with Similar Sized Clusters

For the sake of visualization, we synthesize 9 Gaussian distributed subsets in 2D plane, each of which consists of 10000 samples, as shown in Fig.1 (a). Fig.1 (b) gives the result of atom clustering, in which each symbol "o" denotes a nucleus. Here we take $e = 0.002$. Since the samples are sparse at the boundaries of each cluster, the corresponding nucleuses have little electrons, which means that they have little typicality. For this purpose, by setting a threshold $T = 200$, the nucleuses with electrons less than T will be canceled. The processed nucleuses are shown in Fig.1 (c). In this way, the typical samples are reduced greatly.

Fig.1 (d) presents the number of electrons around each nucleus. It is obvious that the larger the amplitude, the more typicality the nucleus is. The clustering prototypes obtained by the TSWFCM algorithm are shown in Fig.1 (e). Fig.1 (f) shows the final clustering result of nucleuses, which agrees with our intuition.

5.2 Experiments on the Data Set with Different Sized Clusters

In many practical applications, the data set under analysis may contain some clusters with different size. The following experiment is used to verify the effectiveness of the proposed novel algorithm on such data set.

As shown in Fig.2 (a), the test data set includes 3 Gaussian distributed subsets with variance of 3, 1, and 1 respectively. As well known, the traditional FCM algorithm is sensitive to initialization, especially for such type of data set, which may get trap into the local optima and cannot obtain the global optimal classification. Fig.2 (b) shows the wrong clustering result of the traditional FCM algorithm.

Fig.2 (c) presents the result of the atom clustering, and Fig.2 (d) gives the typical nucleuses after thresholding with $T = 200$. The SWFCM algorithm is performed on the typical nucleuses, and the obtained clustering prototypes are

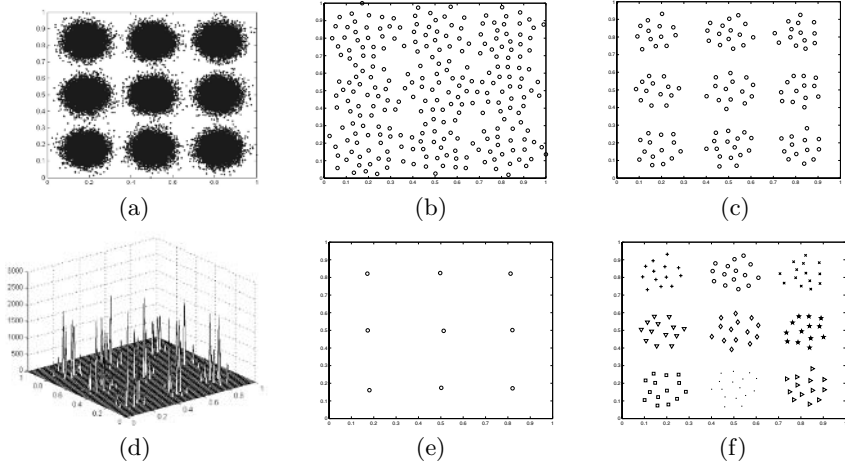


Fig. 1. Experimental results on the large data set with the similar sized clusters

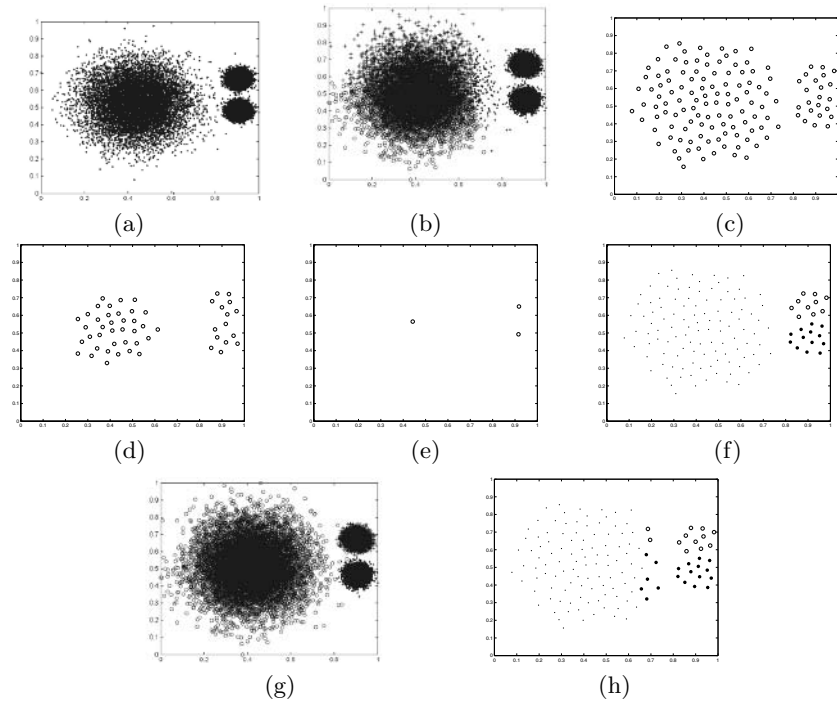


Fig. 2. Experimental results on the large data set with different sized clusters

shown in Fig.2 (e). The nuclei and the all the samples are classified into clusters with sample labeling algorithm, shown in Fig.2 (f) and (g) respectively. Fig.2 (h) presents the classification result of the nuclei of the traditional FCM algorithm, in which some homogeneous nuclei are classified into different clusters.

5.3 The Influence of Sample Weighting on the Clustering Prototypes

According the discussion above, it is known that the typical-sample-weighted FCM algorithm can not only reduce the computational complexity, but also adjust the clustering prototypes according to the distributed density and typicality of samples. This experiment is used to test the influence of sample weighting on the clustering prototypes.

Fig. 3 (a) presents a data set with two clusters, in which the centroid of the above cluster does not locate at the center of the cluster. Fig. 3 (b) presents the electrons distribution graph around each nucleus, which will be served as the weight of each nucleus for the TSWFCM algorithm. Fig. 3 (c) shows the obtained clustering prototypes by the traditional FCM algorithm and the TSWFCM algorithm respectively with comparing with the real centroids of clusters. It is found that the clustering prototypes obtained by the TSWFCM algorithm are much closer to the real centroids than those by the traditional FCM algorithm.

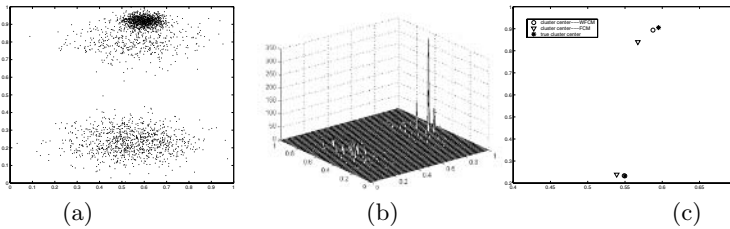


Fig. 3. Experimental result of the influence of sample weighting on the clustering prototypes

5.4 The Scalability of the Proposed Algorithm on Large Data Sets

To test the scalability of the proposed TSWFCM algorithm, we design two groups of experiments, one for sample number scalability and another for cluster number scalability. Both experiments are conducted on the PC with Intel Pentium 4 CPU (2.4GHz) and memory of 512MB. Fig. 4 (a) shows the variation

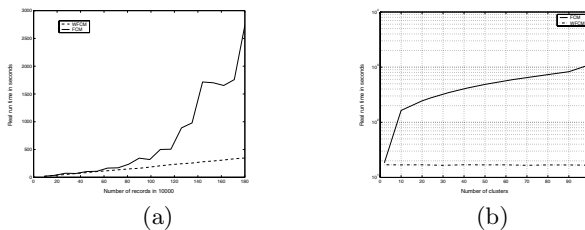


Fig. 4. The scalability comparison of FCM and TSWFCM algorithms. (a) The CPU time vs. data amount, (b) The CPU time vs. cluster number.

of the CPU time with the increasing of the data amount, in which the solid line and dashed line denote the variation curves of the FCM algorithm and the proposed TSWFCM algorithm with the data amount ranging from 90 thousands to 1.8 millions. Each curve is an average result of 10 independent experiments. Fig. 4 (b) presents the variation of the CPU time of the two algorithms with the increasing of the cluster number, in which the data set contains 1.8 millions of samples. From Fig.4, it can be found that the TSWFCM algorithm improves the efficiency of cluster analysis greatly by selecting the typical samples with atom clustering algorithm and reducing the data amount. So, it is suitable for clustering the large data sets.

6 Conclusions

This paper presents a typical-sample-weighted FCM clustering algorithm. The novel algorithm reduces the data amount greatly by selecting the typical samples with the atom clustering algorithm. The number of electrons or typicality is used to determine the weight of each nucleus, and then the sample-weighted FCM algorithm can cluster the nucleuses in real time. By weighting the typical sample, the TSWFCM algorithm can not only processes large data set, but also analyzes the data set with different sized clusters quickly and accurately.

References

1. He Qing: Advance of the theory and application of fuzzy clustering analysis. *Fuzzy System and Fuzzy Mathematics*. **12(2)** (1998) 89–94 (In Chinese)
2. Xinbo Gao: Optimization and Applications Research on Fuzzy Clustering Algorithms. Doctoral Thesis, Xidian University, Xi'an 710071, China. (1999)
3. Anderberg, M.R.: *Cluster Analysis for Applications*. Academic Press. (1973)
4. Kaufman, L., Rousseeuw, P. J.: *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons. (1990)
5. Everitt, B.: *Cluster Analysis*. New York, Heinemann Educational Books Ltd., (1974) 45–60
6. Xinbo Gao, Jie Li, Hongbing Ji: An automatic multi-threshold image segmentation algorithm based on weighting FCM and statistical test. *Acta Electronica Sinica*. **32(4)** (2004) 661–664

Mining Weighted Generalized Fuzzy Association Rules with Fuzzy Taxonomies

Shen Bin, Yao Min, and Yuan Bo

College of Computer, Zhejiang University, Hangzhou 310027, China
{tsingbin, myao}@zju.edu.cn

Abstract. This paper proposes the problem of mining weighted generalized fuzzy association rules with fuzzy taxonomies (WGF-ARs). It is an extension of the generalized fuzzy association rules with fuzzy taxonomies problem. In order to reflect the importance of different items, the notion of generalized weights is introduced, and leaf-node items and ancestor items are assigned generalized weights in our WGF-ARs. The definitions of weighted support and weighted confidence of WGF-ARs is also proposed. Then a new mining algorithm for WGF-ARs is also proposed, and several optimizations have been applied to reduce the computational complexity of the algorithm.

1 Introduction

The problem of association rules (AR) can be expressed as $X \Rightarrow Y$, which means that a data record that contains the set of items X is likely to contain items Y as well. If X or Y in AR is a set of fuzzy sets, then we call this kind of association rules fuzzy ARs. Recently, various efforts have been made to develop and improve the theory or applications of fuzzy ARs. For example, Guoqing Chen^[1] introduces two kind of fuzzy extensions to classical ARs. One extension is that, in generalized association rules, the taxonomies concerned is fuzzy. The other extension is that managers are likely to use fuzzy linguistic expressions when referring to decision rules. In [3] authors focus on positive and negative fuzzy association rules mining. M. De Cock et al.^[4] discuss the true semantics of support and confidence in fuzzy ARs. Fuzzy rules also can be obtained in quantitative rules by replacing intervals by fuzzy sets^[2].

In this paper, based on fuzzy taxonomies of reference [1], we try to give weights to the items of fuzzy association rules with fuzzy taxonomies, and propose the corresponding concepts, models and algorithm about weighted generalized fuzzy association rules with fuzzy taxonomies (WGF-AR for short).

2 Fuzzy Association Rules with Fuzzy Taxonomies^[1]

Reference [1] extends the classical generalized association rules from two different aspects: fuzzy taxonomies and fuzzy terms of linguistic expressions. We list them in figure 1 and figure 2 respectively. In figure 1, an example of a fuzzy taxonomic structure is shown, where child-node can belong to its parent-node with a certain degree in $(0, 1)$. Figure 2 shows the new fuzzy taxonomic structures which incorporates

linguistic hedges on existing fuzzy taxonomies. It is necessary to point out that only leaf-node items are listed in the real transaction set.

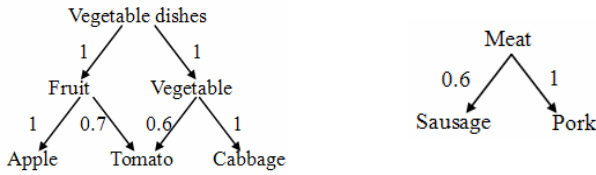


Fig. 1. Example of fuzzy taxonomies in generalized association rules

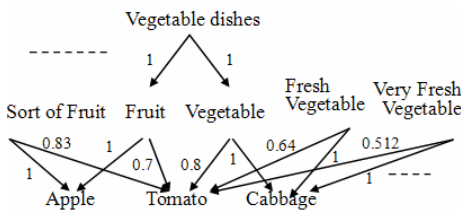


Fig. 2. Example of linguistically modified fuzzy taxonomies

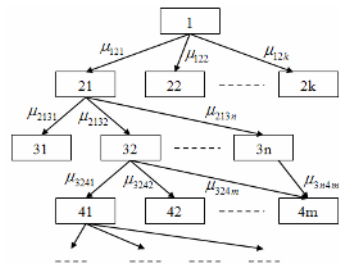


Fig. 3. An abstract fuzzy taxonomy

Assume there are transaction database D , and transaction set T , we can abstract the fuzzy taxonomies of Figure 1, 2 into the fuzzy taxonomic structure shown in Figure 3. In order to describe the degree that node y belongs to its ancestor node x , notion μ_{xy} is introduced, which can be derived by the following formula:

$$\mu_{xy} = \bigoplus_{\forall l: x \rightarrow y} (\bigotimes_{\forall e \text{ on } l} \mu_{le}) \tag{1}$$

Where $l: x \rightarrow y$ is one of the accesses from node x to y in the abstract fuzzy taxonomy, e on l is one of the edges on access l , μ_{le} is the fuzzy degree on the edge e on l . If there is no path between x and y , then $\mu_{xy} = 0$. Operators \oplus and \otimes depends on the content of the problems at hand and can be the appropriate operations. For example, \oplus and \otimes can be max and min operations respectively.

Then the computation methods of support value and confidence value can be further studied. Readers can refer to reference [1] for detail information.

3 Mining WGF-AR

3.1 WGF-AR

We consider the importance of different items for generalized fuzzy association rules, that is to say weight factor is taken into account within the problem of generalized fuzzy association rules with fuzzy taxonomies. Given a set of leaf-nodes of fuzzy

taxonomies $\{y_1, y_2, \dots, y_m\}$, we assign a weight W_{y_j} for each item y_j where $j = \{1, 2, \dots, m\}$, to show the importance or interest of the leaf-node. Thus for every item, that appears in transaction set, has been given its corresponding weight.

Further, we should assign the weights to the ancestors of the leaf-nodes, which are not appear in transaction set, but can be generalized from the low-level leaf-node items. Because the high-level ancestor-node can be generalized from different leaf-nodes, and at the same time different leaf-nodes always have different weights, we argue that the same ancestor-node, that are generalized by different leaf-nodes, should be assigned different weights. So we use the concept of generalized weight to depict the weight of same ancestor which can be generalized by different leaf-nodes.

Definition 1. Generalized weight W_{xy} expresses the importance or interest of ancestor item x , which is generalized from leaf-node item y .

Assume there are m leaf-node items y_1, y_2, \dots, y_m , and their corresponding weights $W_{y_1}, W_{y_2}, \dots, W_{y_m}$, firstly we compute the degree of belonging μ_{xy_j} between leaf-node item y_j and ancestor item x according to formula 1, where $j = \{1, 2, \dots, m\}$. Then generalized weight W_{xy_j} ($j = 1, \dots, m$) can be obtained from the following formula:

$$W_{xy_j} = \mu_{xy_j} \cdot W_{y_j} \tag{6}$$

It is obvious that for leaf-node item y , its generalized weight and weight are equal.

After assigning the weights for all nodes, that include leaf-nodes and ancestor of leaf-nodes, in an analogous manner, we can give the computation methods for support and confidence of weighted generalized fuzzy association rules:

Let T be transaction set, $t \in T$ be a transaction of T , and let a be a item in transaction t . X is a item set, and $x \in X$ is an item in X . Thus, W_{xa} can be viewed as the weighted support measure that item set $\{a\}$ support x . Further, we can obtain the weighted support measure that t support X as following:

$$W_{tX} = Support_{tX} = \min_{x \in X} (\max_{a \in t} (W_{xa})) \tag{7}$$

Analogously, consider the whole transaction set T , the weighted support degree that T support X can be computed as following:

$$Support(X) = \sum_{t \in T} Support_{tX} / |T| = \sum_{t \in T} W_{tX} / |T| \tag{8}$$

Definition 2. A weighted generalized fuzzy association rule with fuzzy taxonomies has the form of $X \Rightarrow Y$, where X, Y are item sets, $X \cap Y = \emptyset$, and there should be no two items which have the relationship of ancestor and child in item set $X \cup Y$.

For a WGF-AR $X \Rightarrow Y$, let $a, b \in X \cup Y$, we constrain that a should not be the ancestor of b or the child of b in definition 2, thus we are able to filter the meaningless WGF-ARs for the mining process.

Furthermore, we can define the weighted support and weighted confidence to our WGF-ARs as definition 3 and definition 4.

Definition 3. Let a WGF-AR be $X \Rightarrow Y$, and $X \cup Y = Z$, then the weighted support of WGF-AR $X \Rightarrow Y$ can be computed as follows:

$$Support(X \Rightarrow Y) = \sum_{i \in T} W_{iz} / |T| \tag{9}$$

Definition 4. For a WGF-AR $X \Rightarrow Y$, let $X \cup Y = Z$, then the weighted confidence of $X \Rightarrow Y$ can be obtained as follows:

$$Confidence(X \Rightarrow Y) = \sum_{i \in T} W_{iz} / \sum_{i \in T} W_{ix} \tag{10}$$

Definition 5. For WGF-ARs, a k – itemset X is called a frequent itemset if X satisfy the following two conditions:

- (1) There should be no two items which have the relationship of ancestor and child in item set X ;
- (2) The weighted support of X is greater than or equal to the minimum weighted support threshold $w_{min\ sup}$:

$$Support(X) = \sum_{i \in T} Support_{ix} / |T| = \sum_{i \in T} W_{ix} / |T| \geq w_{min\ sup} \tag{11}$$

Condition (1) is used to avoid the production of meaningless WGF-ARs, and condition (2) is applied in order to satisfy the frequent characteristic.

Definition 6. An WGF-AR $X \Rightarrow Y$ is called an interesting rule if it is satisfied that $X \cup Y$ is a frequent itemset, and its weighted confidence is greater than or equal to the minimum weighted confidence threshold $w_{min\ conf}$.

3.2 Mining WGF-AR: An Illustrative Example

We use the example in reference [1] to illustrate the concepts, computation methods of our WGF-ARs.

Suppose in a retailing store, a database is shown in Table 1 and 2. Table 1 shows the database of transaction. Table 2 shows the information about the items in the retailing store, such as the bar code number, the names, the total profits and the given weights of the items. The underlying fuzzy taxonomies of the items are shown in Figure 1.

Table 1. Transactions in a retailing store

Table 2. Product information database

Transaction No.	Purchased Products
#100	Apple
#200	Tomato, Sausage
#300	Cabbage, Sausage
#400	Tomato, Pork
#500	Pork
#600	Cabbage, Pork

Bar Code Number	Product Name	Total Profit	Weight
1	Apple	100	0.1
2	Tomato	100	0.1
3	Cabbage	200	0.2
4	Sausage	300	0.3
5	Pork	600	0.6

Then according to formula (1), we can obtain those leaf-nodes and their ancestor's degree as shown in Table 3. For example: $\mu_{Fruit, Tomato} = \max(\min(0.7)) = 0.7$.

Table 3. Leaf-nodes and their ancestor's degrees ^[1]

Leaf-node items	The degrees of the ancestors and its own
Apple	1/Apple, 1/Fruit, 1/Vegetable dishes
Tomato	1/Tomato, 0.6/Vegetable, 0.7/Fruit, 0.7/vegetable dishes
Cabbage	1/Cabbage, 1/Vegetable, 1/Vegetable dishes
Pork	1/Pork, 1/Meat
Sausage	1/Sausage, 0.6/Meat

Furthermore, according to the formula (6) and Table 2, the generalized weights of those leaf-nodes and their ancestor are listed in Table 4. For instance, $W_{Fruit, Tomato} = \mu_{Fruit, Tomato} \cdot W_{Tomato} = 0.7 \times 0.1 = 0.07$.

Table 4. Leaf-nodes and their ancestor's generalized weights

Leaf-node items	The generalized weights of the ancestors and their own
Apple	0.1/Apple, 0.1/Fruit, 0.1/Vegetable dishes
Tomato	0.1/Tomato, 0.06/Vegetable, 0.07/Fruit, 0.07/vegetable dishes
Cabbage	0.2/Cabbage, 0.2/Vegetable, 0.2/Vegetable dishes
Pork	0.6/Pork, 0.6/Meat
Sausage	0.3/Sausage, 0.18/Meat

If the value of $w \min \sup$ is 0.5/6, then according to definition 5 and formula (8), all the frequent itemsets are listed in Table 5 along with their corresponding support value. For instance, for itemset {Vegetable, Meat}, its support is $(\min(0.06, 0.18) + \min(0.2, 0.18) + \min(0.06, 0.6) + \min(0.2, 0.6)) / 6 = 0.5 / 6$. For itemset {Pork, Meat}, it satisfies the condition of minimal weighted support threshold, but fails to meet the first condition of definition 5. So it is not listed in frequent itemset table.

Table 5. Frequent itemsets and their corresponding support values

Frequent itemsets	Support values
{Pork}	1.8/6
{Sausage}	0.6/6
{Vegetable dishes}	0.64/6
{Vegetable}	0.52/6
{Meat}	2.16/6
{Vegetable dishes, Meat}	0.52/6
{Vegetable, Meat}	0.5/6

Table 6. The discovered WGF-ARs

Discovered rules	Weighted support value	Weighted confidence value
Vegetable dishes \Rightarrow Meat	0.52/6	0.813
Vegetable \Rightarrow Meat	0.5/6	0.962

If the value of $w_{minconf}$ is 0.8, then according to the frequent itemsets, those WGF-ARs are listed in Table 6. For instance, $Support (Vegetable \Rightarrow Meat) = Support (Vegetable, Meat) = 0.52/6$, $Confidence (Vegetable \Rightarrow Meat) = Support (Vegetable, Meat) / Support(Vegetable) = (0.52/6) / (0.52/6) = 0.962$.

3.3 The Mining Algorithm for WGF-ARs

The most direct way of solving generalized association rules mining is that all the ancestors of each leaf item in fuzzy taxonomies are added into the transaction set T to form an extended transaction set T' [1,5]. And then a large number of traditional association rules mining algorithm such as Apriori algorithm can be applied to T' directly. Similarly to Guoqing Chen's approach [1], T' is generated by not only adding to T all the ancestors of each leaf-node item in fuzzy taxonomies, but also the weighted support degrees that the transactions support these ancestors.

Referring to the algorithm in reference [1], we decompose the whole WGF-AR algorithm into a series of sub-algorithms, and give their corresponding pseudocodes.

Firstly, we should determine the belonging measures that leaf-nodes belongs to their ancestors according to formula (1). Then we should use formula (6) to obtain the generalized weights of ancestors that are generalized by various leaf-node items. Concretely, we have the following Degree & Weight sub-algorithm:

```

Sub-algorithm Degree & Weight:
forall leaf nodes  $LN_i \in Tax$  do
    forall interior nodes  $IN_j \in Tax$  do
         $\mu(IN_j, LN_i) = \max_{\forall l: IN_j \rightarrow LN_i} (\min_{\forall e \in l} (\mu_{le}))$ 
        Insert into Degree,
        Values  $IN_j, LN_i, \mu(IN_j, LN_i), W(IN_j, LN_i) = \mu(IN_j, LN_i) \cdot W(LN_i)$ 
        Insert into Weight, Values  $IN_j, LN_i, W(IN_j, LN_i)$ 
    endfor
endfor
    
```

Consequently, we need to produce the extended transaction set T' by adding the corresponding ancestors and their generalized weights to transaction set T . The pseudocode of this sub-algorithm is shown in following:

```

Sub-algorithm New Extended Transaction Set  $T'$ :
forall  $t \in T$  do
    Insert into  $T'$ , Values all the elements  $y_i \in t$  with weight  $W_{y_i}$ 
    forall the ancestor  $x_j$  of elements  $\in t$  do
         $W_{tx_j} = Support_{tx_j} = \min_{x=x_j} (\max_{o \in t} (W_{xo}))$ 
        Insert into  $T'$ , Values  $x_j$  with the support value  $W_{tx_j}$ 
    endfor
endfor
    
```

After the extended transaction set T' is produced, we then can generate the candidate frequent itemsets. We extend the classical Apriori algorithm, and have the extended Apriori sub-algorithm as follows which is similar to reference [1]'s extended

Apriori sub-algorithm, where C_k is the set of all candidate k – itemsets, and L_k is the set of all frequent k – itemsets:

```

Sub-algorithm Extended Apriori:
 $L_1 = \{ \text{frequent 1-itemsets} \}$ 
for {  $k = 2; L_{k-1} \neq \emptyset; k ++$  } do
     $C_k = \text{Apriori-Gen}(L_{k-1}) // \text{Generating all candidate } k\text{-itemsets from } L_{k-1}$ 
    forall transactions  $t \in T'$  do
         $C_t = \text{subset}(C_k, t) // \text{Generating all candidate } k\text{-itemsets that are contained in } t$ 
        forall candidates  $c \in C_t$  do  $c.\text{Support} = c.\text{Support} + W_c$  endfor
    endfor
     $L_k = \{ c \in C_k \mid c.\text{Support} \geq w \text{ min sup} \}$ 
endfor
Answer =  $\bigcup_k L_k$ 

```

Here Apriori-Gen function has the input parameter of frequent $k-1$ -itemset L_{k-1} , and then return all of the candidate frequent itemsets. It contains two processes:

(1) **Filtering and Join Step:** In order to obey the condition (1) of definition 5, we need to make the corresponding decision whether they should be joined or not in this step. Assume that items in each itemset are kept sorted in lexicographic order, we have the following pseudocodes:

```

insert into  $C_k$ 
select  $p.\text{item}_1, p.\text{item}_2, \dots, p.\text{item}_{k-1}, q.\text{item}_{k-1}$  from  $L_{k-1} p, L_{k-1} q$ 
where  $p.\text{item}_1 = q.\text{item}_1, \dots, p.\text{item}_{k-2} = q.\text{item}_{k-2}, p.\text{item}_{k-1} < q.\text{item}_{k-1}$ ,  $p.\text{item}_{k-1}$  is not
the ancestor or child of  $q.\text{item}_{k-1}$ 

```

If we consider fuzzy taxonomies with linguistic hedges that are shown in figure 2, two kinds of optimizations could be taken. One optimization is that we should eliminate the itemsets which contain a pair of child item and its corresponding ancestor item together. The other optimization is that we also should remove the itemsets that contain a pair of same class items. The same class items are the original item and linguistic modified items based on this original item in fuzzy taxonomies [1]. For example, in Figure 2, “Vegetable”, “Fresh Vegetable”, “Very Fresh Vegetable” are in the same class. For this kind of fuzzy taxonomies with linguistic hedges, this filtering and join step is listed as following:

```

insert into  $C_k$ 
select  $p.\text{item}_1, p.\text{item}_2, \dots, p.\text{item}_{k-1}, q.\text{item}_{k-1}$  from  $L_{k-1} p, L_{k-1} q$ 
where  $p.\text{item}_1 = q.\text{item}_1, \dots, p.\text{item}_{k-2} = q.\text{item}_{k-2}, p.\text{item}_{k-1} < q.\text{item}_{k-1}$ ,  $p.\text{item}_{k-1}$  is not the
ancestor or child of  $q.\text{item}_{k-1}$  and  $\text{same\_class}(p.\text{item}_{k-1}, q.\text{item}_{k-1}) = 0$ 

```

(2) **Pruning Step:** In this step, the itemset $c \in C_k$ will be pruned, if a $(k-1)$ -subset of the candidate itemset c does not exist in frequent $k-1$ -itemsets L_{k-1} :

```

forall candidates  $c \in C_k$  do
  forall  $(k-1)$ -itemsets  $d \subseteq c$  do
    if  $d \notin L_{k-1}$  then delete  $C$  from  $C_k$  endif
  endfor
endfor

```

Thus, the corresponding WGF-ARs will be generated based on such frequent itemsets. The extended Rules Generation sub-algorithm is shown as following:

```

Sub-algorithm Rules Generation:
forall frequent itemsets  $L_k, k > 1$  do call gen-rules  $(L_k, L_k)$  endfor
procedure gen-rules( $L_k$ : frequent  $k$ -itemset,  $a_m$ : frequent  $m$ -itemset)
   $A = \{ (m-1)\text{-itemsets } a_{m-1} \mid a_{m-1} \subset a_m \}$ 
  forall  $a_{m-1} \in A$  do
     $Conf = \sum_{l \in T} W_{l_k} / \sum_{l \in T} W_{a_{m-1}}$ 
    if  $(Conf > wminconf)$  then
      output the rule  $a_{m-1} \Rightarrow (l_k - a_{m-1})$ , with  $Confidence = Conf$  and  $Support = l_k.Support$ 
      if  $((m-1) > 1)$  then call gen-rules  $(l_k, a_{m-1})$  endif
    endif
  endfor
endprocedure

```

The WGF-ARs that is generated by the above method also can be filtered by the measure of interesting, and produce the filtered WGF-ARs.

4 Conclusions

In this paper, we have proposed the mining problem of weighted generalized fuzzy association rules with fuzzy taxonomies (WGF-ARs). This is an extension of the generalized fuzzy association rules with fuzzy taxonomies problem. In our WGF-ARs, the leaf-node items and ancestor items are assigned weights to reflect their correlative importance to the user. The main difference between mining WGF-ARs and unweighted generalized fuzzy association rules with fuzzy taxonomies is that generalized weights are taken into consideration in WGF-ARs.

We propose the definitions of generalized weights, the weighted support and the weighted confidence of WGF-ARs. A new mining algorithm for WGF-ARs is also proposed, and several optimizations have been applied to reduce the computational complexity of the algorithm.

Acknowledgement. We would like to thank the financial support of the Specialized Research Fund for the Doctoral Program of Higher Education (SRFDP) (20040335129) and Key Program of Natural Science Foundation of Zhejiang Province (No. Z104267).

References

1. Guoqing Chen, Qiang Wei. Fuzzy association rules and the extended mining algorithms. *Information Sciences* 147(2002): 201-228.
2. Cubero, J.C.; Medina, J. M.; Pons, O.; Vila, M.A., 1995. Rules discovery in fuzzy relational databases. In conference of the North American Fuzzy Information Processing Society, NAFIPS'95. pp. 414-419.
3. Peng Yan, Guoqing Chen, Chris Cornelis, Martine De Cock, Etienne Kerre. Mining positive and negative fuzzy association rules. In: *Lecture Notes in Computer Science* 3213 (M. G. Negoita, R. J. Howlett, L. C. Jain, eds.), Springer-Verlag, 2004, p. 270-276.
4. M. De Cock, C. Cornelis, E. E. Kerre. Elicitation of fuzzy association rules from positive and negative examples.
5. R. Srikant, R. Agrawal, Mining generalized association rules. *Proceedings of the 21st VLDB conference*, Zurich, Switzerland, 1995.
6. C.H. Cai, Ada W.C. Fu, C.H. Cheng and W.W. Kwong. Mining association rules with weighted items. *IEEE International Database Engineering and Applications Symposium*, Cardiff, 1998.

Concept Chain Based Text Clustering

Shaoxu Song, Jian Zhang, and Chunping Li

School of Software, Tsinghua University, 100084 Beijing, China
{song-sx03, zjian03}@mails.tsinghua.edu.cn
cli@tsinghua.edu.cn

Abstract. Different from familiar clustering objects, text documents have sparse data spaces. A common way of representing a document is as a bag of its component words, but the semantic relations between words are ignored. In this paper, we propose a novel document representation approach to strengthen the discriminative feature of document objects. We replace terms of documents with concepts in WordNet and construct a model named Concept CHain Model(CCHM) for document representation. CCHM is applied in both partitioning and agglomerative clustering analysis. Hierarchical clustering processes in different levels of concept chains. The experimental evaluation on textual data sets demonstrates the validity and efficiency of CCHM. The results of experiments with concept show the superiority of our approach in hierarchical clustering.

1 Introduction

Text clustering is a process of grouping a set of text objects into classes with high inner proximity. For text mining, data objects are always semi-structured or unstructured data, like text and hypertext documents. Current existing document representation systems are typically the bag-of-words model [1], where single word and word stem are used as features for representing document content. Methods, such as TFxIDF [2], try to strengthen the discriminative features of objects when describing the correlations between them.

In the specific case of textual data, we should consider the discriminative feature of document objects in clustering. Current existing approaches (including TFxIDF) can hardly deal with the special textual relation between words, such as synonymy and hypernymy-hyponymy between nouns. For example, *penury* and *poverty* are mapped into different features and there is no way to generalize similar terms like *basketball* and *football* to their common hypernymy *sport*. However, these semantic relations can reduce the dimension of documents and improve the accuracy of text clustering.

In this paper, concept chain in WordNet [3] is used to describe the document features. The relations between concepts are considered by using the WordNet hyper graph structure. We propose an approach named Concept CHain Model(CCHM) for document representation. Based on this model, several text clustering approaches are applied in our work. Furthermore, we mainly focus on how the concept chain knowledge supervises the agglomerative hierarchical clustering process.

This paper is structured as follows. In Section 2 we briefly introduce the relative knowledge of WordNet and propose the construction of CCHM. In Section 3 we apply our concept model to both partitioning and agglomerative clustering approaches. Section 4 reports an experimental evaluation on real textual data and shows the quality of our approach. Finally, Section 5 is a summary.

2 Concept Representation

In order to use concept knowledge in text clustering, we need to transform documents to numeric vectors. Documents are represented by *vector space model*, in which each text document d is represented by a vector of weights of p terms:

$$d_i = (w_{i1}, \dots, w_{ip}) \quad (1)$$

where d_i is vector of document i , w_{ik} is weight of term k in document i .

2.1 WordNet

The English WordNet is an online lexical reference system whose design is inspired by current psycho-linguistic theories of human lexical memory. Let us see some important items of WordNet first.

Unique beginner: In WordNet, unique beginner is a noun synset with no hypernymy synsets.

Concept chain: Concept chain is a structure $\zeta := (C, <)$ consisting of a set C , whose elements are concepts. These concepts are related to each other through hypernymy-hyponymy relation.

Lexicographer id: A decimal integer uniquely identifies a sense (synset).

In WordNet, there are 25 unique beginners for nouns, which are related to other noun synsets through hyponymy relation. And other noun synsets are related to each other through hypernymy-hyponymy (generalization-specialization) and holonymy-meronymy (whole of-part of). A noun synset represents an underlying concept, and the last element of a concept chain is a unique beginner.

2.2 Concept Chains

In our work, we mainly consider the hypernymy-hyponymy relation between noun synsets. This kind of semantic relation can help us extract more generalized information of documents. And we don't initially retain only one sense per word in WordNet. We consider all concepts (senses) of concept chains which are related to terms in documents and built by using WordNet.

Fig. 1 shows an example of concept chains. The small triangle represents word "football". Every small circle is a concept which represents a certain synset in WordNet. The comment near the small circle is the content of synset. The number in the comment is the lexicographer index of the synset. Two senses of the word "football" are described with two concept chains respectively, whose starting nodes are represented with the two small circles. The two

circles, which represent ending nodes of concept chains, represent two unique beginners of WordNet. For example, concept chain ch_1 can be described as follows: (3255461) < (2681909) < (3289024) < (3174243) < (3443493) < (19244) < (2645) < (16236) < (1740). (3255461) represents a sense of word “football”, as the starting node in the bottom of the chain. (1740) is a unique beginner, as the ending node in the top of the chain. High level concept “game equipment” (3289024) is a generalization of low concept “ball” (2681909).

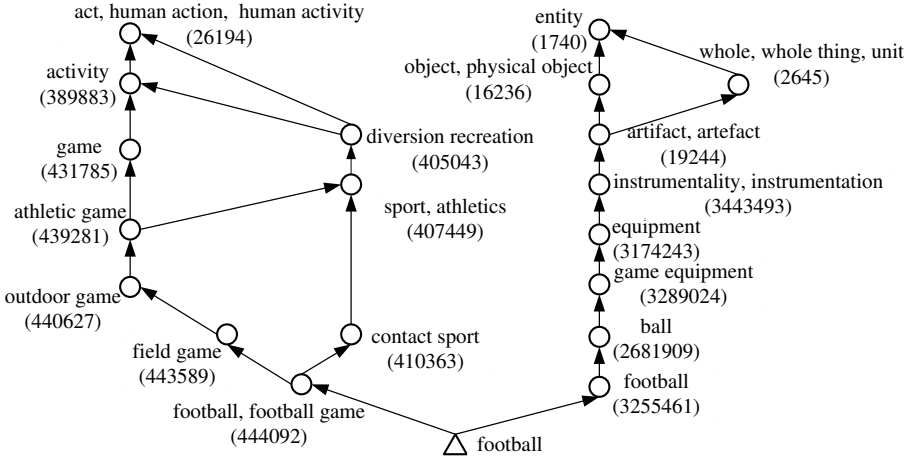


Fig. 1. An example of concept chain

2.3 Concept Weights

The weight of a term in a vector is computed using a form of TF*IWF*IWF weighting developed by Roberto Basili [4], outperforms TFxIDF. So we adopt equation (2) to calculate term vectors.

$$w_d(t) = n(t, d) \times \left[\log_e \left(\frac{|D|}{n(t)} \right) \right]^2 \tag{2}$$

where $n(t, d)$ is the number of occurrence of term t in document d , and $|D|$ is the number of documents used in computing the inverse document frequency weights. $n(t)$ is the number of documents in which term t occurs.

The weight of a concept, which occurs in the concept chain of term t , is computed as follows

$$w_d(c) = w_d(t) \times \beta \times \gamma^k \tag{3}$$

where $w_d(t)$ is the weight of word t in document d . $w_d(c)$ is the weight of concept c which is one node of the concept chain to describe term t . Input parameter γ is a decline factor in the range of $0 < \gamma < 1$, which is used to limit the weight of concept and distinguish the levels of concept chain. k is the level position of

concept c in the chain. β is standardization factor which is used to adjust the weight of concept c .

$$\beta = \varepsilon \left/ \sum_{i=1}^m \sum_{j=1}^{n_i} \gamma^j \right. \tag{4}$$

where ε is a concept effect factor, which means the weight of all concept chains added by term t . Here, we set $\varepsilon = m$. m is the number of concept chains described all senses of term t . n_i is the number of nodes in concept chain ch_i . And at the same time, ε can guarantee the weights of concepts large enough to supervise clustering process.

By looking up WordNet, we can use several concept chains to describe term t . For example, in Fig. 1, two concept chains describe word “football”, then $m = 2$. And the right concept chain ch_1 contains 8 nodes. So, $n_1 = 8$. (Here, we see (16236) and (2645) as two parallel nodes.) Concept c_1 (3255461) is start node of ch_1 and c_3 (3289024) is the third node of ch_1 . If the weight of c_1 is w_1 , the weight of c_3 will be $w_1 * \gamma^2$.

3 Clustering Analysis

In this section, we apply the Concept CHain Model(CCHM) in both partitioning and agglomerative hierarchical clustering analysis. In order to exert the hierarchical structure of concept chain, we integrate the CCHM into an agglomerative hierarchical clustering algorithm, named TCUAP [5].

3.1 Concept Based Similarity Measure

The similarity between documents can be quantified by the correlation, ϕ_{ij} , which is so-called the *Cosine Measure*:

$$\phi_{ij} = \sum_{k=1}^p w_{ik}w_{jk} \left/ \left(\sum_{k=1}^p w_{ik}^2 \sum_{k=1}^p w_{jk}^2 \right)^{1/2} \right. \tag{5}$$

where ϕ_{ij} is the cosine similarity value between documents i and j , w_{ik} means the weight value of concept k in document i . Clearly, a measure based on cosine similarity can be used when all concepts are measured on the same scale.

In TCUAP, the definition of asymmetric similarity is needed. Consider that the common concepts of two documents d_i and d_j are s . If the weight of s in d_j is large, then the asymmetric similarity from d_i to d_j is high; and if the weight of s in d_i is small, then the asymmetric similarity from d_j to d_i is low. Specifically,

$$\vec{\phi}(d_i, d_j) = \sum_{k=1}^p |w_{ik} - |w_{ik} - w_{jk}|| \left/ \sum_{k=1}^p w_{jk} \right. \tag{6}$$

where $\vec{\phi}(d_i, d_j)$ is the direct similarity value from document d_i to d_j , w_{ik} is the weight value of concept k in document d_i . Clearly, the similarity between documents is asymmetric (i.e., $\vec{\phi}(d_i, d_j) \neq \vec{\phi}(d_j, d_i)$).

3.2 Clustering Algorithms

For partitioning clustering algorithm, we choose K-means which concept similarity can be directly applied to. Also we use the TCUAP as an agglomerative hierarchical clustering algorithm. The clustering analysis work of TCUAP is based on the strong components partition algorithm. The strong components of sparse matrix are dense, and the similarity between dense objects are high [6]. This partition analysis satisfies the principle of clustering that high inner proximity and low exterior proximity. Furthermore, The strong components method merges several objects each time, whereas the classical agglomerative algorithms only merge the two most similar objects in each iteration. So the performance of TCUAP with strong components merger strategy is high.

3.3 Concept Levels in Hierarchical Clustering

Concepts in different levels of the chain have distinct effect in hierarchical clustering. Top concepts of the chains denote nonobjective topics, while the bottom ones mean specific topics. Documents about different specific topics may have common high level concepts, which help us to find semantic correlations between documents. For example, documents about *basketball* and documents about *football* can be associated together with the common top concepts *sports*. Note that the weights of high level concepts are smaller after taking IDF process in CCHM.

According to the structure feature of concept chain, hierarchical clustering is able to process in different levels of concepts. In the early clustering iterations of TCUAP, those concepts in the bottom of the chain are especially concerned because of their larger weight values than top ones. Documents with common specific concepts are classified in the same cluster. In the subsequent process, the weak correlations of high level concepts between clusters are considered. It is the reason why we use IDF of concept in CCHM. Fig. 2 shows a simple example of the clustering process. Gray points mean documents, and arrowheads denote asymmetric similarity between documents.

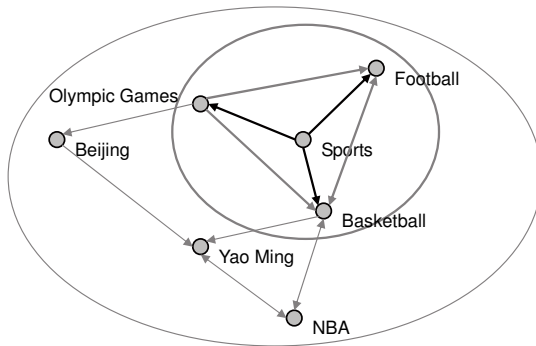


Fig. 2. An example of TUCAP clustering

4 Experiment and Evaluation

4.1 Data Sets

20Newsgroups. The 20 Newsgroups¹ (20Ng) data sets, collected by Lang [7], contain about 20,000 articles. Each newsgroup represents one class in the classification task. Each article is designated to one or more semantic categories and the total number of categories is 20. We chose 2 topic categories with 1000 documents in our experiments, i.e. “comp.graphics”, “comp.sys.ibm.pc.hardware”.

4.2 Evaluation Criteria

We evaluate our approach by the classification accuracy, which has also been used in [8]. Our classification accuracy is very similar to Kohonen’s, but we count the correct documents instead of errors. The article is correct, if one of the original labels assigned by data set matches the cluster label. The accuracy is calculated by the proportion of the number of correct articles to the number of all input news articles.

4.3 Evaluation

Partitioning Clustering Approaches. This experiment was focused on evaluating the quality of CCHM in partitioning clustering algorithm. As a comparison on classical algorithm, we choose K-means. Initial centroids are selected randomly, so we run the program 20 times and show the best 5 results.

The comparison of partition algorithms is given in Table 1. 1000 documents of *20Newsgroups* are classified into 2 clusters. The decline factor of concepts is set to be $\gamma = 0.6$. From the table we can learn that the effects of concepts in K-means are not so evident.

Table 1. Comparison of the accuracy among K-means

	1	2	3	4	5	Average
K-means with concept	0.664	0.628	0.627	0.602	0.599	0.624
K-means without concept	0.654	0.644	0.602	0.594	0.591	0.617

Agglomerative Hierarchical Clustering Approaches. We also compare agglomerative hierarchical clustering algorithms with and without concepts. Also, 1000 documents of *20Newsgroups* are classified. The results are given in Fig. 3. In this experiment, the decline factor γ of concept chain is set to be $\gamma = 0.6$. As shown in the figure, TCUAP with concept achieves a higher performance when dealing with smaller number of end clusters. Whereas the accuracy of TCUAP without concept drops quickly in the small number of end clusters, especially on

¹ The 20 newsgroups collection can be achieved at: <http://kdd.ics.uci.edu/>.

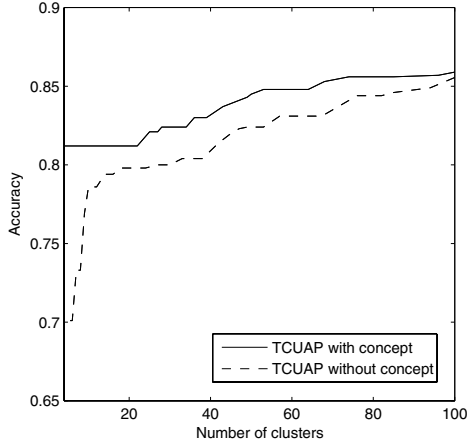


Fig. 3. Accuracy in different number of clusters

the final results less than 20 clusters. This observation illustrates the validity of high level concepts in the last clustering process. However, from the observation of the early clustering process (with the big number of clusters), we can find that the performances of these two kinds of TCUAP are quite similar. Because of the influence of concepts, the TCUAP with concepts performs even worse in the early steps.

Finally, the low weight concepts in the high level help to find weak semantic correlations between clusters. However, the influence of concepts should be reduced in the early process of TCUAP, and the decline factor of concept chain plays a huge role in this aspect. In Table 2, we compare the results of different values of decline factor γ . From the table, we can learn that factor $\gamma = 0.6$ achieves the best result in most cases of end clusters. If the decline value is too large, the concepts can hardly supervise the clustering and the performance is poor. On the other hand, if the decline value is too small, terms with more hypernymy concepts may influence the original features of documents. As shown in the table, the accuracy of $\gamma = 0.5$ is lower.

Table 2. Comparison of the accuracy among different decline factors γ

Number of clusters	Concept decline factor γ					Without concept
	0.5	0.6	0.7	0.8	0.9	
4	0.641	0.812	0.763	0.752	0.750	0.701
8	0.744	0.812	0.804	0.752	0.759	0.733
16	0.758	0.812	0.804	0.770	0.811	0.794
32	0.782	0.824	0.810	0.785	0.827	0.804
64	0.809	0.848	0.835	0.836	0.840	0.831

In our CCHM representation approach, a number of concepts are inducted to enhance the features of documents. Due to the IDF operation and the decline factor, the weights of these concepts are smaller than the original terms. The effects of concepts are weak when concepts coexist with terms. It is the reason why the performance of K-means with concepts is not evident. The superiority of concept will be shown until the end steps of hierarchical clustering.

5 Conclusions

In this paper, we replace terms of documents with concepts in WordNet and construct Concept CHain Model(CCHM) for document representation. In this model, the semantic relations between terms are concerned. Concept chains are well integrated with an hierarchical clustering algorithm TCUAP. Concepts in different levels of the chain have distinct effects in each clustering iteration. High level concepts help to find semantic correlations of clusters in the last steps.

Our experiment on real textual data sets demonstrates that clustering with concept yields a better performance. Especially in the hierarchical approach TCUAP, the concepts improve the final results a lot. However, some defects still exist, like the time performance of the document preprocess with WordNet.

Acknowledgement

This work was supported by Chinese 973 Research Project under grant No. 2004CB719401.

References

1. Dumais, S. T., Furnas, G. W., Landauer, T. K., Deerwester, S.: Using latent semantic analysis to improve information retrieval. In Proc. of CHI'88 on Human factors in computing systems (1998) 281–285
2. Salton, G.: Automatic text processing: the transformation, analysis, and retrieval of information by computer. Addison-Wesley (1989)
3. Fellbaum, C., Miller, G.: WordNet: an electronic lexical database. MIT Press (1998)
4. Basili, R., Moschitti, A., Pazienza, M.: A text classifier based on linguistic processing. In Proc. of IJCAI'99, Machine Learning for Information Filtering (1999)
5. Song, S., Li, C.: TCUAP: a novel approach of text clustering using asymmetric proximity. In Proc. 2nd Indian International Conf. on Artificial Intelligence (2005)
6. Pissanetzky, S.: Sparse matrix technology. London: Academic Press (1984)
7. Lang, K.: NewsWeeder: Learning to filter netnews. In Proc. 12th International Conf. on Machine Learning, ICML'95 (1995) 331–339
8. Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Honkela, J., Paatero, V., Saarela, A.: Self organization of a massive document collection. In IEEE Transactions on Neural Networks, Vol. 11, **3** (2000) 574–585

An Efficient Range Query Under the Time Warping Distance

Chuyu Li, Long Jin, Sungbo Seo, and Keun Ho Ryu

Dept. of Computer Science, Chungbuk National University,
12, Gaesin-dong, Heungdeok-gu, Chungbuk 361-763, Korea
{lichuyu, kimlyong, sbseo, khryu}@dblab.chungbuk.ac.kr
<http://dblab.chungbuk.ac.kr/index.html>

Abstract. Time series are comprehensively appeared and developed in many applications. Similarity search under time warping has attracted much interest between the time series in the large databases. DTW (Dynamic Time Warping) is a robust distance measure and is superior to Euclidean distance. Nevertheless, it is more unfortunate that DTW has a quadratic time and the false dismissals are come forth since DTW distance does not satisfy the triangular inequality. In this paper, we propose an efficient range query algorithm based on a new similarity search method under time warping. When our range query applies for this method, it can remove the significant non-qualify time series as early as possible. Hence, it speeds up the calculation time and reduces the number of scanning the time series. Guaranteeing no false dismissals the lower bounding function is advised that consistently underestimate the DTW distance and satisfy the triangular inequality. Through the experimental results, our range query algorithm outperforms the existing others.

1 Introduction

Time series is a ubiquitous form of data that appeared and developed in virtually applications, such as science, technology, business and entertainment. Similarity search in time series database has attracted much interest. It is of growing importance in many fields such as information retrieval, data mining and data warehousing [3,6].

Similarity search focused on two interesting problems. The one is whole sequence matching [1,5,7,10,11] and another is subsequence matching [4]. Similarity measure popularly used is the Euclidean distance [1,4,9] between the subject sequences. Although it can be computed relatively fast, it is not an intuitively effective distance measure due to its sensitivity to some outliers such as amplitude scaling, offset translation and distortions in the time axis. Dynamic time warping (DTW) [2,7] is a much robust distance measure, allowing similarity matching although one of the sequences can elastic shift along the time axis. However, it is unfortunate that DTW has a quadratic time. It fails to satisfy the triangular inequality [10] so that false dismissals [1,4] come forth. To overcome these limitations, previous researches proposed some computationally cheap lower bounding functions [5,7,8,10,11] instead of the calculation of the DTW distance that can guarantee no false dismissals.

In this paper, we propose an efficient ϵ -range query algorithm to operate the similarity search based on [8] in that one k -nearest neighbor algorithm only was introduced. It can speed up the calculation time of the DTW distance and reduce the number of scanning the time series. Through experimental result, our range query algorithm outperforms other existing approaches.

The rest of the paper is organized as follows. In Section 2, we will consider the background and related work. Section 3 will describe the efficient ϵ -range query algorithm proposed by us. In section 4 we will discuss the experimental evaluation. Finally, in section 5 conclusions are given.

2 Background and Related Work

We first summarize in table 1 a list of symbols used in the rest of this paper.

Table 1. List of symbols

Symbol	Definition
S	a data sequence
Q	a query sequence
s_i	the i -th element of the S
q_i	the i -th element of the Q
ϵ	the tolerance in range query
S^{seg}	the approximate segment sequence of S
s_i^{seg}	the i -th segment in the S^{seg}
$s_i^{seg}.max$	the maximum value of the i -th segment
$s_i^{seg}.min$	the minimum value of the i -th segment
s_i^{TI}	the time interval of the S

2.1 Dynamic Time Warping (DTW)

Definition 1: The DTW distance between two time series S, Q is

$$D_{DTW}(S, Q) = D_{base}(First(S), First(Q)) + \min \begin{cases} D_{DTW}(S, Rest(Q)) \\ D_{DTW}(Rest(S), Q) \\ D_{DTW}(Rest(S), Rest(Q)) \end{cases}$$

D_{base} can be any of the distance function defined in [10]. We can use the cumulative distance $\alpha(i, j)$ as the matrix distance $d(i, j)$ in the current cell and the minimum of the cumulative distance of the adjacent elements:

$$\alpha(i, j) = d(i, j) + \min\{\alpha(i, j - 1), \alpha(i - 1, j - 1), \alpha(i - 1, j)\}$$

We can find the optimal warp path according to the minimum warp cost. This process is illustrated in Fig. 1.

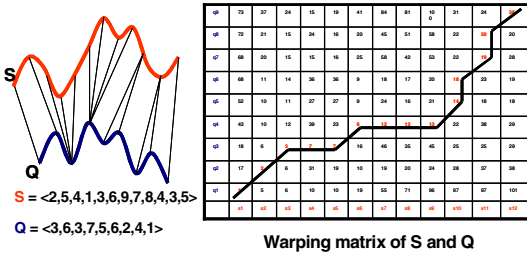


Fig. 1. Illustrate the DTW

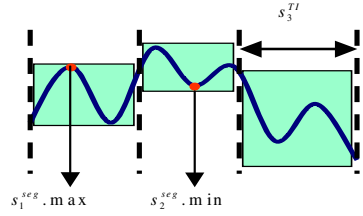


Fig. 2. Approximate segment

2.2 Related Work

The database community has been researching problems in similarity query for time series databases for many years. Agrawal et al. [1] utilized the Discrete Fourier Transform (DFT) to transform data from the time domain into the frequency domain and used the R*-tree to index the first few DFT coefficients. Yi et al. [10] first investigated the DTW in large database that used FastMap technique to approximate index the time series under dynamic time warping distance but it is not able to guarantee no false dismissal. Kim et al. [7] introduced the range query based on proposed lower bounding function that is employed on four features extracted from each time series that are first, last, greatest and smallest elements.

Keogh [5] proposed another lower bounding function and exact indexing of DTW, which was later optimized by Zhu and Shasha [11]. The proposed lower bounding function also guaranteed no false dismissals. Sakurai et al. [8] proposed a new method FTW (Fast search method for dynamic Time Warping), which efficiently pruned a significant number of the search candidates to reduce the search costs. They applied a *k*-nearest neighbor search algorithm during the similarity query processing. Utilizing this method as a start point we propose an efficient ϵ -range query algorithm under the time warping distance.

3 Proposed Method

In this section, we consider two processes that are coarsening computation and refinement during the range query that improves the search costs.

3.1 Coarsening Computation

3.1.1 Approximate Segment Sequence

Considering the time complexity of DTW $O(NM)$ where lengths of two time series are *N* and *M* respectively, it is an optimal method that we use the approximate segment sequence of the time series to compute the DTW distance between two time series especially for long sequences. First we divide one time series into several

segments by the time interval (TI) and divided time series is called the approximate segment sequence S^{seg} that is described as $S^{seg} = \{s_1^{seg}, \dots, s_i^{seg}\}$. Every separated segment s_i^{seg} is denoted by its maximum value $s_i^{seg}.max$, minimum value $s_i^{seg}.min$ and time interval s_i^{TI} as in Fig. 2. We can construct approximate segment sequences S^{seg} and Q^{seg} in terms to Fig. 1. We assume the time interval is 3 (that is $TI = 3$). Instead of compute the exact DTW distance between the time series S and Q using the D_{DTW} , we can use a new lower bounding function D_{TI-LB} to calculate the approximate DTW distance of S^{seg} and Q^{seg} . Moreover, the time complexity is reduced to $O(\frac{NM}{TI^2})$.

3.1.2 A New Lower Bounding Function

Definition 2: The approximate DTW distance between two approximate segment sequences S^{seg} , Q^{seg} is:

$$D_{TI-LB}(S^{seg}, Q^{seg}) = D_{base}^{seg}(First(S^{seg}), First(Q^{seg})) + \min \begin{cases} D_{TI-LB}(S^{seg}, Rest(Q^{seg})) \\ D_{TI-LB}(Rest(S^{seg}), Q^{seg}) \\ D_{TI-LB}(Rest(S^{seg}), Rest(Q^{seg})) \end{cases}$$

The cumulative distance $\beta(i, j)$ as the matrix distance $d^{seg}(i, j)$ in the current cell and the minimum of the cumulative distance of the adjacent elements:

$$\beta(i, j) = d^{seg}(i, j) + \min\{\beta(i, j-1), \beta(i-1, j), \beta(i-1, j-1)\},$$

$$d^{seg}(i, j) = \min(s_i^{TI}, q_j^{TI}) \times D^{seg}(s_i^{seg}, q_j^{seg}),$$

where $D^{seg}(s_i^{seg}, q_j^{seg})$ denotes the distance between s_i^{seg} and q_j^{seg} . The distance between them can be obtained by following formula:

$$D^{seg}(s_i^{seg}, q_j^{seg}) = \begin{cases} (s_i^{seg}.min - q_j^{seg}.max)^2 & s_i^{seg}.min > q_j^{seg}.max \\ (q_j^{seg}.min - s_i^{seg}.max)^2 & q_j^{seg}.min > s_i^{seg}.max \\ 0 & (otherwise) \end{cases}$$

Theorem 1: For any two time series S and Q , S^{seg} and Q^{seg} are their approximate segment sequences, the following inequality always holds.

$$D_{TI-LB}(S^{seg}, Q^{seg}) \leq D_{DTW}(S, Q).$$

Due to the limitation of the space we omit this proof here.

So we can easily derive one corollary from the theorem 1 that is:

$$D_{DTW}(S, Q) \leq \epsilon \Rightarrow D_{TI-LB}(S^{seg}, Q^{seg}) \leq \epsilon.$$

This corollary implies that similarity search that uses D_{TI-LB} rather than D_{DTW} in order to discard dissimilar sequences does not incur false dismissals.

```

Algorithm  $CC(S^{seg}, Q^{seg}, \epsilon)$ 
//Initialize every matrix cell,  $n$  and  $m$  are the length of  $S^{seg}$  and  $Q^{seg}$ 
for  $i=1$  to  $n$ 
    for  $j=1$  to  $m$ 
         $cell(i, j) = \infty$ ;
//  $w_i$  and  $w_j$  are the coordination of the element in the warping path
 $w_i=1$ ;  $w_j=1$ ;  $D_{seg}=0$ ;
while  $w_i \leq n$  and  $w_j \leq m$  do
    compute  $d(w_i, w_j)$ ;
    if  $d(w_i-1, w_j) = \min\{d(w_i, w_j), d(w_i-1, w_j), d(w_i, w_j-1)\}$   $w_i = w_i - 1$ ;
    if  $d(w_i, w_j-1) = \min\{d(w_i, w_j), d(w_i-1, w_j), d(w_i, w_j-1)\}$   $w_j = w_j - 1$ ;
    for  $i=w_i$  to  $n$ 
        compute  $d(i, w_j)$ ;
        if  $d(i, w_j) \leq (i/n)*\epsilon$ 
             $cell(i, w_j) = d(i, w_j)$ ;
        else break;
    for  $j=w_j$  to  $m$ 
        compute  $d(w_i, j)$ ;
        if  $d(w_i, j) \leq (j/m)*\epsilon$ 
             $cell(w_i, j) = d(w_i, j)$ ;
        else break;
//Compute the distance  $D_{seg}$  of  $S^{seg}$  and  $Q^{seg}$ , operate the range query
 $D_{seg} = D_{seg} + d(w_i, w_j)$ ;
if  $D_{seg} > \epsilon$ 
    break;
 $w_i = w_i + 1$ ;  $w_j = w_j + 1$ ;
if  $D_{seg} \leq \epsilon$ 
    return  $D_{seg}$ ;
else return null;

```

Fig. 3. Coarsening Computation algorithm

3.1.3 Coarsening Computation

To compute the DTW distance, every cell of the matrix must be filled. Through reduce the calculation of the matrix cells we can speed up the computation of the distance. During the process of finding the warping path we can get every element in this warping path and its coordinate are w_i and w_j respectively. Beginning from $cell(1,1)$, every element of the optimal warping path can be got in terms of DTW definition and the proposed new lower bounding function. As soon as getting a suitable element, we can fill those matrix cells that start from the coordinate of this element and lie in the row and column. After filling them, the computed distance value of every cell is compared with the value $(i/n)*\epsilon$ (along the w_i) and $(j/m)*\epsilon$ (along the w_j) to determine whether all of the cell values are computed. In the Fig. 4, due to $cell(1,2) \geq (2/3)*\epsilon$ we will not consider matrix $cell(1,3)$. As a result we can get some white cells in one matrix that need not to compute. During computing the approximate distance, we operate range query to confirm whether two sequences are similar simultaneously. It can save a lot of computed time that we do not need to compute whole distance but partly distance. We call this calculative process and range query for coarsening computation. Our algorithm in the coarsening computation process (CC algorithm) is described in the Fig. 3.

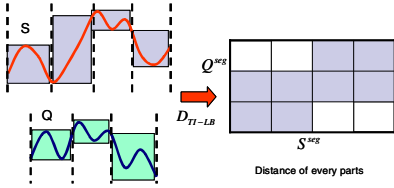


Fig. 4. Calculation of the DTW distance between S^{seg} and Q^{seg}

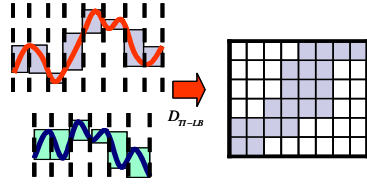


Fig. 5. Farther refine and calculate the approximate distance between S^{seg} and Q^{seg}

3.2 Refinement

In this paper we regard the time interval as the division granularity. We always get the time interval that is as larger as possible. According to different refinement the time interval will be decreased gradually. We propose a data structure to store every time series in the database. This data structure is a simple array that stores the information of every approximate segment sequence divided by different time intervals.

$$N > TI_x > TI_{x-1} > \dots > TI_1 > 1$$

$$A[S] = \{(S_1^{seg}, TI_1), (S_2^{seg}, TI_2), \dots, (S_x^{seg}, TI_x)\}$$

$$A[Q] = \{(Q_1^{seg}, TI_1), (Q_2^{seg}, TI_2), \dots, (Q_x^{seg}, TI_x)\}$$

After coarsening computation process is finished we begin to the refinement process. Followed the Fig. 4 if the data sequence satisfies the condition $D_{seg}(S^{seg}, Q^{seg}) \leq \epsilon$, we will continue to refine this sequence using smaller time interval (see Fig. 5) and calculate the approximate distance between two approximate segment sequences. Fig. 6 describes our ϵ -range query algorithm, TI-RangeSearch, which uses the array data structure.

```

Algorithm TI-RangeSearch (Q,  $\epsilon$ )
ResultSet = {}; compute A[Q]; //ResultSet stored candidate time series
for each S  $\in$  Database
    compute A[S];
    for i=x to 1
         $D_{seg} = CC(S^{seg}, Q^{seg}, \epsilon)$ ;
        if  $D_{seg} = null$ 
            break;
        else if i=1
            put S into the ResultSet;
// Post-processing step
for each S  $\in$  ResultSet
    if  $D_{DTW}(S, Q) > \epsilon$ 
        remove it from ResultSet;
return ResultSet;
    
```

Fig. 6. TI-RangeSearch algorithm

4 Experiments

We applied two kinds of data set in our experiment: Sunspot data set (that comes from <http://xweb.nrl.navy.mil/timeseries/multi.diskette>) and synthetic data set. To evaluate the superior performance of our range query algorithm, we compared TI-RangeSearch that are indexed by the array with the previous ones: TW-Sim-Search [7], Keogh-RangeSearch [5] and Zhu-RangeSearch [11].

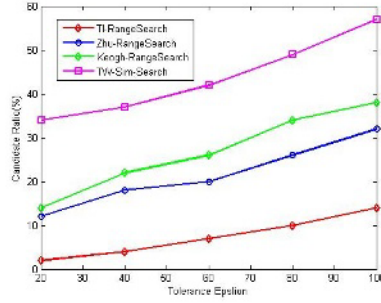


Fig. 7. Candidate ratio using the Sunspot data set

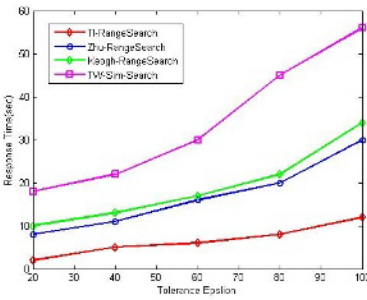


Fig. 8. Response time using Sunspot data set

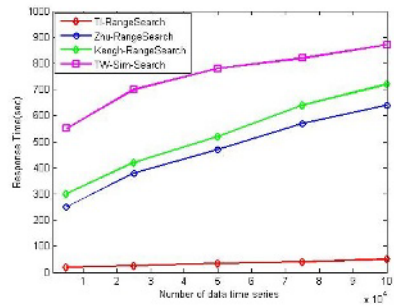


Fig. 9. Response time using synthetic data set

4.1 Candidate Ratio and Response Time

Candidate ratio is the important indicator of the filtering effect of all kinds of range query algorithms. This ratio is the smaller the better.

$$candidate\ ratio = \frac{\text{the number of candidate time series}}{\text{the number of data time series}}$$

Our first experiment compared the filtering effect of four methods using the Sunspot data set. Fig. 7 described the results of the candidate ratio of four different methods. Candidate ratio and response time are two indivisible parts for overall experimental performance. Fig. 8 showed the response time of four methods for the Sunspot data set. As a result, for overall performance the TI-RangeSearch is best one

among all methods. To verify the scalability of all methods, we used the synthetic data set in the following experiment due to not large enough Sunspot data set. Our experiment increased the number of the synthetic data set from 1,000 to 100,000. Under the fixed tolerance ε , the experimental result was described in the Fig. 9.

5 Conclusions

In this paper, we proposed an efficient ε -range query algorithm under the time warping distance. Used our algorithm the non-qualify time series are pruned as early as possible before computing the exact DTW distance. The computation time and quantity of time series scanned in the large time series database are significantly reduced during the calculation of the DTW distance. For the new lower bounding function it satisfied the triangular inequality, guaranteed no false dismissal. Moreover, the similarity search can be applied between the different length time series. Through the experimental results our range query is superior to other existing approaches. In the future, we plan to further find a new method to calculate the DTW distance that is suitable with our range query algorithm.

Acknowledgement. This work was partially supported by ETRI (Telematics & USN Research Division) in Korea.

References

1. Agrawal, R., Faloutsos, C., Swami, A.: Efficient Similarity Search in Sequence Databases. In FODO (1993) 69-84
2. Berndt, D. J., Clifford, J.: Finding Patterns in Time Series: A Dynamic Programming Approach. In Advances in Knowledge Discovery and Data Mining (1996) 229-248
3. Chen, M. S., Han, J., Yu, P. S.: Data Mining: An Overview from Database Perspective. In IEEE TKDK (1996) 866-883
4. Faloutsos, C., Ranganathan, M., Manolopoulos, Y.: Fast Subsequence Matching in Time-series Databases. In Proc. ACM SIGMOD (1994) 419-429
5. Keogh, E. J.: Exact indexing of dynamic time warping. In VLDB (2002) 406-417
6. Kim, J. S., Lee, H. G., Seo, S. B., Ryu, K. H.: CTAR: Classification Based on Temporal Class-association Rules for Intrusion Detection. In WISA (2004) 84-96
7. Kim, S. W., Park, S., Chu, W. W.: An Index-based Approach for Similarity Search Supporting Time Warping in Large Sequence Databases. In ICDE (2001) 607-614
8. Sakurai, Y., Yoshikawa, M., Faloutsos, C.: FTW: Fast Similarity Search under the Time Warping Distance. In PODS (2005)
9. Seo, S. B., Jin, L., Lee, J. W., Ryu, K. H.: Similarity Pattern Discovery Using Calendar Concept Hierarchy in Time Series Data. In APWeb (2004) 565-571
10. Yi, B. K., Jagadish, H. V., Faloutsos, C.: Efficient Retrieval of Similar Time Sequences under Time Warping. In IEEE ICDE (1998) 201-208
11. Zhu, Y., Shasha, D.: Warping Indexes with Envelope Transforms for Query by Humming. In ACM SIGMOD (2003) 181-192

Robust Scene Boundary Detection Based on Audiovisual Information

Soon-tak Lee¹, Joon-sik Baek¹, and Joong-hwan Baek²

¹ Multimedia Retrieval Lab. in School of Electronics and Communication Engineering,
Hankuk Aviation University
stlee@telechips.com, waterbat@mail.hangkong.ac.kr

² School of Electronics and Communication Engineering, Hankuk Aviation University
jhbaek@mail.hangkong.ac.kr

Abstract. In this paper, an efficient and robust scene change detection algorithm is proposed by using low-level audiovisual features and several classification methods. The proposed algorithm consists of three stages. The first stage is shot boundary detection by using Support Vector Machine (SVM) and the second stage is the scene boundary detection using shot clustering based on visual information. In the last stage, the scene boundary correction with audio information is described.

1 Introduction

To detect scene boundary, several researches have focused on visual similarity of video sequence. In [3], shot grouping based on visual similarity was proposed. To calculate visual similarity, color histogram and MPEG-1 motion vector is used. However, by using only video information, accuracy of scene boundary detection is poor.

In this paper, we present a robust and efficient content-based video summary technique on scene level by using both video and audio data. The proposed technique consists of three stages; splitting video sequence into shot unit, scene boundary detection based on visual information, and scene boundary correction by using audio information. We used shot boundary detection method in [5] for the first stage, and it is not mentioned in this paper. In second stage, by using selected key-frames, distance matrix based on visual information between shots is extracted with variable length time-window method. After distance matrix extraction is finished, shot clustering is performed with extracted distance matrix to create shot clusters [1].

Previous researches have proposed sound categorization method to detect the scene boundary [7][8][9]. In this paper, we correct extracted scene boundaries by detecting shots containing background music on the assumption that the shots in a scene contain same background music. Shots are classified into two categories; music and non-music, and classification tree is used for classification method in this stage.

This paper is organized as follow. Shot clustering with variable length time-window method is provided in Section 2. In Section 3, scene boundary correction by using audio source is discussed. Experiment evaluation results are given in Section 4.

2 Scene Change Detection Based on Visual Similarity

In previous research, shot clustering method with time restriction has been accomplished. The most difficult one, however, is the decision of several parameters due to the various characteristic of shots. In this section, we propose an optimal and flexible threshold making which is robust to variation of shot characteristic. The previous method is briefly described in Section 2.1. In Section 2.2, we discuss the improvements of the previous method by using variable length time-window method.

2.1 Shot Clustering with Time Restriction [1]

In general, the number of shots included in a scene or length of a scene has limitation in time domain. Therefore, if the time distance of two shots is over than general length, the probability which they are included in same scene is low even though their visual similarity is high. By using time-window method, similarity between two shots which are far from each other would be ignored. After the distance matrix based on visual similarity is calculated, shot clustering is performed.

2.2 Variable Length Time-Window Method

The previous time-window method [1] is shown *below*.

$$d(S_i, S_j) = \begin{cases} d(S_i, S_j), & T_b(S_j) - T_e(S_i) \leq Th_T \\ 1, & otherwise \end{cases}, \quad i < j \quad (1)$$

where, $d_K(S_i, S_j)$ is the similarity between two shots S_i and S_j , and $T_b(S)$ and $T_e(S)$ represent the start time and the end time of the shot S respectively.

The length of time-window should vary according to length of each scene. To predict a proper length of each time-window, we propose variable length time-window method. The variable length time-window $Th_T(S_i, S_j)$ could be defined like shown *below*.

$$Th_T(S_i, S_j) = Th_B + \omega_S \cdot \left\{ \omega_C \cdot \frac{1}{\sum_{k=1}^{j-i} [d(S_i, S_{i+k}) - \mu_{d(S_i, S_j)}]^2} + \omega_R \cdot R(\gamma) \right\}, \quad i < j \quad (2)$$

$$where, \mu_{D(S_i, S_j)} = \frac{1}{j-i} \sum_{k=1}^{j-i} d(S_i, S_{i+k})$$

Th_B is basis time length and decided experimentally. ω_S , ω_C , and ω_R are constants for scaling and $R(\gamma)$ is ratio of the number of shots which are most similar to S_i . $R(\gamma)$ is described like *below*. $n(C)$ represents the number of units in set C and $D(f_n, f_m)$ is color-histogram intersection [2] between two frames.

$$R(\gamma) = \frac{n[C_{i,j}(\gamma)]}{j-i}, C_{i,j}(\gamma) = \{S_k \mid d(S_i, S_k) \leq \gamma\} \quad (3)$$

$$\text{where, } d(S_i, S_{i+k}) = \min_{f_n \in S_i, f_m \in S_{i+k}} D(f_n, f_m)$$

3 Scene Boundary Correction by Audio Analysis

In this paper, we present more suitable rule for scene boundary correction procedure by using audio analysis. The combination scheme is described in Fig. 1. In Section 3.1, we discuss about the relative characteristic between music and non-music component. Feature selection is described in Section 3.2. Classification Tree is used for classification between music shot and non-music shot.

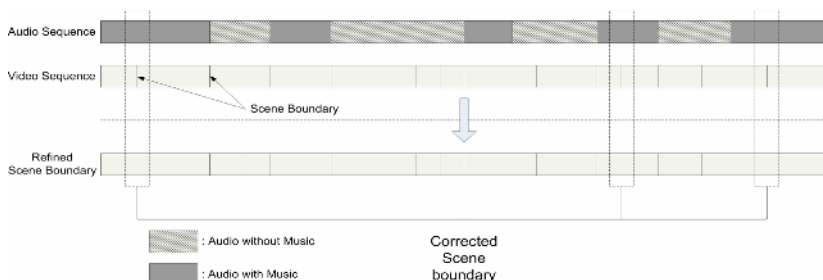


Fig. 1. The combination of the results from the video analysis and audio analysis is described

3.1 Audio Characteristic of Music, and Non-music Component

Since voice and music component are the most important contents of audio source, volume is relatively louder than another kinds of audio. However, in consideration of compactness characteristic of audio, voice has less compact sound than music due to the syllable. However, there are several cases which the volume of environmental sound is louder than music or voice, e.g. the sound of rain, waves, construction works, and so on. To discriminate between these kinds of sounds and music, analysis of frequency components is essentially required. In general, silence, voice, or environmental noise consists of low frequency component in the most part. On the other hand, most of all music contains sound of musical instruments which have a high tone. Therefore, by utilizing the frequency characteristic, the music component could be discriminated from another sound. By feature extraction based on these characteristics of audio, we could detect shots which contain music component.

3.2 Features Based on RMS and Fundamental Frequency for Music Detection

In this paper, we use statistical values of RMS and fundamental frequency of short time audio source for analysis of the sound characteristic. All audio features are

extracted from each shot, and then corresponding shot is categorized into class of shot with music or shot without music.

RMS efficiently and easily represents volume of audio signals, and computational cost is relatively low. But the number of windows in each shot is different according to the length of corresponding shot, and a proper method which integrates various number of RMS in a shot into fixed number of representations is required. As mentioned *above*, average volume level and sound compactness of a shot should be analyzed for detection of music. To precisely represent these characteristics, we use statistical parameters which are mean and variance of RMS. The representations of mean and variance of RMS are shown *below*, and $x_{i,j}(k)$ is the k -th sample in the i -th window of the j -th shot.

$$\mu_{rms}(j) = \frac{1}{n} \sum_{i=1}^n P(i, j), \sigma_{rms}^2(j) = \frac{1}{n} \sum_{i=1}^n [P(i, j) - \mu_{rms}(j)]^2$$

$$\text{where, } P(i, j) = \sqrt{\frac{1}{m} \sum_{k=1}^m x_{i,j}^2(k)}$$
(4)

We use fundamental frequency for frequency analysis. Since most of all environmental sounds are similar to noise or appeared intermittently, fundamental frequency is lower than music component and variance of short-time frequency is large. As used in combination of RMS values, mean and variance could be used for combination of fundamental frequency levels in a shot. But mean and variance are not enough to fully represent a frequency characteristic. That is because several redundantly high fundamental frequency components increase average level of fundamental frequency. Therefore, another representation is required for a combination method of fundamental frequency. That is clipping by using sigmoid function. The mean and the variance of clipped fundamental frequency are used for the audio feature, and described *below*.

$$\mu_{Fu}(j) = \frac{1}{n} \sum_{i=1}^n Fu_C(i, j), \sigma_{Fu}^2 = \frac{1}{n} \sum_{i=1}^n [Fu_C(i, j) - \mu_{Fu}(j)]^2$$

$$\text{where, } Fu_C(i, j) = \frac{1}{1 + \exp[\alpha \times Fu(i, j) + \beta]}, \beta \leq 0, 1 \leq \alpha$$
(5)

$Fu(i, j)$ represents the fundamental frequency of i -th window in j -th shot. By using clipping function based on sigmoid function, we prevent the increase of fundamental frequency average level due to several redundantly high fundamental frequency components.

4 Experimental Result

For a sample data set, two color films which have 25 fps are used; one is used for training, and the other is used for test. There are 135,000 and 79,500 frames respectively, at a spatial resolution of 160×120 . These video have each audio source which is digitized at 44.1 KHz with 16-bit precision in mono sound. For audio analysis, 249

audio clips which are containing music and 483 audio clips which are non-music are used for positive and negative sample set respectively. Due to the insufficiency of samples, classification tree was trained by using 10-fold cross validation.

To confirm the improvement by using proposed method, shot clustering with fixed length time-window method is accomplished on same sample set. We set fixed length of time-window as 110 sec. Results from both of them and music detection performance are described in Table 1 and Table 2 respectively. In this experiment, we prefer over-segmentation to less-segmentation. That is because less-segmentation directly means the loss of information. Note that the result of proposed method has fewer scenes than existing method, without reduction of performance. Therefore, proposed method could provide more precise information.

Table 1. Scene boundary detection result comparison. As a test video sample, ‘Love Letter’ is used. () is incorrect detection, and [**italic**] is corrected scene boundary by audio analysis. FL, VL, and Au mean fixed length time-window, variable length time-window, and audio analysis respectively.

Location	Duration (<i>h:mm:ss</i>)	Scene number	
		FL	VL & Au
Hill	0:00:20 ~ 0:02:57	1~(4)	1~(2)
Funeral(a)	0:02:57 ~ 0:04:02	(5)	(3)
Funeral (b)	0:04:02 ~ 0:05:41	6	4
In a car	0:05:41 ~ 0:06:37	7~9	5~7
Ichuki(m)’s Living Room	0:06:37 ~ 0:07:06	10~11	8~9
Ichuki(m)’s Room	0:07:06 ~ 0:10:30	12~22	10~13
Landscape of town	0:10:30 ~ 0:10:46	23	14
Ichuki(f)’s home(a)	0:10:46 ~ 0:11:25	24	15~16
Ichuki(f)’s home(b)	0:11:25 ~ 0:12:52	25~(28)	17~(19)
Ichuki(f)’s home(a)	0:12:52 ~ 0:15:16	(29)~30	(20)~21
Ichuki(f)’s room	0:15:16 ~ 0:16:44	31~34	22~24 [22]
Workroom	0:16:44 ~ 0:22:19	35~36	25
Ichuki(f)’s home	0:22:19 ~ 0:22:44	37	26
Library	0:22:44 ~ 0:28:30	43~44	27~28
Cafeteria	0:28:30 ~ 0:31:44	45~(46)	29
Way to Workroom2	0:31:44 ~ 0:32:26	(47)	30
Workroom2	0:32:26 ~ 0:33:13	(47)~48	31~33
Ichuki(f)’s yard	0:33:13 ~ 0:33:27	49~50	34~35
Ichuki(f)’s Living room	0:33:27 ~ 0:33:47	51	36
Recollection in hospital	0:33:47 ~ 0:35:32	52~(59)	37~41
Ichuki(f)’s yard	0:35:32 ~ 0:40:44	(60)~62	42~46 [42~43]
Ichuki(f)’s room	0:40:44 ~ 0:41:10	63	47
Road	0:41:10 ~ 0:43:00	64~67	48~50 [48]
Ichuki(m)’s room	0:43:00 ~ 0:47:49	68~72	51~54
High school classroom	0:47:59 ~ 0:53:51	73~80	55~59 [55~56]

Table 2. Performance of audio analysis

Category	# of samples	Recall Rate	Precision Rate
Shot with Music	249	84.36%	82.20%
Shot without Music	483	92.74%	95.24%

6 Conclusions

Existing content-based video summary which is used for shot boundary detection is not proper to provide a meaningful unit of video. Through this proposed method, more precise and meaningful content-based video summary has been available. The number of over-segmented scenes has been reduced by using proposed method and combination rule. However, combination rule between results from video and audio analysis is still poor. To maximize the efficiency of audio analysis, more robust and reasonable rule is required.

Acknowledgement. This research was supported by the Internet information Retrieval Research Center(IRC) in Hankuk Aviation University. IRC is a Regional Research Center of Kyounggi Province, designated by ITEP and Ministry of Commerce, Industry and Energy.

References

1. Minerva Yeung; Boon-Lock Yeo: Segmentation of Video by Clustering and Graph Analysis. *Computer Vision and Image Understanding*, vol. 71, no. 1. (1998) 94-109
2. Yeung, M.M.; Bede Liu: Efficient matching and clustering of video shots. *Image Processing, 1995. Proceedings., International Conference on* vol. 1. (1995) 338 - 341
3. Yong-Moo Kwon; Chang-Jun Song; Ig-Jae Kim: A new approach for high level video structuring. *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on* vol. 2, no. 30. (2000) 773-776
4. Ba Tu Truong; Venkatesh, S.; Dorai, C.: Scene extraction in motion pictures. *Circuits and Systems for Video Technology, IEEE Transactions on* vol. 13, Issue. 1. (2003) 5-15
5. S.B. Hong; W.Nah; J.H. Baek: Abrupt Shot Change Detection Using Multiple Features and Classification Tree. *IDEAL 4th International Conference on Intelligent Data Engineering and Automated Learning 2003, LNCS 2690*. (2003) 553-560
6. Richard O. Duda, Peter E. Hart, David G. Stork, *Pattern Classification 2nd edition*, Wiley Interscience, (2001) 259-265
7. Minami, K., Akutsu, A., Hamada, H., Tonomura, Y.: Video handling with music and speech detection. *Multimedia, IEEE*. Vol. 5, Issue. 3, (1998) 17 – 25
8. Zhang, T., Jay Kuo, C.-C.: Audio content analysis for online audiovisual data segmentation and classification *Speech and Audio Processing, IEEE Transactions on* Volume 9, Issue 4. (2001) 441 – 457
9. Lie Lu, Hong-Jiang Zhang, Hao Jiang: Content analysis for audio classification and segmentation *Speech and Audio Processing, IEEE Transactions on* Volume 10, Issue 7. (2002) 504 – 516

An FP-Tree Based Approach for Mining All Strongly Correlated Item Pairs

Zengyou He, Shengchun Deng, and Xiaofei Xu

Department of Computer Science and Engineering, Harbin Institute of Technology, China
zengyouhe@yahoo.com, {dsc, xiaofei}@hit.edu.cn

Abstract. Based on the FP-tree data structure, this paper presents an efficient algorithm for mining the complete set of positive correlated item pairs. Our experimental results on both synthetic and real world datasets show that, the performance of our algorithm is significantly better than that of the previously developed Taper algorithm over practical ranges of correlation threshold specifications.

1 Introduction

More recently, the discovery of statistical correlations is applied to transaction database [1]. The problem can be formalized as follows: Given a user-specified minimum correlation threshold and a transaction database, all-strong-pairs correlation query finds all item pairs with correlations above the minimum correlation threshold.

Different from the association-rule mining problem [e.g., 2-4], it is well known that an item pair with high support may have a very low correlation. Additionally, some item pairs with high correlations may also have very low support. Hence, we have to consider all possible item pairs in the mining process. Consequently, when the number of items and transactions is very large, candidate pair generation and test will be very costly.

To efficiently identify all strongly correlated pairs between items, Xiong et al. [1] proposed the Taper algorithm, in which an upper bound of Pearson's correlation coefficient is provided to prune candidate pairs. However, when the database contains a large number of items and transactions, even testing those remaining candidate pairs over the whole transaction database is still costly.

Motivated by the above observation, we examine other possible solutions for the problem of correlated pairs mining. We note that, the FP-tree [4] approach does not rely on the candidate generation step. We, therefore, consider how to make use of the FP-tree for discovering all item pairs with high positive correlation.

In this paper, we propose an efficient algorithm, called Tcp (FP-Tree based Correlation Pairs Mining), based on the FP-tree data structure, for mining the complete set of positive correlated pairs between items. Our experimental results show that Tcp's performance is significantly better than that of the Taper algorithm for mining correlated pairs on both synthetic and real world datasets over practical ranges of correlation threshold specifications.

2 Related Work

Brin et al. [5] introduced *lift* and a χ^2 correlation measure and developed methods for mining such correlations. Ma and Hellerstein [6] proposed mutually dependent patterns and an Apriori-based algorithm for mining such patterns. Recently, Omiecinski [7] introduced two interesting measures, called *all confidence* and *bond*, both of which have the downward closure property. Lee et al. [8] proposed two algorithms by extending the pattern-growth methodology [4] for mining *all confidence* and *bond* correlation patterns. As an extension to the concepts proposed in [7], a new notion of the *confidence-closed* correlated patterns is presented in [9].

In [1], efficiently computing Pearson's correlation coefficient for binary variables is considered. In this paper, we focus on developing an FP-Tree based method for efficiently identifying the complete set of strongly correlated pairs between items, with Pearson's correlation coefficient as correlation measure.

3 Background-Pearson's Correlation Coefficient

In statistics, a measure of association is a numerical index that describes the strength or magnitude of a relationship among variables. Relationships among nominal variables can be analyzed with nominal measures of association such as Pearson's Correlation Coefficient. The ϕ correlation coefficient is the computation form of Pearson's Correlation Coefficient for binary variables.

As shown in [1], when adopting the support measure of association rule mining [2], for two items A and B in a transaction database, we can derive the support form of the ϕ correlation coefficient as shown below in Equation (1):

$$\phi = \frac{\text{sup}(A, B) - \text{sup}(A)\text{sup}(B)}{\sqrt{\text{sup}(A)\text{sup}(B)(1 - \text{sup}(A))(1 - \text{sup}(B))}} \tag{1}$$

where $\text{sup}(A)$, $\text{sup}(B)$ and $\text{sup}(A, B)$ are the support of item(s) A , B and AB separately.

Furthermore, given an item pair $\{A, B\}$, the support value $\text{sup}(A)$ for item A , and the support value $\text{sup}(B)$ for item B , without loss of generality, let $\text{sup}(A) \geq \text{sup}(B)$. The upper bound $\text{upper}(\phi_{(A,B)})$ of an item pair $\{A, B\}$ is [1]:

$$\text{upper}(\phi_{(A,B)}) = \sqrt{\frac{\text{sup}(B)}{\text{sup}(A)}} \sqrt{\frac{1 - \text{sup}(A)}{1 - \text{sup}(B)}} \tag{2}$$

As shown in Equation (2), the upper bound of ϕ correlation coefficient for an item pair $\{A, B\}$ only relies on the support value of item A and the support value of item B . As a result, in the Taper algorithm [1], this upper bound is used to serve as a coarse filter to filter out item pairs that are of no interest, thus saving I/O cost by reducing the computation of the support values of those pruned pairs.

4 Tcp Algorithm

In this section, we introduce Tcp algorithm for mining correlated pairs between items. We adopt ideas of the FP-tree structure [4] in our algorithm. An FP-tree (frequent pattern tree) is a variation of the trie data structure, which is a prefix-tree structure for storing crucial and compressed information about frequent patterns.

The Tcp algorithm consists of two steps: FP-tree construction and correlation mining. In the FP-tree construction step, the Tcp algorithm constructs a FP-tree from the transaction database. In the correlation-mining step, we utilize a pattern-growth method to generate all item pairs and compute their *exact* correlation values. Fig.1 shows the main algorithm of Tcp. In the following, we will illustrate each step thoroughly.

Tcp Algorithm

Input: (1): A transaction database D

(2): θ : A user specified correlation threshold

Output: CP : All item pairs having correlation coefficients above correlation threshold

Steps:

(A) FP-Tree Construction Step

- (1) Scan the transaction database, D . Find the support of each item
- (2) Sort the items by their supports in descending order, denoted as *sorted-list*
- (3) Create a FP-Tree T , with only a root node with label being "NULL".
- (4) Scan transaction database again to build the whole FP-Tree with all ordered items

(B) Correlation Mining Step

- (5) For each item A_i in the header table of T do
 - (6) Get the conditional pattern base of A_i and compute its conditional FP-Tree T_i
 - (7) For each item B_j in the header table of T_i do
 - (8) Compute the correlation value of (A_i, B_j) , i.e., $\phi(A_i, B_j)$
 - (9) if $\phi(A_i, B_j) \geq \theta$ then add (A_i, B_j) to CP
 - (10) Return CP
-

Fig. 1. The Tcp Algorithm

(A) FP-Tree Construction Step

In the original FP-tree method [4], the FP-tree is built only with the items with sufficient support. However, in our problem setting, there is no support threshold used initially. We, therefore, propose to build a complete FP-tree with all items in the database (Step 3-4). Note that this is equivalent to setting the initial support threshold to zero. The size of an FP-tree is bounded by the size of its corresponding database because each transaction will contribute at most one path to the FP-tree, with the path length equal to the number of items in that transaction. Since there is often a lot of sharing of frequent items among transactions, the size of the tree is usually much smaller than its original database [4].

As can be seen from Fig.1, before constructing the FP-Tree, we use an additional scan over the database to count the support of each item (Step 1-2). At the first glance, one may argue that this additional I/O cost is not necessary since we have to build the complete FP-Tree with zero support. However, the rationale behind is as follows. (1). Knowing the support of each item will help us in building the more

compact FP-Tree, e.g., ordered FP-Tree. (2). In the correlation-mining step, single item support is required in the computation of the correlation value of each item pair.

(B) Correlation Mining Step

In the correlation-mining step, we utilize a pattern-growth method to generate all item pairs and compute their *exact* correlation values (Step 5-9), since we can get all support information from the FP-tree.

As can be seen from Fig.1, different from the Taper algorithm, Tcp algorithm doesn't need to use the upper bound based pruning technique, since in Step 8 and Step 9 we can directly compute the correlation values. Therefore, the Tcp algorithm is insensitive to the input minimum correlation threshold θ , which is highly desirable in real data mining applications. In contrast, the performance of Taper algorithm is highly dependent on both data distribution and minimum correlation threshold θ , because these factors have a great effect on the results of upper bound based pruning.

5 Experimental Results

All algorithms were implemented in Java. All experiments were conducted on a Pentium4-2.4G machine with 512 M of RAM and running Windows 2000.

We experimented with both real datasets and synthetic datasets. The mushroom dataset from UCI [10] has 119 items and 8124 transactions.

The synthesized datasets are created with the data generator in ARMiner software (<http://www.cs.umb.edu/~laur/ARMiner/>), which also follows the basic spirit of well-known IBM synthetic data generator for association rule mining. The data size (i.e., number of transactions), the number of items and the average size of transactions are the major parameters in the synthesized data generation. Table 1 shows the four datasets generated with different parameters and used in the experiments. The main difference between these datasets is that they have different number of items, range from 400 to 1000.

Table 1. Test Synthetic Data Sets

Data Set Name	Number of Transactions	Number of Items	Average Size of Transactions
T10I400D100K	100,000	400	10
T10I600D100K	100,000	600	10
T10I800D100K	100,000	800	10
T10I1000D100K	100,000	1000	10

We compared the performance of Tcp and Taper on the 5 datasets by varying correlation threshold from 0.9 to 0.1.

Fig. 2 shows the running time of the two algorithms on the mushroom dataset for θ ranging from 0.9 to 0.1. We observe that, Tcp's running time remains stable over the whole range of θ . The Tcp's stability on execution time is also observed in the consequent experiments on synthetic datasets. The reason is that Tcp doesn't depend

on upper bound based pruning technique, and hence is less affected by threshold parameter. In contrast, the execution time of Taper algorithm increases as the correlation thresholds are decreased. When θ reaches 0.4, Taper starts to outperform Taper.

Although the performance of Taper is not good as that of Taper when θ is larger than 0.4, Taper at least achieves same level performance as that of Taper on the mushroom dataset. The reason for the Taper's unsatisfied performance is that, for dataset with fewer items, the advantage of Taper is not very significant. As can be seen in consequent experiments, Taper's performance is significantly better than that of the Taper algorithm when the dataset has more items and transactions.

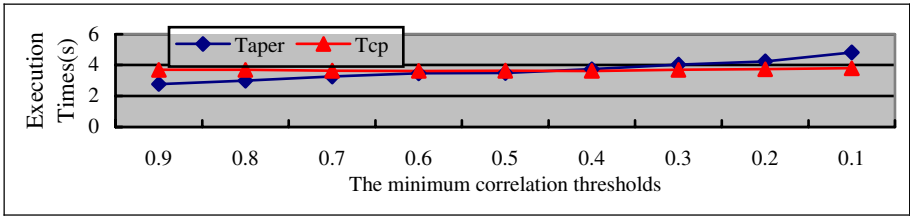


Fig. 2. Execution Time Comparison between Taper and Taper on Mushroom Dataset

Fig.3 shows the execution times of the two algorithms on four synthetic different datasets as correlation threshold is decreased.

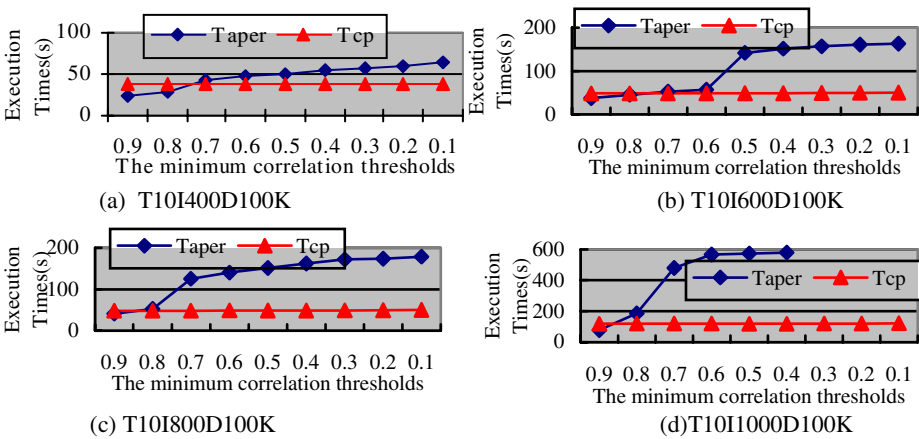


Fig. 3. Execution Time Comparison between Taper and Taper on 4 Synthetic Datasets

From Fig.3, some important observations are summarized as follows.

(1) Taper keeps its stable running time for the whole range of correlation threshold on all the synthetic datasets. It further confirmed the fact that Taper algorithm is robust with respect to input parameters.

(2) Taper keeps its increase in running time when the correlation threshold is decreased. And on all datasets, when correlation threshold reaches 0.7, Taper starts to outperform Taper. Even when the correlation threshold is set to be a very higher value (e.g., 0.9 or 0.8), Taper's execution time is only a little longer than that of Taper. Besides these extreme large values, Taper always outperform Taper significantly.

(3) With the increase in the number of items (from Fig.3(a) to Fig. 3(d)), we can see that the gap between Taper and Taper on execution time increases. Furthermore, on the T10I1000D100K dataset, when correlation threshold reaches 0.3, the Taper algorithm failed to continue its work because of the limited memory size. While the Taper algorithm is still very effective on the same dataset even when correlation threshold reaches 0.1. Hence, we can conclude that Taper is more suitable for mining transaction database with very large number of items and transactions, which is highly desirable in real data mining applications.

6 Conclusions

In this paper, we propose a new algorithm called Taper for mining statistical correlations on transaction databases. The salient feature of Taper is that it generates correlated item pairs directly without candidate generation through the usage of FP-Tree structure. Performance study demonstrates the superiority of our algorithm

Acknowledgements

This work was supported by High Technology Research and Development Program of China (No. 2003AA4Z2170, No. 2003AA413021) and IBM SUR Research Fund.

References

1. Xiong, H., Shekhar, S., Tan, P-N., Kumar V.: Exploiting a Support-based Upper Bound of Pearson's Correlation Coefficient for Efficiently Identifying Strongly Correlated Pairs. In: Proceedings of SIGKDD'04 (2004) 334-343
2. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: Proceedings of VLDB'94 (1994) 478-499
3. Zaki, M. J.: Scalable Algorithms for Association Mining. IEEE Transactions on Knowledge and Data Engineering **12(3)** (2000) 372-390
4. Han, J., Pei, J., Yin, J.: Mining Frequent Patterns without Candidate Generation. In: Proceedings of SIGMOD'00 (2000) 1-12
5. Brin, S., Motwani, R., Silverstein, C.: Beyond Market Basket: Generalizing Association Rules to Correlations. In: Proceedings of SIGMOD'97 (1997) 265-276
6. Ma, S., Hellerstein, J. L.: Mining Mutually Dependent Patterns. In: Proceedings of ICDM'01 (2001) 409-416
7. Omiecinski, E.: Alternative Interest Measures for Mining Associations. IEEE Transactions on Knowledge and Data Engineering **15(1)** (2003) 57-69
8. Lee, Y.K., Kim, W.Y., Cai, Y. D., Han, J.: CoMine: Efficient Mining of Correlated Patterns. In: Proceedings of ICDM'03 (2003) 581-584
9. Kim, W.Y., Lee, Y.K., Han, J.: CCMine: Efficient Mining of Confidence-Closed Correlated Patterns. In: Proceedings of PAKDD'04 (2004) 569-579
10. Merz, C. J., Merphy, P.: UCI Repository of Machine Learning Databases. [<http://www.ics.uci.edu/~mlern/MLRRepository.html>] (1996)

An Improved kNN Algorithm – Fuzzy kNN

Wenqian Shang¹, Houkuan Huang¹, Haibin Zhu², Yongmin Lin¹
Zhihai Wang¹, and Youli Qu¹

¹ School of Computer and Information Technology,
Beijing Jiaotong University, 100044, China
shangwenqian@hotmail.com

² Senior Member, IEEE, Dept. of Computer Science,
Nipissing University, North Bay, ON PIB 8L7, Canada
haibinz@nipissingu.ca

Abstract. As a simple, effective and nonparametric classification method, kNN algorithm is widely used in text classification. However, there is an obvious problem: when the density of training data is uneven it may decrease the precision of classification if we only consider the sequence of first k nearest neighbors but do not consider the differences of distances. To solve this problem, we adopt the theory of fuzzy sets, constructing a new membership function based on document similarities. A comparison between the proposed method and other existing kNN methods is made by experiments. The experimental results show that the algorithm based on the theory of fuzzy sets (fkNN) can promote the precision and recall of text categorization to a certain degree.

1 Introduction

kNN (k-nearest neighborhood) is one of the most famous statistic methods in pattern recognition area. It has developed for over forty years. The original kNN algorithm was put forward by T. M. Cover and P. E. Hart in [1]. Then many researchers develop it in theory deeply; it is one of the most important methods in nonparametric algorithm. With the advance of WWW (World Wide Web), text classification becomes a key technology to deal with and organize large numbers of documents. The kNN algorithm is a simple, effective and nonparametric classification method. It is widely used in text classification. It was studied in text classification very early [2] [3] [4] and is one of the algorithms that obtain better classification results in Reuter data sets (including Reuter 21450 and Apte data sets).

However, there are some problems need to be studied further, such as, how to improve decision rules, how to select k, how to select the feature set to make the classification result better and their effect to each other in classification performance. This paper is focused on the improvement of decision rules and its effect to classification performance. It adopts the theory of fuzzy sets, designing a new weighted factor. In the experiment of Chinese documents, this method shows a higher precision and recall. It solves the effect of uneven classes in the training set to classification performance effectively. The structure of the whole paper can be described as follows:

In Section 2, we discuss the classical kNN algorithm. In Section 3, we put forward a new decision rule based on the theory of fuzzy set. In Section 4, we use experiments to prove the new decision rule’s validity. In Section 5, we conclude this paper and point out the future work.

2 The Classical kNN Algorithm

The classical kNN algorithm is very simple: the more aggregate pattern in a pattern space belongs to the same class or at least the class they belong to has the same posterior distributing. Suppose there are c classes $\omega_1, \omega_2, \dots, \omega_c$ in a text classification. Fetch the unknown sample’s k -nearest neighbors, judge the majority of these k -nearest neighbors belong to some class, and then assign X to this class. Suppose k_1, k_2, \dots, k_c are respective sample numbers that belong to classes $\omega_1, \omega_2, \dots, \omega_c$ in k -nearest neighbors. The decision function can be defined as:

$$\mu_i(X) = k_i, i = 1, 2, \dots, c \tag{1}$$

According to this formula, the decision rule is: if $\mu_j(X) = \max_i k_i$, then $X \in \omega_j$. This is so called DVF (discrete-valued function) [7] rule.

But at present, the decision rule SWF (similarity-weighted function) [7] is widely used in kNN text classification systems. The system lookups k documents (called neighbors) that have the maximal similarity (cosine similarity) in training sets. According to what classes these neighbors are affiliated with, it grades the test document’s candidate class. The similarity between the neighbor document and the test document is taken as this class weight of neighbor documents. The decision function can be defined as follows:

$$\mu_j(X) = \sum_{i=1}^k \mu_j(X_i) sim(X, X_i) \tag{2}$$

Where $\mu_j(X_i) \in \{0, 1\}$ shows whether X_i belongs to ω_j ($\mu_j(X_i) = 1$ is True, $\mu_j(X_i) = 0$ is False); $sim(X, X_i)$ denotes the similarity between training document and test document. Then the decision rule is: If $\mu_j(X) = \max_i \mu_i(X)$, then $X \in \omega_j$.

3 The Improved Fuzzy kNN Decision Rule

According to the relationship between distance and similarity, let the distance between X and X_i be $d_0(X, X_i) = 1 - sim(X, X_i)$, in the meaning of “existing

fields” [5], adopting the membership function design method in fuzzy C-means algorithm [8][9][10] to compute the membership of candidate classifying sample to every class. After getting k-nearest neighbors’ known samples $\{X_i, i = 1, 2, \dots, k\}$ of the candidate classifying sample X, compute the membership of sample X to every class as follows:

$$\mu_j(X) = \frac{\sum_{i=1}^k \frac{1}{(1 - sim(X, X_i))^{2/(b-1)}}}{\sum_{i=1}^k \frac{1}{(1 - sim(X, X_i))^{2/(b-1)}}} \tag{3}$$

Where $j=1, 2, \dots, c$, $\mu_j(X_i)$ is the membership of known sample X to class j. If sample X belongs to class j then the value is 1, otherwise 0. From this formula, we can see that in reality the membership is using the different distance of every neighbor to the candidate classifying sample to weigh its effect. Parameter b is used to adjust the degree of a distance weight. From paper [6] we can know that the best value field of b is between 1.5 and 2.5, always using 2, in this paper we take the value 2. Then fuzzy k-nearest neighbors’ decision rule is: If $\mu_j(X) = \max_i \mu_i(X)$, then $X \in \omega_j$.

Using similarity as a weight factor in membership formula, we design the second membership formula as follows:

$$\mu_j(X) = \frac{\sum_{i=1}^k \mu_j(X_i) sim(X, X_i) \frac{1}{(1 - sim(X, X_i))^{2/(b-1)}}}{\sum_{i=1}^k \frac{1}{(1 - sim(X, X_i))^{2/(b-1)}}} \tag{4}$$

The meaning of every symbol in the formula is the same as above.

4 Experiment

4.1 The Datasets

Our experiment’s data set comes from the International Database Center, Dept. of Computing and Information Technology, Fudan University. The data set includes politics, economy, culture, military affairs, physical culture, education, agriculture, environment and energy sources etc., totally 15 different fields. The whole numbers of these documents are 6670.

We use 3148 documents as training samples and 3522 documents as test samples. The training samples are divided into document sets A and B. In document set A, the

class distribution is uneven. In these documents, the political documents are 619 pieces, occupying 34.43% of the training document set A, the energy sources documents are only 59 pieces, occupying 3.28% of the training document set A. In training sample B, the class distribution is even. Every class is 150 pieces. The circumstance of class distribution can be described as Table 1 and Table 2.

Surely, before using kNN and fkNN, we pre-process the documents, for example, getting rid of the stop words, stemming, fuzz matching and so on. Here, we adopt the N-gram method to extract attribute vectors, using Information Gain to extract feature vectors.

Table 1. The class distribution in training set A

Category	Example	Category	Example
Space	87	Politics	619
Computer	360	Sports	150
Environment	150	Education	59
Agriculture	73	History	150
Economy	150		

Table 2. The class distribution in training set B

Category	Example	Category	Example
Space	150	Politics	150
Computer	150	Sports	150
Environment	150	Art	150
Agriculture	150	History	150
Economy	150		

4.2 Accuracy Measure

For a classification system, Recall and Precision always affect each other, so in order to reflect the classification system's performance generally, C.J. Van Rijsbergen [11] bring forward using F1 to express the classifier's performance, the formula is as follows:

$$F1 = \frac{2 \times \text{Recall} \times \text{Precision}}{(\text{Recall} + \text{Precision})} \quad (5)$$

In this paper, we use this formula to evaluate the performance of classification.

4.3 Experiment Result and Analysis

In our experiment the feature dimension is 2000, the step of k is 10. The experiment result can be described as Table 3 and Table 4:

Table 3. The classification performance of four decision functions in training set A

Value k	Formula (1)		Formula (2)		Formula (3)		Formula (4)	
	Macro- F1	Micro- F1	Macro- F1	Micro- F1	Macro- F1	Micro- F1	Macro- -F1	Micro- F1
10	82.295	80.488	81.972	79.907	83.396	81.882	83.802	82.346
20	80.780	77.816	81.077	78.397	82.456	80.488	82.564	80.720
30	79.827	76.790	80.307	76.887	81.946	79.559	81.876	79.443
40	77.909	73.171	78.458	74.100	80.641	77.468	81.016	78.165
50	76.281	70.616	77.657	72.822	79.679	76.190	80.232	77.120

Table 4. The classification performance of four decision functions in training set B

Value k	Formula (1)		Formula (2)		Formula (3)		Formula (4)	
	Macro- F1	Micro- F1	Macro- F1	Micro- F1	Macro- F1	Micro- F1	Macro- -F1	Micro- F1
30	84.479	83.889	85.254	84.667	86.091	85.667	85.796	85.444
40	84.040	83.000	85.444	84.556	85.701	85.111	86.067	85.556
50	83.987	82.889	85.121	84.222	85.926	85.222	86.327	85.778
60	84.393	83.333	84.787	83.778	85.656	84.889	86.253	85.556
70	83.656	82.222	84.828	83.667	85.444	84.556	86.418	85.667
80	83.565	82.000	83.863	82.444	85.361	84.333	86.517	85.667

From Table 3 and Table 4, we can see that the classification performance of four decision functions in training set A is a little lower than in training B. This conclusion is the same as paper [4], that is, when the class distribution is uneven, the classification performance of kNN has some drop. But our experiment results show that regardless whether the class distribution is even or uneven, the improved fuzzy kNN algorithm has an increase in classification performance, i.e.:

Formula (4)>Formula (3)>Formula (2)>Formula (1)

Where in the left of “>”, Macro-F1 and Micro-F1 are more better in classification performance than in the right of “>”. This shows that our improvement is effective.

5 Conclusion

In this paper, we develop an improved kNN algorithm----fkNN (fuzz kNN). We mainly improve it in the decision rule. This method behaves better classification performance when the class distribution is uneven.

In the future, we need to study further on how to select k; how to select feature set; how to improve the decision rule further, what their effects to be on each other and so on.

Acknowledgment

This research is partly supported by Beijing Jiaotong University Science Foundation under the grant 2004RC008.

References

1. Cover T. M., Hart P. E.: Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory*, Vol IT-13 (1967) 21–27
2. Masand B., Lino G., Waltz D.: Classifying news stories using memory based reasoning. In: *The 15th Ann Int ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR'92)*. For Collins, CO, USA: ACM Press (1992)
3. Iwayama M. and Tkenobu: Cluster-based text categorization: a comparison of category search strategies. In: *The 18th Ann Int ACM SIGIR Conf. on Research and Development in Information Retrieval (SIGIR'95)*. For Collins, CO, USA: ACM Press (1995) 273-281
4. Yang Y., Lin X.: A re-examination of text categorization methods. In: *the 22th Annual Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval*. New York: ACM Press (1999)
5. Zhao S.: *The method of fuzzy mathematics in pattern recognition*. School of the West-North Electronic Engineering Press, Xian (1987)
6. Wang J., Pan J., Zhang F.: Research on web text mining. *Journal of Computer Research and Development*, Beijing Vol 37 (5) (2000) 518-519
7. Hevi-Seok L.: An improved kNN learning based Korean text classifier with heuristic information. In: *Proceeding of the 9th International Conference on Neural Information Processing*, Singapore (2002) 732-735
8. Pal N. R., Bezdek J. C.: On cluster validity for the fuzzy c-mean model. *IEEE Trans. On Fuzzy System*, Vol 3 (3) (1995) 370-379
9. Gao X., Xie W.: A study on the weighting exponent of the FCM. *Acta Electronic Sinica*, Vol 28 (4) (2000) 80-83
10. Yu J.: On the fuzziness index of the FCM algorithm. *Chinese Journal of Computers*, Vol 26 (8) (2003) 969-972
11. Van Rijsbergen C.: *Information retrieval*, CUP, London, Butterworths, 1979

A New Integrated Personalized Recommendation Algorithm*

Hongfang Zhou^{1,2}, Boqin Feng¹, Lintao Lv², and Zhurong Wang²

¹ School of Electronics and Information Engineering,
Xi'an Jiaotong University, Xi'an, 710049, Shaanxi, China
zhouhf@xaut.edu.cn

² School of Computer Science and Engineering,
Xi'an University of Technology, Xi'an, 710048, Shaanxi, China

Abstract. Traditional information retrieval technologies can satisfy users' needs to some extent. But they cannot satisfy any query from different backgrounds, with different intentions and at different time because of their all-purpose characteristics. An integrated searching algorithm by combining filtering with collaborative technologies is presented in this paper. The user model is represented as the probability distribution over the domain classification model. A method of computing similarity and a method of revising user model are provided. Compared with the vector space model, the probability model is more effective on describing users' interests. Furthermore, collaborative-based technologies are used, and as a result the scalability of the new algorithm is highly enhanced.

1 Document Model and User-Interest Model

With the popularization of WWW, it becomes more and more difficult to retrieve appropriate information suitable for a general user. So some personalized systems are developed. But unfortunately, some of them cannot deal with web information with uncertainty and variation property, some of them cannot achieve ideal coverage, similarity measure and correlation coefficient. Accordingly, we present a more robust and reasonable searching algorithm. And the controlled experiments show that this algorithm is more effective on processing Web information.

To make comparison between documents and users' interests, a consistent model for them is required to be built. VSM is a traditional approach expressing documents' characteristics. As a fact of the variety, user-interests can be presented with suitable subject terms selected from Web logs [2]. But in this way a training procedure is needed and it involves large amount of works. Furthermore, it cannot deal with fuzzy information existed largely on Web. Thus a domain categorization model is built, and then probability distribution of all documents and user-interests based on the model are computed. Let us use $C = \{c_1, c_2, \dots, c_n\}$ to represent n clusters, where n is the magnitude of the model; c_j means the j -th domain. Thus a document d can be expressed as $d = \langle p(c_1 | d), p(c_2 | d), \dots, p(c_n | d) \rangle$ where for each cluster $c_i, i = 1, 2, \dots, n$.

*This work is supported by the National High Technology Research and Development (2001AA113182).

To the feature selection, two disadvantages exist in the general χ^2 -test algorithm. One is that it keeps noises or outliers that should be filtered. The other is that it deletes useful characteristic items that should be kept. In view of these reasons, we have modified the χ^2 -test to be robust for noisy data.

If we use $n(c_j, t)$ to denote the number of characteristic item t in cluster c_j , $n(c_j)$ to denote the sum of the number of all characteristics in cluster c_j , $|V|$ to denote the sum of the number of all distinct characteristics in document sets, we can get into the following equation according to Lidstone continuous law.

$$p(t | c_j) = \frac{n(c_j, t) + \lambda}{n(c_j) + \lambda |V|} \tag{1}$$

2 Update of User-Interest Model

User interests can be represented as a probability distribution over domain knowledge. When a document is recommended to the user, every item’s value can be adjusted according to user’s corresponding action.

$$p(c_j | u) = \frac{p(c_j | u) + \eta w p(c_j | d)}{1 + \eta w} \tag{2}$$

where η shows the changeable speed of weight, w expresses the weight value of cluster c_j in document d .

3 Personalized Recommendation Algorithm

After modeling document and user-interest, we can compute their similarity, then abstract our required information and filter out non-intended information according to computed results.

3.1 Computation of Similarity

Most recommendation system’s matching methods are largely based on Vector Space Model. Using a vector model, the relevance of document d_i and document d_j is computed as

$$Sim(d_i, d_j) = \frac{\sum_{k=1}^t [w_k(d_i)w_k(d_j)]}{\sqrt{\sum_{k=1}^t [w_k(d_i)]^2 \sum_{k=1}^t [w_k(d_j)]^2}} \tag{3}$$

where d_i, d_j are the i -th document and j -th document, each of which is represented with a vector of keyword weights and $w_k(d_i)$ is the k -th keyword weight in the vector representing the document d_i .

To the probability model, it is unsuitable obviously to compute directly their cosine similarity. Here, we use proposition below proposed in [4].

Proposition 1. When modeling user interests we assume it is independent of document, and the probability that the document d is recommended to user u can be calculated through the Eq. (4).

$$p(u | d) = p(u) \sum_{j=1}^n \frac{p(c_j | u)p(c_j | d)}{p(c_j)} . \tag{4}$$

As a fact there exist a large number of noises. Hence, we enhance our algorithm more robust by introducing some collaborative filtering technologies. We compute the prediction on a document i for a user u by computing the sum of the rating given by the user on the items similar to i . Each ratings is weighted by the corresponding similarity $S_{i,j}$ between items i and j . Thus, we can calculate the predication $R_{u,i}$ as

$$R_{u,i} = \frac{\sum_{Sim(i,j) < threshold} P(u, j)}{\sum P(u, j)} . \tag{5}$$

where threshold is a pre-specified parameter according application’s requirements. That is to say, document j is the document similar to document i . Finally, we rate the searching results in descending order according to the value of $R_{u,j}$.

3.2 Personalize Searching Algorithm

We can achieve our purpose if we re-rank the searching results according to the computed conditional probability value.

Algorithm:. A personalized searching algorithm by integrating filtering with collaborative technologies.

Input: a domain categorization model, a user interest model, keywords related to query, and a search engine.

Output: personalized searching results

1. Using the search engine to initial searching result set X
2. Setting iterative number to be zero
3. To the i -th document in X, calculating its probability distribution in domain categorization model
4. Calculating iterated ranking value by means of Eq.(10)
5. Computing the i -th document’s probability recommended to the user, and adding it to the list Y
6. If the i -th document is the last document in set X, turn to (7); otherwise, set $i = i + 1$
7. Ranking and output documents in the descending order according to the probability value in list Y

This algorithm is based on searching results, and probability distribution of every document in searching results over domain categorization model must be calculated. And this can improve the algorithm’s performance to a larger degree.

4 Experiment

The main focus in evaluating the top-*N* recommendations is to determine the extent to which the recommended Web pages match the actual user. To evaluate the quality of the recommendations, *recall* and *precision* have been widely used in recommendation systems research [4,5]. However, these two metrics are in conflict because increasing the size of the top-*N* set leads to an increase in recall but at the same time a decrease in precision. The *F1 metric* [5] is a widely used combination metric that gives equal weight to both recall and precision:

$$F1 = \frac{2 \times recall \times precision}{recall + precision} . \tag{6}$$

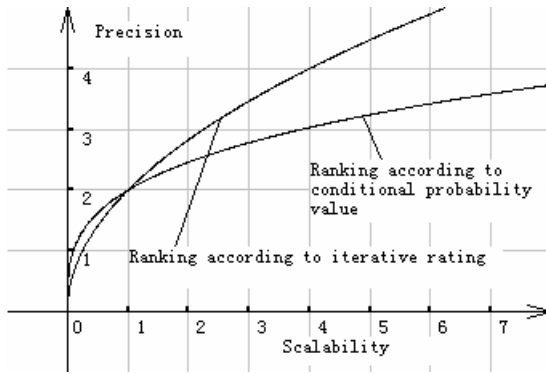


Fig. 1. Comparison between two ranking algorithms

The *F1 metric* was used for our quality experiments. And in view of consistency, we use precision replace F1 metric.

The data required for experiment is derived from Web server of a college’s library. Then the proposed algorithm is realized on VC++ platform. Lastly, experiment results are analyzed. Comparison between two ranking algorithms is shown in Fig.1. Our experiment show the proposed algorithm improves not only the searching precision, but also searching scalability.

5 Conclusion

Personalized services are the requirements and results of developing search engines. A general retrieval system cannot satisfy various queries from different background, different purpose, different knowledge and different time. In this paper, we discuss some related techniques in solving personalized services, propose probability distribution model expressing user-interest over different domains. Furthermore, in order to enhance algorithm’s scalability, we introduce related collaborative technology. Lastly, we made some experiments and the experiment results show that the algorithm’s

precision and scalability are both improved. In order to make recommendation system visualized and have better explanation, large masses of work require to be done. And this is just our further work.

References

1. A. Houston, Information classification: usability studies on two automatic approaches, Doctoral Dissertation, University of Arizona (1998).
2. Dumais ST, Platt J, Heckerman D, Sahami M. Inductive learning algorithms and representations for text categorization. In: French J, Gardarin G, eds. Proceedings of the International Conference on Information and Knowledge Management. New York: ACM Press, 1998. 148-155.
3. Zeng C, Xing CX, Zhou LZ. A survey of personalization technology. Journal of Software, 2002,13(10):1952~1961 (in Chinese with English abstract).
4. C. Basu, H. Hirsh, W. Cohen, Recommendation as classification: using social and content-based information in recommendation, in: Proceedings of the 1998 Workshop on Recommender Systems, AAAI Press, 1998, pp. 11-15.
5. B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Analysis of recommendation algorithms for e-commerce, in: Proceedings of ACM E-Commerce 2000 conference, 2000, pp. 158-167.

An Improved EMASK Algorithm for Privacy-Preserving Frequent Pattern Mining

Congfu Xu, Jinlong Wang, Hongwei Dan, and Yunhe Pan

Institute of Artificial Intelligence, Zhejiang University,
Hangzhou 310027, China

xucongfu@cs.zju.edu.cn, zjupaper@yahoo.com
danhow2008@hotmail.com, panyh@sun.zju.edu.cn

Abstract. As a novel research direction, privacy-preserving data mining (PPDM) has received a great deal of attentions from more and more researchers, and a large number of PPDM algorithms use randomization distortion techniques to mask the data for preserving the privacy of sensitive data. In reality, for PPDM in the data sets, which consist of terabytes or even petabytes of data, efficiency is a paramount important consideration in addition to the requirements of privacy and accuracy. Recently, EMASK, an efficient privacy-preserving frequent pattern mining algorithm, was proposed. Motivated by EMASK, in this paper, we improve on it, and present an improved algorithm BV-EMASK to furthermore enhance efficiency. Performance evaluation shows that BV-EMASK reduces the execution time significantly when comparing with EMASK.

1 Introduction

In recent years, due to the increasing applications in many areas, including decision support, market strategy and financial forecast, the importance of data mining is growing at a rapid pace. As an iterative and interactive process, data mining requires individual data sets to be gathered and studied collectively, threatening individual privacy, which makes privacy become one of the prime concerns in data mining research community. Thus, preserving privacy in data mining [1, 2, 3, 4, 5, 6, 7] has emerged to address the problem.

As a novel research direction, the main objective of privacy-preserving data mining (PPDM) is defined as meeting privacy requirements and providing valid data mining results [2, 3]. R. Agrawal and R. Srikant [1] first proposed using randomization to solve the problem of decision-tree mining from distorted database. Under the scheme of the randomization method, some other algorithms [4, 5, 6, 7] were proposed for data mining models building in preserving privacy, they acquired accurate mining results without accessing precise information in individual data records. Although in PPDM, privacy and accuracy are paramount important aspects, with the development of computing requirements, the runtime efficiency issue is more and more prominent. In [7], the authors first addressed the runtime efficiency issue in privacy-preserving frequent pattern mining and

proposed a new algorithm EMASK for improving computation speed through the application of symbol-specific distortion. EMASK algorithm was based on the classical Apriori algorithm [8]. Similar to Apriori-like [8], EMASK also suffered from tediously pattern matching in checking a large set of candidates and repeatedly scanning database.

Motivated by [7], in this paper, we further speed up the process in privacy-preserving frequent pattern mining, and an algorithm, called BV-EMASK (**Bit Vector-based EMASK**) is proposed. Different from [7], the algorithm applies column-wise fashion, where a column is regarded as a list of row identifiers represented by a bit vector, to privacy-preserving frequent pattern mining. The experimental results show that BV-EMASK has better efficiency performance than EMASK algorithm [7] proposed previously.

The remainder of this paper is organized as follows. The related works are described in Section 2. In Section 3, the bit vector representation is introduced to develop an efficient privacy-preserving frequent pattern mining algorithm BV-EMASK. The performance study is reported in Section 4. Finally, Section 5 concludes this paper.

2 Related Works

By the virtue of the randomization to distort the original database, frequent pattern can be mined in privacy preserving. All the papers [4, 5, 7] made use of randomization techniques that offer privacy while maintaining high utility for the data set in privacy-preserving frequent pattern mining. The work presented in [4] deals with categorical data in the context of frequent pattern mining. [5] presented a scheme called MASK to mine frequent patterns with secrecy constraints. In [7], the authors first addressed the runtime efficiency issue in privacy-preserving frequent pattern mining, and developed a new algorithm EMASK (Efficient MASK) to improve the efficiency.

3 The Algorithm BV-EMASK

In this section, an algorithm BV-EMASK, is represented for speeding up the process of frequent pattern mining in privacy preserving. In this approach, relational data tables are transformed into the bit vector representations, making computing easy, supporting fast frequency computing via intersection operations on these bit vector representations, avoiding expensive computing to check a large set of candidates by pattern matching, which is especially true for mining huge database.

3.1 Bit Vector Representation

In the Table 1, the sample transaction data set (Table 1(a)) is represented with the bit vector (Table 1(b)). In the sample data set, a set of items $I =$

$\{a, b, c, d, e, f\}$. For convenience, we write an itemset $\{a, c\}$ as $\{ac\}$. In order to speed up the computing efficiency, this paper represents the sample data set with bit vector. For each attribute item $a \in I$, its bit vector is a binary vector of $|D|$ dimensions ($|D|$ denotes the number of transactions in database). If the attribute set $\{a\}$ is contained in the i th transaction, the i th dimension in its bit vector is set to be 1; otherwise, the corresponding dimension is set to be 0. In this way, the original data set is transformed into bit vector representations.

Table 1. Transaction data set and its bit vector representation

(a) Data set		(b) Bit vector	
Transaction ID	Attribute	Attribute	Bit Vector
ID1	b	a	(00110)
ID2	b d e	b	(11001)
ID3	a d f	c	(00010)
ID4	a c e f	d	(01100)
ID5	b f	e	(01010)
		f	(00111)

In computing the support of the itemsets, for each single item $a \in I$, the number of dimensions with value 1 in its bit vector implies its support count in the database. For computing the support count of itemset $P \subseteq I$, consisting of k items, we performs AND operations on the bit vectors of its k attributes, then the number of dimensions with 1 in the result represents the support count of it.

3.2 The Algorithm BV-EMASK

The algorithm BV-EMASK makes use of the data distortion model in EMASK [7]. Before mining, the original database D is distorted into D^* , then the distorted database D^* is transformed into bit vector representations, in this way, the efficiency is improved.

In computing, because of the memory constraints, the bit vector representations of the entire database cannot sometimes be loaded into memory entirely. Our approach seeks to solve the scalable problem through horizontally partitioning the data set into subsets. Each subset is disjointed with each other and is sized to fit into the memory, then the entire bit vector representations will be loaded into memory partition by partition for support computing. In essence, by partitioning a transaction database into several partitions, the algorithm saves the I/O costs, and the accuracy will not be affected.

In Algorithm 1, BV-EMASK is described based on the bit vector representations introduced before. Function $Tr_BitVector(D^*)$ represents transforming the database D^* into the bit vector representations $BitVector(D^*)$. Function $sup(A, d_m^*)$ computes the support count of the itemset $A \subseteq I$ in data file d_m^* , $m \in \{1, 2, \dots, w\}$ ($D^* = \{d_1^*, d_2^*, \dots, d_w^*\}$).

Algorithm 1: BV-EMASK Algorithm

Data: Distorted data file D^* (each transaction in D^* is the set composed with $I = \{i_1, i_2, \dots, i_m\}$) through parameter p, q , minimum support s .

Result: Frequent itemsets Fp in the original database D .

begin

Step 1. Compute 1-itemset Fp_1 , at the same time transforming D^* into bit vector representations and save it into hard disk partition by partition;

for $i = 1; i \leq w; i++$ **do**

 Scan the transaction database d_i^* ;

 Tr_BitVector(d_i^*);

for each $a \in I$ **do**

$sup(a, D^*) += sup(a, d_i^*)$;

 Save BitVector(d_i^*) into hard disk;

 Reconstruct the support of item a in D with $sup(a, D^*)$ and p, q ;

 Generate Fp_1 ;

Step 2. Compute frequent patterns $Fp_k (k \geq 2)$ using *BitVector*(D^*);

for $k = 2; Fp_{k-1} \neq \emptyset; k++$ **do**

$C_k = gen_candidate(Fp_{k-1})$; /*generating candidate sets C_k^* */

while Load BitVector(D^*) in main memory partition by partition **do**

for each $c \in C_k$ **do**

$sup(c, D^*) += sup(c, d_i^*)$

 Reconstruct the support of itemset c in D with $sup(c, D^*)$ and p, q ;

 Generate Fp_k ;

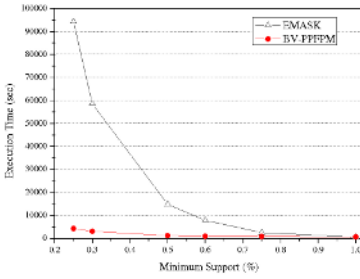
end

4 Performance Evaluation

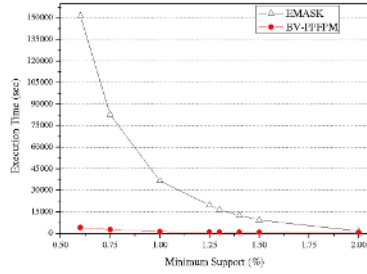
In order to show the efficiency and effectiveness of our approach, we have done an extensive performance study by comparing with EMASK [7]. Both algorithms are implemented using Cygwin with gcc 2.9.5. Our target platform is a Pentium4 3GHz processor, with 1GB memory. The operating system is Windows XP Professional with Service Pack 2. The experiments were performed on synthetic data sets (T25.I4.D1M.N1K and T40.I10.D1M.N1K), publicly available from IBM synthetic market-basket data generator [8].

In the following experiments, the original database is distorted with parameter $p=0.5, q=0.97$ as EMASK [7], and the execution time and efficiency with various minimum support threshold on the two databases are evaluated shown in Fig.1 and Fig.2.

Fig.1 shows that the execution time of BV-EMASK is much less than the time of EMASK on the two different data sets. When the setting on minimum support is decreased, the execution time of both algorithms increases, moreover, the increasing rates of time in both algorithms are dissimilar. As in the Fig.2, for the same data sets, the less the minimum support is, the more the execution time ratio (EMASK/BV-EMASK) is. For example, on the data set T25.I4.D1M.N1K, when the support decreases from 0.75% to 0.6%, the execution time ratio in-



(a) T25.I4.D1M.N1K



(b) T40.I10.D1M.N1K

Fig. 1. Execution Time

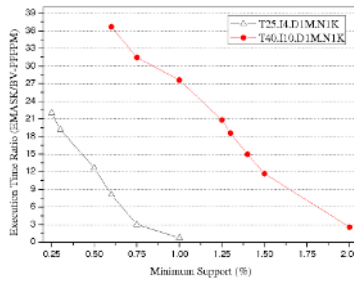


Fig. 2. T25.I4.D1M.N1K vs T40.I10.D1M.N1K

creases from 3.01 to 8.09. Especially, when the minimum support is lower than 0.5%, the execution time ratio increases significantly. As the Fig.2 shows, for the same minimum support, the more dense the data sets are, the much more quickly the execution time ratio increases. Just like Fig.2, when the minimum support is 0.75%, the execution time ratio increases from 3.01 to 31.46 from data set T25.I4.D1M.N1K to T40.I10.D1M.N1K.

From the above experiments, in general, BV-EMASK outperforms EMASK on execution time with respect to various minimum support parameters setting under different data sets. The algorithm BV-EMASK can produce higher efficiency compared with EMASK at low support values and on dense data set.

5 Conclusion

In PPDM, because of the consideration of privacy preserving, the original data is distorted, the runtime efficiency of mining the distorted data is decreased as compared with directly mining the original data. In order to achieve the performance improvement of the time requirements, in this paper, an algorithm

named BV-EMASK, improvement of EMASK, is proposed for mining privacy-preserving frequent pattern efficiently. BV-EMASK algorithm is designed based on the bit vector representations. According to the representations, the support of candidate patterns could be obtained efficiently by performing vector operations on corresponding bit vector. The experimental results show our approach has significant improvement on execution time than EMASK algorithm.

Acknowledgements

This paper was supported by the Natural Science Foundation of China (No. 60402010) and the Advanced Research Project sponsored by China Defense Ministry (No. 413150804, 41101010207), and was partially supported by the Aerospace Research Foundation sponsored by China Aerospace Science and Industry Corporation (No. 2003-HT-ZJDX-13).

References

1. Agrawal, R., and Srikant, R.: Privacy-preserving data mining. In: Proceedings of SIGMOD. (2000) 439-450
2. Clifton, C., Kantarcioglu, M., and Vaidya, J.: Defining privacy for data mining. In: Proceedings of the National Science Foundation Workshop on Next Generation Data Mining. (2002) 126-133.
3. Oliveira, S., and Zaïane, O.: Toward standardization in privacy-preserving data mining. In: Proceedings of DM-SSP (In conjunction with SIGKDD 2004). (2004)
4. Evfimievski, A., Srikant, R., Agrawal, R., and Gehrke, J.: Privacy preserving mining of association rules. In: Proceedings of SIGKDD. (2002) 217-228
5. Rizvi, S., and Haritsa, J.: Maintaining data privacy in association rule mining. In: Proceedings of VLDB. (2002) 682-693
6. Du, W., and Zhan, Z.: Using randomized response techniques for privacy-preserving data mining. In: Proceedings of SIGKDD. (2003) 505-510
7. Agrawal, S., Krishnan, V., and Haritsa, J.: On addressing efficiency concerns in privacy-preserving mining. In: Proceedings of DASFAA. (2004) 113-124
8. Agrawal, R., and Srikant, R.: Fast algorithms for mining association rules. In: Proceedings of VLDB. (1994) 487-499

CR*-Tree: An Improved R-Tree Using Cost Model

Haibo Chen¹ and Zhanquan Wang^{2,*}

¹ Department of Computer Science, Zhejiang University, Hangzhou 310027, China
chenhb@hzcnc.com

² Department of computer science and engineering,
East China University of Science and Technology, Shanghai 200237, China
wangz2212345@163.com

Abstract. We present a cost model for predicting the performance of R-tree and its variants. Optimization base on the cost model can be apply in R-tree construction. we construct a new R-tree variant called CR*-tree using this optimization technique. Experiments have been carried out ,results show that relative error of the cost model is around 12.6%,and the performance for querying CR*-tree has been improved 4.25% by contrast with R*-tree's.

1 Introduction

It is well known that the efficiency of the spilt algorithm of R-tree [1] and its variants changes with distribution and order of spatial data. This has caused very great random on algorithms realization. Some researchers attempt to set up a standard for judging what is a “good” R-tree. In the paper we present a cost model [2] predicting the performance of R-tree and its variants. An improvement algorithm of R-tree has be proposed through analysis of the cost model. The experiment has verified correctness of this kind of analysis.

The paper is organized as follows. Section 2 analyses the existing cost model and presents a new cost model for R-tree; Section 3 constructs a new variant of R-tree on the foundation of the new cost model ,which we called CR*-tree; Section 4 presents our experimental results that compare the new cost model and new variant with old ones; Section 5 concludes the paper. It must be pointed out that we have only considered two-dimensional spatial object in this paper, but the conclusion may be popularized to the higher dimensional spatial object.

2 Cost Model

[3] have proposed a simple spatial cost model. The model suppose that the spatial data are distributed evenly, the same with query windows of spatial data. The overlap probability Pr of query window $W(w_x, w_y)$ and spatial object $R(r_x, r_y)$ is:

$$Pr = (r_x + w_x) * (r_y + w_y). \quad (1)$$

Where w_x, w_y are height and width of W , r_x, r_y are height and width of R .

[4] has have already proved:

* Corresponding author.

$$\sum_{i=1}^M i \times p_i = \sum_{i=1}^M Pr_i = \sum_{i=1}^M (rx_i + wx) \times (ry_j + wy).$$

However the cost model is too strict with distribution of spatial data, which does not accord with the actual conditions. In order to avoid this kind of situation, [4] set up a new cost model according to a experience rule of distribution of spatial data. The experience rule is called Region Rule Law(RUL): $C(a) = k \times a^{-B}$, $k, B > 0, a > 0$. Where a is area of minimum rectangle border(MRB) of spatial object, $C(a)$ is number of spatial object whose area is greater than or equal to a ; k and b is experience parameter.[5] point out the realistic data follow this law, and $B \approx 0.5$. [5] present a new cost model according to this law. The cost model suppose the nodes of R-tree are not related to each other, and average number of nodes the nodes visited is sum of overlapping probability of query window and every node's MBR..

Let us consider about query performance for a full R-tree: According to tree structure of R-tree, every query on R-tree need to visit node along the tree one by one. Therefore given an query window, average visiting node number is not simply sum of overlapping probability of query window and every node's MBR, but is sum of product of the overlapping probability and overlapping number of query window and child nodes in the node.

Suppose that there is a full R-tree which constitutes by N spatial objects. the number of nodes in the level I is m^i , and $1 \leq i \leq H + 1$, $H+1$ is height of R-tree. m is maxium number of child nodes in one node of R-tree. In order to simplify the model, we translate MBR into Orthogonal unit coordinates space $(0, 1) \times (0, 1)$.

The number of nodes visited in the whole R-tree is the sum of the average number of sub-nodes visited in every level (root node must be visit):

$$p_n = 1 + \sum_{k=1}^H \left(\sum_{i=1}^{m^k} (PR_{ki} \times \sum_{j=(i-1)m+1}^{im} PR_{(k+1)j}) \right) \tag{2}$$

Suppose that all child nodes of node $R_{ki} (rx, ry)$ have the same size: $R_{(k+1)j}(\overline{rx}, \overline{ry})$, where:

$$\overline{rx}_{ki} = \sum_{j=1}^m rx_{(k+1)j}, \quad \overline{ry}_{ki} = \sum_{j=1}^m ry_{(k+1)j} \tag{3}$$

We deduct (6) from Formula(1), (3):

$$p_n = 1 + \sum_{k=1}^H \sum_{i=1}^{m^k} m (rx_{ki} + wx) \times (ry_{ki} + wy) \times (\overline{rx}_{ki} + wx) \times (\overline{ry}_{ki} + wy) \tag{4}$$

A cost mode for querying R-tree is :

$$p_n = 1 + \sum_{k=1}^H \sum_{i=1}^{n_k} n_{ki} (rx_{ki} + wx) \times (ry_{ki} + wy) \times \overline{(rx_{ki} + wx)} \times \overline{(ry_{ki} + wy)} \quad (5)$$

where n_{ki} is the number of child nodes of the i -th node in level k , n_k is number of nodes in level k . Because the formula has already considered to stratified structure among tree nodes, the result will be more accurate than (1) theoretically. we have compared the theoretical model with the actual result using true data through experiment behind, result show that relative error of the cost model is around 12.6%.

3 CR*-Tree

R*-tree [3] is one of best spatial indexing structure in all variants of R-tree. According to the new cost model ,we improve the R*-tree to a new tree called CR*-tree. there are two aspects on improving R*-tree:

1) reconstruct R*-tree dynamically base on cost model

Query efficiency is close correlation with spital distribution of child nodes in tree. There are less recollection disk operation, if it is smller overlapping probability of query window and inner sub-nodes in one node. therefore we need to considerate emphasisy how to reduce number of child nodes visited. .In fact, a query window often appears frequently in some specific space regions, We need to change the distribution of R-tree’s nodes along with the distribution of query winidow for constructing a “better” R-tree.

We record the frequency of query window, and adjust nodes of R-tree dynamicly according to these records. the rule of adjust is: merge certain nodes into one, if these nodes are often covered by the same query window.

Thus, the number of recollection disk operation will be reduced when a high frequency query window appears, along with reduction of number of accessing physic disk. This method is suitable for R-tree and all its variants.

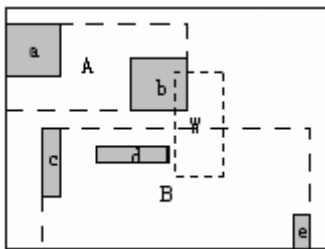


Fig. 1. A sample for node distribution

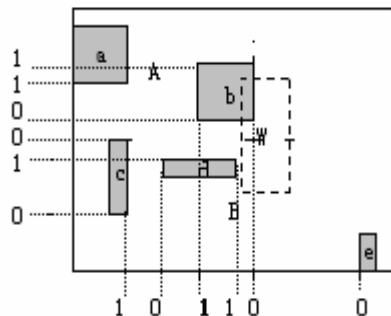


Fig. 2. An example of node split algorithm

[3] has explained characteristic of R-tree and R*-tree ,where we can find that the overlapping area of A and B in R*-tree is smaller than in R-tree, [3] conclude that R*-

tree' efficiency is better than the R-tree's. We will use the same example to explain the re-construction process of CR*-tree based on cost model.

We expressed leaf node in R*-tree for four-tuple, such as : (id,MBR,ptr, fre) ,in which fre is frequency of the node be visited. value of fre plus one automatically, Once the node is visited by query window. one thread will be reconstruct R-tree using algorithm RECreate when fre is bigger than certain valve value.The new R-tree will be not only spilt node according to distribution of spatial data,but also adjust by distribution of query windows.

```

Algorithm RECreate(RootNode:Root Node Of R*-tree)
begin
k:=h;
for j=1 to  $m_k - 1$  do
if(sec( $Node_{kj}$ ,  $Node_{(k+1)j}$ )=0) continue;
SortFre( $Node_{kj}$ ,  $Node_{(k+1)j}$ );
if(compare( $Node_{kj}$ ,  $Node_{(k+1)j}$ )<1) continue;
ReCreateNode( $Node_{kj}$ ,  $Node_{(k+1)j}$ );
Enddo
i=0;
while(mod( $m_k / m$ )>0) do
    CreateImNode();
    i:=i+1;
     $m_k = m_k - m$ ;
enddo
end

```

The child algorithm ReCreateNode arranges all nodes order by fre,and assigns m nodes ahead in the queue to $Node_{kj}$,others to $Node_{(k+1)j}$.Through reconstruct, the space Use factor approaches 100%

2) improve spilt algorithm

The algorithm in [3] need to compute overlap area, margin area and total area for every spilt strategies, which will consume a lot of resource. The basic idea for improving the spilt algorithm is to avoid these computation.We can spilt nodes by frequency of visiting node,let the fre value of these child nodes in the same parent close each other. the new algorithm is show in Fig2:

- A) For every rectangle compute the number of other rectangles which are pass through by its border extend line, sort ascending these rectangle
- B) divide into them two groups,in which variance of fre value is smallest.

Suppose that fre of every spatial object is: (a:0,b:4,c:1,d:3,e:0) ,we can conclude that spilt along with left border of object d is the best spilt strategies(the variance is 8.72)

```

Algorithm CR*TreeSpilt(E:set of entries)
begin
for each axis a do
sort E with respect to a
while(e∈E) do
compute the number of other rectangles which are pass
through by its's border extend line
Enddo
enddo
sort ascending these rectangle
if(the minnum value equals 0,and these is only one
rectangle e1)
return e1;
S=null;
while(e∈E and the number of rectangle overlaping with e
is smallest) do
compute the variance of two child nodes's fre
if(the variance is smallest)
S=e;
enddo
return S;
end
    
```

4 Experiment

We have carried on experiment to compute the relative error of the cost model, and to compare query performance of R*-tree with CR*-tree's. We use 1:100000 terrain map of Zhejiang (62108 polygons and 3741 polylines included) and 1:500 map of hangzhou(49337 polylines included)

The result showed in Fig3 and Fig4,in which the solid line in figures represents the number of visiting nodes computed by formalue (5),the dashed line represents the actual number of nodes visited in these experiment.

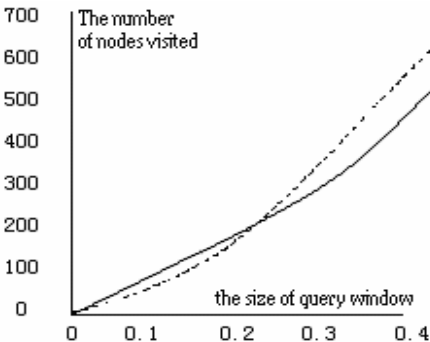


Fig. 3. The result of CR*-tree statistic zhejiang terrain map

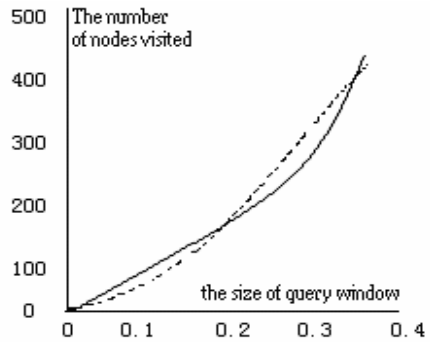


Fig. 4. The result of CR*-tree statistic for hangzhou terrain map

The following tables show statistical result of the experiment:

Table 1. The following tables show statistical result of the experiment

Query size	window	Map of Zhejiang			Map of Hangzhou		
		Theoretic/actual	Error	Percent	Theoretic/actual	Error	Percent
0.1		43/61	18	29%	95/91	4	4%
0.2		165/172	13	7%	163/159	4	2%
0.3		308/277	31	11%	305/257	48	18%
0.4		553/496	57	11%	427/531	96	18%

The average error of the cost model is $(29+7+11+11+4+2+18+18) / 8 = 12.6\%$.

We have also carried on experiment to analyze performance of CR*-tree. Experiments are carried out by government service personnel. The distribution of query window completely is decided by his business characteristic. The first reconstruct processing appeared after the 150th query. The result curve show in fig5 and Fig6.

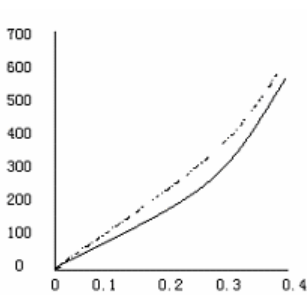


Fig. 5. The result after calling recreate algorithm for zhejiang terrain map

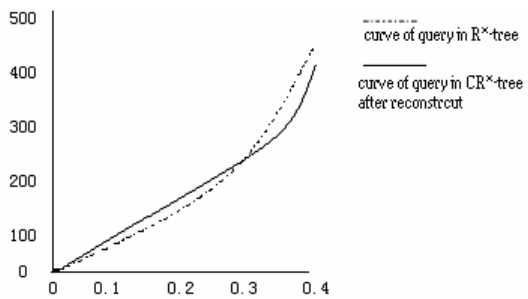


Fig. 6. The result after calling recreate algorithm for hangzhou terrain map

The statistical information show the following table:

Table 2. The statistical information show the following table

W	Map of Zhejiang			Map of Hangzhou		
	R*/CR*	NR	NR%	R*/CR*	NR	NR%
0.1	105/87	+18	17%	74/82	-8	-9%
0.2	230/193	+37	16%	146/157	-11	-7%
0.3	366/387	+21	5%	259/233	26	10%
0.4	582/623	-41	-7%	439/402	37	9%

where W is width of query window; R* is the number of R*-tree's nodes visited by the query and CR* is CR*-tree's; NR is the value of the number of R*-tree's nodes visited minus CR*-tree's; NR% is percent of NR according to the number of R*-tree's nodes visited.

5 Conclusion

In the paper we present a new cost model of R-tree, which has considered to relativity among nodes of R-tree. We constructed variant of R-tree called CR*-tree based on the model. Through experimental confirmation, the relative error of the cost model is around 12.6%, and the performance for querying CR*-tree has been improved 4.25% by contrast with R*-tree's.

References

1. Guttman, A.: R-tree: A Dynamic Index Structure for Spatial Search. In: Proceedings of the ACM SIGMOD Conference on Management of Data, Boston, Massachusetts (1984) 47–57
2. Theodoridis, Y., Sellis, T.: On the Performance Analysis of Multi-Dimensional R-tree-based Data Structures. Greece: Knowledge and Database Systems Laboratory, National Technical University of Athens, Greece (1995)
3. Rsv October. Spatial Access Methods: RSV, citeseer.nj.nec.com/rigaux99spatial.html.(2001)
4. BerndUwe Pagel. Towards a analysis of range query performance in spatial data structure. www.citeseer.nj.nec.com.(2001)
5. Proietti, G., Faloutsos, C.: Accurate Modeling of Region Data, Pittsburgh : Department of Computer Science, Carnegie Mellon University, Pittsburgh, USA(1998)
6. Theodoridis, Y., Sellis, T.: A Model for The Prediction of r-tree Performance. In Proceedings of the 15th ACM PODS symposium, Montreal, Canada (1996)

Grid-ODF: Detecting Outliers Effectively and Efficiently in Large Multi-dimensional Databases

Wei Wang¹, Ji Zhang², and Hai Wang³

¹ College of Educational Science,
Nanjing Normal University, China
njnuwangwei@yahoo.com

² Faculty of Computer Science,
Dalhousie University, Halifax, Canada
jizhang@dal.ca

³ Sobey School of Business,
Saint Mary's University, Halifax, Canada
hwang@smu.ca

Abstract. In this paper, we will propose a novel outlier mining algorithm, called *Grid-ODF*, that takes into account both the local and global perspectives of outliers for effective detection. The notion of Outlying Degree Factor (*ODF*), that reflects the factors of both the density and distance, is introduced to rank outliers. A grid structure partitioning the data space is employed to enable Grid-ODF to be implemented efficiently. Experimental results show that Grid-ODF outperforms existing outlier detection algorithms such as LOF and KNN-distance in terms of effectiveness and efficiency.

1 Introduction

In recent years, outlier detection has been seen as an important task in data mining that enjoys a wide range of applications such as detections of credit card fraud, criminal activity and exceptional patterns in databases. Outlier detection problem can be typically formulated as follows: given a set of data points or objects, find a specific number of objects that are considerably dissimilar, exceptional and inconsistent w.r.t the remaining data [5]. Recently, there have been numerous research work in outlier detection and the new notions such as distance-based outliers [9, 10, 13] and density-based local outliers [3, 8] have been proposed. However, the existing outlier detection algorithms are inefficient in dealing with large multi-dimensional datasets and most of them are only able to capture certain kinds of outliers. In this paper, we introduce a novel approach to find the *top_n* outliers in a large multi-dimensional database. Our approach takes into account the mechanisms used in detecting both global and local outliers. A new measurement of outlying factor of data points, called *Outlying Degree Factor (ODF)*, is proposed to measure the outlier-ness of points from both the global and local perspectives. The technique of grid-based data space partitioning is utilized to enable Grid-ODF to achieve a notable efficiency. The experimental results showing that our algorithm is both efficient and effective in detecting outliers compared to LOF and *k*-nearest neighbor distance methods.

2 Related Work

Literatures on outlier detection algorithms have been abundant in recent years and can be classified into following major categories based on the techniques used, i.e. *distribution-based*, *distance-based*, *density-based* and *clustering-based* methods. *Distribution-based* methods [2, 6] rely on the statistical approaches that assume a distribution (probability model) to fit the dataset. These methods cannot be applied in multi-dimensional scenario because they are univariate in nature. In addition, a lack of prior knowledge regarding the underlying distribution of the dataset renders the distribution-based methods difficult to use in practical. Finally, the quality of results cannot be guaranteed because they are largely depended on the distribution chosen to fit the data. [9] and [10] proposed the notion of *distance-based* outliers, i.e. $DB(pct, d_{min})$ -Outlier, which defines an object in a dataset as a $DB(pct, d_{min})$ -Outlier if at least $pct\%$ of the objects in the datasets having the distance larger than d_{min} from this object. In [13], the notion of distance-based outlier was extended and the distance to the k^{th} nearest neighbors of a point p , denoted as $D^k(p)$, is proposed to rank the point. Unlike distribution-based methods, distance-based methods do not rely on any assumed distribution to fit the data. Because they only examine the neighborhood for each object in the outlier detection, distance-based methods achieve better efficiency than distribution-based and depth-based methods. But, as [3] has pointed out, DB -Outliers has a limited ability in detecting outliers and cannot work well in some complex structure. Recently, a *density-based* formulation scheme of outlier was proposed in [3]. This formulation ranks the outlying degree of the points using Local Outlier Factor (LOF). Because LOF ranks points only considering the neighborhood density of the points, thus it may miss the potential outliers whose densities are close to those of their neighbors. [8] improved the efficiency of algorithm of [3] by proposing an efficient micro-cluster-based local outlier mining algorithm, but it still use LOF to mine outliers in dataset. The final category of outlier detection algorithm is *clustering-based*. So far, they are numerous studies on the clustering and a number of them are equipped with some mechanisms to detect outliers, such as CLARANS [11], DBSCAN[4], BIRCH[16], WaveCluster [15] and CLIQUE [1].

3 Observations in Human Perception of Outliers

Outliers can be viewed from either the local or global perspective. In the local perspective, human examine the point's immediate neighborhood and consider it as an outlier if its neighborhood density is low. The global observation considers the dense regions where the data points are densely populated in the data space. Now, we will discuss in details the two factors for deciding the outlying degree of a point:

(1) The Neighboring Density of the Point

The neighboring density of the point serves as a good indicator of its outlying degree. The lower the neighboring density of the point p , the higher the outlying degree it has, otherwise it is not. In Figure 1, two square boxes of equal size are used to delimit the neighborhood of points p_1 and p_2 . Because the neighboring density of p_1 is less than that of p_2 , so the outlying degree of p_1 is larger than p_2 .

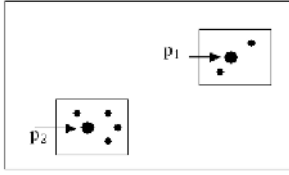


Fig. 1. Two points with different neighboring densities

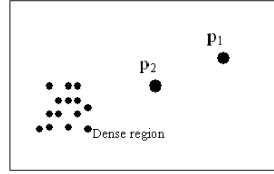


Fig. 2. Two points with different distances to the dense region

(2) The Distance Between the Point and the Dense Regions

The distance between the point and the dense regions reflects the similarity between this point and the dense regions. Intuitively, the larger such distance is, the more remarkably p is deviated from the main population of the data points and therefore the higher outlying degree it has, otherwise it is not. In Figure 2, we can see a dense region and two outlying points, p_1 and p_2 . Because the distance between p_1 and the dense region is larger than that between p_2 and the dense region, so the outlying degree of p_1 is larger than p_2 .

4 Formal Definitions of Top- k Outliers

Definition 1. Density of a Grid Cell. The density of a grid cell, denoted as $Den(C)$, measures the denseness of this cell. It is measured by the total number of points belong to the cell in the data space. A cell is dense if its density is high with respect to a human specified density threshold, say D .

Definition 2. Neighborhood Density Factor (NDF). The Neighborhood Density of a point p , denoted as $NDF(p)$, measures the denseness of p 's neighborhood. $NDF(p)$ is quantitized using the number of points falling into the grid cell to which p belongs.

Definition 3. Distance Factor (DF). The Distance Factor (DF) of point p with respect to a dense cell dc measures the distance between p and dc , indicating how far away is point p from dense cell dc . The Distance Factor of p with respect to dc is defined as the distance between p and the centroid of all the points residing in dc , i.e. $DF(p,dc)=Dist(p, Centriod(dc))$. $Dist()$ and $Centriod()$ are the distance function that computes the distance between two points and the function that returns the centroid of a group of points, respectively.

Definition 4. k -Distance Factor (k_DF). Dense regions can be viewed as the union of a few dense cells, and $DF(p)$ wrt a dense region can thus be defined as the average of DFs of p with respect to the k nearest dense cells of p , i.e. dc_1, dc_2, \dots, dc_k , i.e.

$$k_DF(p)=\frac{\sum_{i=1}^k DF (p , dc_i)}{k}$$

where k is input as a human-specified parameter. The concept of k nearest dense cells, unlike k^{th} nearest neighboring point used in [13], enables the measurement of outlier to reflect some global characteristics of the dataset.

Definition 5. Normalized NDF (N_NDF) and Normalized k_DF (N_k_DF). $NDF(p)$ and $k_DF(p)$ are normalized in order to make them comparable with each other in the computation of $ODF(p)$. Specifically, given the minimum and maximum values of NDF and k_DF among all the points examined in the dataset, $NDF(p)$ and $k_DF(p)$ can be normalized as

$$N_NDF(p) = \frac{(NDF(p)) - (NDF_{min})}{(NDF_{max}) - (NDF_{min})} \quad N_k_DF(p) = \frac{(k_DF(p)) - (k_DF_{min})}{(k_DF_{max}) - (k_DF_{min})}$$

Definition 6. Outlying Degree Factor (ODF). Combining the two factors of NDF and N_k_DF , Outlying Degree Factor (ODF) of a point p is defined as

$$ODF(p) = \frac{N_k_DF(p)}{N_NDF(p)}$$

$ODF(p)$ captures the degree to which we call p an outlier. The higher $ODF(p)$ is, the higher outlying degree of p is and vice versa. The top- k outliers are therefore defined as those points that have the k highest ODF values in the database.

5 Grid-Based Data Space Partition and Algorithm of Grid-ODF

In order to implement the computation of ODF of points efficiently, we use grid structure to partition the data space. The main idea of grid-based data space partition is to super-impose a multi-dimensional cube in the data space, with equal-volumed cells. It is characterized by the following advantages. First, $NDF(p)$ can be obtained instantly without the involvement of indexing techniques. Secondly, the dense regions can be efficiently identified, thus the computation of $k_DF(p)$ can be very fast. Finally, based on the density of grid cells, we will be able to select the top_n outliers only from a specified number of points viewed as outliers candidates, rather than the whole dataset. Now, we will present the details on the algorithm of Grid-ODF. Grid-ODF detects outliers in the following 5 major steps:

Step 1: Assigning Points. Each point in the dataset is read in sequentially and assigned into the grid. A linear scan of all cells in a multi-dimensional cube is often practically infeasible in terms of space and time. To address this problem, we employ hash method for point assignment. Without physically creating the multi-dimensional cube in main memory, Grid-ODF maps points into cells and hash the cells into a hash table.

Step 2: Sorting Occupied Grid Cells. The occupied cells are those cells that contain at least one data point. All the occupied cells are sorted in ascending order based on their density values.

Step 3: Producing the Outlier Candidates. Given top_n and m , the points residing in the first t cells in the sorting list in the second step that satisfies the inequality given in Section 5 will be chosen to be outlier candidates. To do so, Grid-ODF will examine the sorted list of primary cells starting from the cell having the smallest density, and terminate this step when the aggregation of the points in the cells that have been visited reaches $m*top_n$.

Step 4: Generating the Dense Cells. Given a density threshold D , the dense cells are found in the cell list if the density of the cell is larger than or equal to D .

Step 5: Computing ODFs and Ranking top_n Outliers. Compute the ODFs of all outlier candidates generated and pick the top_n outliers that have the top_n largest ODF values. This step involves searching the k nearest dense cells w.r.t each outlier candidate. Please note that the NDF_{min} , NDF_{max} , k_DF_{min} and k_DF_{max} used to normalize NDF and k_DF are obtained from all the outlier candidates, rather than from all the points in the dataset, to further speed up the detection process.

6 Experimental Results

We run experiments to evaluate the robustness of Grid-ODF and compare its performance against LOF and k -nearest neighbor distance (called KNN-distance for short) methods. To evaluate the quality of outlier detection result, we use Standard Deviation of Clusters (SDC) and Precision of outliers detected as the validation criteria.

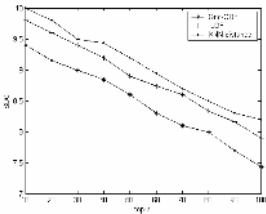


Fig. 3. SDC study

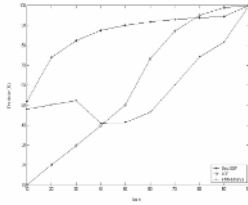


Fig. 4. Precision study

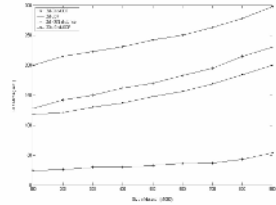


Fig. 5. Runtime study

Figure 3 shows the results of the comparative studies of these three algorithms in terms of SDC. Because it takes into account of both factors of density and distance in its outlier formulation, Grid-ODF outperforms LOF and KNN-distance in terms of SDC. Secondly, we study the precision of the three algorithms in detecting outliers. From Figure 4, we can see that Grid-ODF achieves the highest precision in almost all cases. The average precision of Grid-ODF is 85.9%, which remarkably higher than 60.4% of LOF and 59.9% of KNN-distance. We also plot the execution time of Grid-ODF against micro-cluster-based LOF [8] and partition-based KNN-distance[13] in Figure 5. From Figure 5, it is clear that Grid-ODF outperforms LOF and KNN-Distance by a margin of magnitude in terms of runtime. Even though drawing on clustering-based pruning approach, LOF and KNN-distance are still slow compared to Grid-ODF. We also see that Grid-ODF is well dimensionally scalable from Figure 5.

7 Conclusions

In this paper, we have proposed a novel outlier mining algorithm, called Grid-ODF, that takes the advantage of both the local and global perspectives in detecting outliers. The notion of Outlying Degree Factor (*ODF*) that integrates the factors of both density and distance is introduced to rank outliers. We have also illustrated how to efficiently implement Grid-ODF by using the technique of grid-based data space partition. The experimental results verify the promising performance of Grid-ODF.

References

1. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic Subspace Clustering of High Dimensional Data Mining Application. *SIGMOD'99*, Philadelphia, PA (1999)
2. Barnett, V., Lewis, T.: Outliers in Statistical Data. John Wiley, 3rd edition (1994)
3. Breuning, M., Kriegel, H-P, Ng, R., Sander, J.: LOF: Identifying Density-Based Local Outliers. *SIGMOD'00*, Dallas, Texas (2000)
4. Ester, M., Kriegel, H-P, Sander, J., Xu, X.: A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, *KDD'96*, Portland, Oregon (1996)
5. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufman Publishers (2000)
6. Hawkins, D.: Identification of Outliers. Chapman and Hall, London (1980)
7. Hinneburg, A., Keim, D. A.: An Efficient Approach to Cluster in Large Multimedia Databases with Noise. *KDD'98*, New York City, NY (1998)
8. Jin, W., Tung, A. K. H., Han, J.: Finding Top_n Local Outliers in Large Database. *SIGKDD'01*, San Francisco, CA (2001)
9. Knorr, E. M., Ng, R. T.: Algorithms for Mining Distance-based Outliers in Large Dataset. *VLDB'98*, New York, NY (1998)
10. Knorr, E. M., Ng, R. T.: Finding Intentional Knowledge of Distance-based Outliers. *VLDB'99*, Edinburgh, Scotland (1999)
11. Ng, R., Han, J.: Efficient and Effective Clustering Methods for Spatial Data Mining, *VLDB'94*, Santiago, Chile (1994)
12. Preparata, F., Shamos, M.: Computational Geometry: an Introduction. Springer-Verlag (1988)
13. Ramaswamy, S., Rastogi, R., Kyuseok, S.: Efficient Algorithms for Mining Outliers from Large Data Sets. *SIGMOD'00*, Dallas, Texas (2000)
14. Ruts, I., Rousseeuw, P.: Computing Depth Contours of Bivariate Point Clouds. *Computational Statistics and Data Analysis*. (1996) 23: 153-168
15. Sheikholeslami, G., Chatterjee, S., Zhang, A.: WaveCluster: A Wavelet based Clustering Approach for Spatial Data in Very Large Database. *VLDB Journal*, vol.8 (3-4) (1999) 289-304
16. Zhang, T., Ramakrishnan, R., Livny, M.: BIRCH: An Efficient Data Clustering Method for Very Large Databases. *SIGMOD'96*, Montreal, Canada (1996)

Clustering XML Documents by Structure Based on Common Neighbor

Xizhe Zhang¹, Tianyang Lv^{1,2}, Zhengxuan Wang¹, and Wanli Zuo¹

¹ College of Computer Science and Technology,
Jilin University, Changchun, China

² College of Computer Science and Technology,
Harbin Engineering University, Harbin, China
zxxok@163.com, raynor1979@163.com

Abstract. It is important to perform the clustering task on XML documents. However, it is difficult to select the appropriate parameters' value for the clustering algorithms. Meanwhile, current clustering algorithms lack the effective mechanism to detect outliers while treating outliers as "noise". By integrating outlier detection with clustering, the paper takes a new approach for analyzing the XML documents by structure. After stating the concept of common neighbor based outlier, the paper proposes a new clustering algorithm, which stops clustering automatically by utilizing the outlier information and needs only one parameter, whose appropriate value range is decided in the outlier mining process. After discussing some features of the proposed algorithm, the paper adopts the XML dataset with different structure and other real-life datasets to compare it with other clustering algorithms.

Keywords: XML Structure; Clustering; Common Neighbor.

1 Introduction

It becomes an interesting topic to retrieval information from the semi-structured data. The paper addresses the problem of clustering structurally similar XML documents. This research can help in devising indexing techniques for such documents and improving the query plans.

Several works have applied clustering algorithms in analyzing XML documents by structure. [1] proposes a method for clustering the DTDs of XML data source based on the similarity of structures and semantics of DTDs. [3] tries to extract common structures from XML documents by modeling a document as an ordered labeled tree, then the notions of tree edit sequence and tree edit distance originating from editing problems for rooted ordered labeled trees is applied.

However, these works just adopted some simple clustering algorithms and can not handle the following difficulties: first, it is difficult to select appropriate parameter's value for clustering algorithm; second, current clustering algorithms are also short at detecting outliers. Take the algorithm ROCK as an example.

ROCK^[5] is proposed to deal with categorical attribute by adopting the novel concepts of *common neighbor*. Its main idea is as follows:

$sim[a_1, a_2] = |a_1 \cap a_2| / |a_1 \cup a_2|$ for data a_1 and a_2 ; a_1 and a_2 are *neighbor* if $sim(a_1, a_2) \geq \theta$; a_3 is a *common neighbor* of a_1 and a_2 , if a_3 is the neighbor of both of a_1 and a_2 ; $link(a_1, a_2)$ is defined as the number of the *common neighbors* of a_1 and a_2 . ROCK assumes that data in the same cluster should have more *common neighbors*. Thus, clusters with the largest number of *common neighbors* are merged first. Let $link[C_i, C_j]$ be the sum of the number of common neighbors of data in C_i and C_j ; then

$$D(C_i, C_j) = \frac{link[C_i, C_j]}{(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}}, \quad f(\theta) = \frac{1-\theta}{1+\theta} \tag{1}$$

And the way to compute the common neighbors of C_{new} and a cluster C_p is:

$$link[C_{new}, C_p] = link[C_i, C_p] + link[C_j, C_p] \tag{2}$$

Obviously ROCK has several shortcomings: (1) needing the user-specified parameter k ; (2) interfered by the outliers before they are pruned and the information implied by outliers is wasted.

To overcome these stated problems, the paper proposes a new strategy that combines outlier detection with clustering. The strategy stops clustering automatically according to the dissimilarity degree implied by outliers. It is based on the following observation: the distances among data or clusters not only show their similarity degree, but also demonstrate the dissimilarity. With the progressing of clustering, the dissimilarity $D(C_{NN-A}, C_{NN-B})$ between the two most similar clusters C_{NN-A} and C_{NN-B} at present is increasing. And the clustering should stop at the moment when C_{NN-A} and C_{NN-B} are so diverse from each other.

In section 2, the paper develops the new concept of *common-neighbor* based outlier and proposed a new clustering algorithm As-ROCK. After discussing some features of As-ROCK in section 3, section 4 applies As-ROCK in analyzing the XML documents by structure and gives the comparison experimental results; section 5 summarizes the paper.

Table 1. Important Notations

Notation	Description
N	Total number of Data
M	Dimensionality of Data
K	Number of Final Clusters
C_i	The i th Cluster
$D(C_i, C_j)$	Distance between C_i and C_j

2 The Auto-stopped Algorithm Based on Common Neighbor

This section proposes the novel concept of *common-neighbor* based outlier and makes several improvements on ROCK algorithm. These works lead to a new clustering algorithm As-ROCK.

2.1 Outlier Mining Based on Common Neighbor

In the clustering mechanism based on *common neighbor*, the parameter θ can be used to detect outliers. However, this is not fully explored in the researches [5]. Suppose each data corresponds to a node; the weight of the edge that links a_i and a_j equals to $sim(a_i, a_j)$; thus, an appropriate θ will make some data/node isolated from all the others, like a_5 in Fig. 1. Obviously, this kind of data should be recognized as outliers. And the dissimilarity of outliers with the others is reflected as the much smaller *link*. Thus, the *common-neighbor based* outlier is defined as follows:

Data a is a *common-neighbor based* outlier, if $\forall b, link(a, b) = 0$.

Easy to prove, the *common-neighbor based* outliers include all outliers like a_5 .

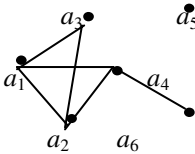


Fig. 1. Graph after neighbor computing

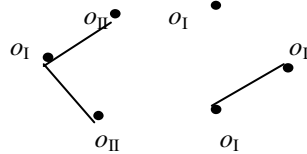


Fig. 2. Examples of Outliers

For any data a with an edge connected with a *common-neighbor based* outlier o_i , $link(a, b)=0$ for any b except o_i . Therefore, an extended definition of outliers is given as follows:

Data a is an outlier, if $sim(a, o_i) \geq \theta$ and o_i is a *common-neighbor based* outlier.

Fig. 2 is the examples of the two kinds of outliers.

2.2 Similarity Between Clusters

In the clustering process, ROCK approximately calculates the *common neighbors* between C_{new} and a cluster C_i according to equation 2. However, in some cases, the sum of $link[C_{NN-A}, C_i]$ and $link[C_{NN-B}, C_i]$ is larger than N . That is unreasonable in real-life dataset.

Therefore, As-ROCK approximate $link[C_{new}, C_i]$ by evaluating its value range: its lower limit $minLink$ equals the smaller one of $link[C_{NN-A}, C_i]$ and $link[C_{NN-B}, C_i]$, while its upper limit $maxLink$ equals the smaller of $(link[C_{NN-A}, C_i] + link[C_{NN-B}, C_i])$ and N . Thus $link[C_{new}, C_i] = (minLink + maxLink) / 2$. And $D(C_{new}, C_i)$ is computed according to equation 1.

The algorithm As-Rock stop automatically according to the outlier information, thus the parameter k is canceled. Fig. 3 is the overview of As-Rock.


```

Algorithm As-ROCK(  $\theta$  )
{ Dataset = read_data();
  compute_Neighbor(Dataset,  $\theta$ );
  compute_commonNeighbor(Dataset);
  Dataset = mine_outlier (Dataset);
  find  $C_{NN-A}$  and  $C_{NN-B}$  at present;
  while(  $D(C_{NN-A}, C_{NN-B}) > \theta$  and size(Dataset) > 1 )
  { merge  $C_{NN-A}, C_{NN-B}$  to get  $C_{New}$ ;
    compute  $D(C_{New}, C_i)$  for any remain cluster  $C_i$ ;
    delete( $C_{NN-A}, Dataset$ ); delete( $C_{NN-B}, Dataset$ );
    insert( $C_{New}, Dataset$ ); update  $C_{NN-A}$  and  $C_{NN-B}$ ;
  } //End of Clustering
} //End of As-ROCK

```

Fig. 3. The Auto-stopped Clustering Algorithm AS-ROCK

3 Discussion

The parameter θ of As-ROCK influences the outlier mining process and the clustering procedure. A method is proposed to determine the appropriate value range of θ based on following observation: Outliers are extremely dissimilar from the others. In mining the *common-neighbor based* outliers, it means that outliers have much smaller *link* with others, while *link* of normal data are much bigger. Therefore, after outliers are detected, the detected outlier number n_{out} will increase much faster with further increasing of θ . And the appropriate value range of θ should be the stable period of n_{out} before the greatly increase.

Since As-ROCK is an algorithm founded on the improved ROCK, it is necessary to analyze the change of complexity caused by improvements. The complexity of outlier mining composes of the computing of neighbor and common neighbor and outlier scanning step. And the scanning step costs $O(N)$. There is no additional computation cost to determine $D(C_{new}, C_i)$ in As-ROCK. The final clusters' number k is automatically determined and influences the computation complexity indirectly. It cost at most $O(2*N)$ to handle the real attribute. Take all these in total, the additional complexity cost of As-Rock is $O(3*N)$. Since it is at least $O(N^2*logN)$ for ROCK, the computation complexity of As-Rock equals that of ROCK.

4 Experiment and Analysis

The XML documents with 8 different kinds of DTDs^[6] and 2 real-life datasets Zoo and Vote from [8] are adopted to perform the experimental comparison with ROCK, DBScan and the algorithm of [2], which is nicknamed as Frozen algorithm.

To measure the clustering results' quality, two criterions *Entropy* and *Purity* of [9] are adopted:

$$Entropy = \sum_{i=1}^k \frac{n_i}{N} \left(-\frac{1}{\log q} \sum_{j=1}^q \frac{n_i^j}{n_i} \log \frac{n_i^j}{n_i} \right), \quad Purity = \sum_{i=1}^k \frac{1}{N} \max_j (n_i^j) \quad (3)$$

n_i^j is the number of data of j th original class assigned to the i th cluster. The better the clustering result, the smaller is the *Entropy* and the bigger is the *Purity*. For an ideal result, $Entropy=0.0$ and $Purity=1.0$.

In clustering XML dataset, the paper adopts same method of [4] to compute the structural distance. Before computing each data's neighbor, the distance is converted as equation (4) to satisfy the requirement that the similarity is less if data are similar:

$$Sim(T_1, T_2) = 1 - \frac{S(T_1, T_2)}{Max[S(T_i, T_j)]}, \quad S(T_1, T_2) = \frac{D(T_1, T_2)}{D_{max}(T_1, T_2)} \quad (4)$$

Where $Max[S(T_i, T_j)]$ equals to maximum distance among all XML data. And $D(T_1, T_2)$ is the tree edit distance and $D_{max}(T_1, T_2)$ be the maximum cost between the costs of all possible sequences of tree edit operations that transform T_1 to T_2 .

Table 2. The Overview of Clustering Results of As-Rock

Dataset	Parameter(s)	k	Entropy	Purity	n_{out}	Range of θ
Zoo	$\theta=0.86$	7	0.0717	0.9394	3	[0.82,0.86]
Vote	$\theta=0.67$	8	0.3679	0.9052	14	[0.57,0.68]
XML	$\theta=0.10$	11	0.1186	0.8843	2	[0.10,0.19]

Table 3. The Clustering Results Overview of ROCK, Frozen and DBScan

	Dataset	Parameter(s)	k	Entropy	Purity
ROCK	Zoo	$\theta=0.82, k=7$	--	0.1046	0.8812
	Vote	$\theta=0.73, k=8$	--	0.3988	0.8598
	XML	$\theta=0.70, k=14$	--	0.2014	0.8023
Frozen	Zoo	$\alpha=0.7$	10	0.2945	0.7822
	Vote	$\alpha=0.7$	21	0.5220	0.8230
	XML	$\alpha=1.5$	16	0.3041	0.7024
DBScan	Zoo	$E=1.5, MinPts=2$	10	0.0127	0.9901
	Vote	$E=2.5, MinPts=3$	2	0.5206	0.3862
	XML	$E=2.2, MinPts=3$	6	0.3014	0.7564

The clustering performance of As-Rock is listed in Table 2 with the appropriate value range of θ and the detected number of outlier etc.. Table 3 gives the best clustering results of other algorithms under all possible parameters' value. It can be seen that As-Rock is the best for almost each dataset. And the final clusters it obtained is also acceptable comparing to the original class number.

5 Conclusion

The paper proposes a new strategy that integrates outlier detection with clustering and proposes a new auto-stopped clustering algorithm As-Rock, which is constructed on a new outlier-detection method and the improved Rock algorithm. As-Rock needs only one parameter, whose value range can be automatically decided during outlier mining. The experimental results show its good performance for both XML dataset and other datasets.

Acknowledgements

This work is sponsored by the Natural Science Foundation of China under grant number 60373099 and the Natural Science Research Foundation of Harbin Engineering University under the grant number HEUFT05007.

References

1. Lee, M. L., Yang, L. H., Hsu, W., Yang, X.: XClust: Clustering XML Schemas for Effective Integration. Proc. 11th ACM Int. Conf. on Information and Knowledge Management (2002) 292-299
2. Ana L.N. Fred, José M.N. Leitão: A new Cluster Isolation criterion Based on Dissimilarity Increments. IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL.25, No. 8 (August 2003) 944-958
3. Dalamagas, T., et al., Clustering XML documents by structure, in Methods and Applications of Artificial Intelligence, Proceedings. (2004) 112-121.
4. Termier, A., Rouster, M. C., Sebag, M.: TreeFinder: A First Step towards XML Data Mining. IEEE Int. Conf. on Data Mining (ICDM) (2002) 450-457
5. S. Guha, R. Rastogi, K. Shim: ROCK: a robust clustering algorithm for categorical attributes. In: Proc. of the 15th Int'l Conf. on Data Eng (1999) 512-521
6. <http://www.cs.wisc.edu/niagara/data.html>
7. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining, Portland, OR (1996) 226-231
8. Hettich, S. & Blake, C.L. & Merz, C.J. (1998). UCI Repository of machine learning databases [<http://www.ics.uci.edu/~mllearn/MLRepository.html>]. Irvine, CA: University of California, Department of Information and Computer Science.
9. Ying Zhao, George Karypis: Criterion Functions for Document Clustering: Experiment and Analysis. Technical Report #01-40, University of Minnesota (2001) 1 – 40

A Generalized Global Convergence Theory of Projection-Type Neural Networks for Optimization*

Rui Zhang^{1,2} and Zongben Xu¹

¹ Institute for Information and System Science,
Xi'an Jiaotong University, Xi'an 710049, China
zbxu@mail.xjtu.edu.cn

² Department of Mathematics, Northwest University, Xi'an 710069, China

Abstract. The projection-type neural networks $\tau \frac{dx}{dt} = -x + P_{\Omega}(x - \Lambda(t)\partial^0 E(x))$ are generic and useful models for solving the constrained optimization problems $\min\{E(x)|x \in \Omega\}$. In the existing convergence/stability analysis, the most are deduced based on the assumptions that E is uniformly or strictly convex and Ω is box-shaped. In this talk we present a generalized theory on convergence/stability of the networks. In the general setting that E is only convex and Ω is any closed bounded convex set, it is shown the global convergence/asymptotic stability of the networks in a specified sense. The presented theory sharpens and generalizes the existing results, and, consequently, underlies the applicability of the neural networks for a broader type of optimization problems.

1 Introduction

Optimization problems commonly arise in many fields of science and engineering. In applications, fast, or more preferably, real-time solution for those optimization problems is often required [1]. The conventional serial-processing optimization algorithms may not be able to meet such requirement, the artificial neural network algorithms are, however, considered to be potentially useful alternatives for the purpose.

Since Hopfield and Tank ([2]-[4]) first demonstrated feasibility of solving a class of optimization problems with neural networks in 1980's, neural network approaches for various optimization problems have attracted great interests ([5]-[11]). Many different neural network models have been developed for linear programming, quadratic programming, convex programming and specific combinatorial optimization problems. In the existing studies, two approaches are generally adopted : the gradient-descent based approach and the projection based approach. In the first approach, an optimization problem which is, in most cases, constrained one, is first transformed into an unconstrained optimization problem by virtue of using the techniques such as penalty function, optimality conditions,

* This research was supported by the NSF project under contract No.10371097.

duality theory, or Lagrange multiplier, and then the unconstrained optimization problem is solved by applying the gradient flow equation (which corresponds to the neural network model adopted). The projection based approach designs the neural network model based on characterizing the considered optimization problem in terms of variational inequality conditions, and then transforming the variational inequality conditions into a projection operator equation. A dynamical system directly solving the projection operator equation is taken as the neural network model for solving the original optimization problem. In this presentation, we are concerned with the projection based approach only.

A generic projection-type neural network can be mathematically described as

$$\tau \frac{dx}{dt} = -x + P_{\Omega}(x - \Lambda(t)A(x)) \quad (1)$$

where τ , $\Lambda(t) = \text{diag}\{\alpha_1(t), \dots, \alpha_n(t)\}$ are adjustable positive parameters, P_{Ω} is the nearest point projection of R^n onto the subset Ω , and $A : R^n \rightarrow R^n$ is an operator associated tightly with the gradient of objective function. When the constrained optimization problem

$$(\mathbf{COP}) : \min\{E(x)|x \in \Omega\} \quad (2)$$

is considered, for instance, A may take as $\nabla E(\cdot)$, $\partial^0 E(\cdot)$ or $P(\cdot)$ respectively, where $\partial^0 E(\cdot)$ is a single-valued selection of subdifferential of E , and $P(\cdot)$ is the feasible direction operator. It is well known that asymptotic stability of the neural network (1) is a prerequisite of using the network as a solver of the optimization problem. Therefore, many studies on stability of the projection-type neural networks have been conducted ([8]-[12]). The basic results derived from the existing studies can be summarized as follows: (i) the system (1) will approach to a stable equilibrium point set when A is monotone, and in this case, there is a subsequence of the trajectory $x(t, x_0)$ of (1) that converges to an equilibrium point of (1); (ii) the system (1) has a unique equilibrium point that is asymptotically stable when A is strictly monotone; and (iii) the system (1) has a unique equilibrium point that is exponentially stable when A is uniformly monotone. If $E(x)$ is differentiable, these conclusions can be equivalently stated as (through taking $A(x) = \nabla E(x)$): (i) the system will approach to the equilibrium point set of (1) when E is convex; (ii) the system has a unique equilibrium point that is asymptotically stable when E is strictly convex; and (iii) the system has a unique equilibrium point that is exponentially stable when E is uniformly convex. From these studies, we can see that up to now, only clear is that the projection-type neural networks can be assuredly used to finding the unique solution of (COP), and it is still open whether or not the projection-type neural networks can be applied to a broader class of functions beyond strictly convex functions, particularly in the sense of having guaranteed global convergence property. The aim of the present research is to make this issue clear. More specifically, through exploring a novel property of subdifferential of a convex function, and by a deep analysis on trajectory of the system (1), we prove that, starting from any initial point in Ω , the neural network (1) can assuredly converge to

an optimal solution of **(COP)** whenever E is convex and twice differentiable. The obtained result provides an affirmative answer to the stated open question, and underlies the applicability of the projection-type neural networks to broader class of functions (particularly to the class of convex functions).

In section 2 we will formally state and prove the generalized global convergence theory on the system (1). And then, in section 3, we specialize the developed theory to several particular cases, demonstrating the generality and powerfulness of the developed theory. We finally conclude the work in section 4 with some useful remarks.

2 The Generalized Global Convergence Theory

In this section we present a generalized theory on global convergence/ asymptotic stability of the projection-type neural networks. We start with reviewing some necessary knowledge on **(COP)** and the model (1).

When E is a proper lower semi-continuous (l.s.c.) convex function, it is known that the subdifferential set $\partial E(x)$ is nonempty for any $x \in R^n$. Denote by $\partial^0 E(x)$ any single-valued selection of $\partial E(x)$ below. We will consider the following specific projection-type neural networks

$$\tau \frac{dx}{dt} = -x + P_\Omega(x - \Lambda(t)\partial^0 E(x)). \tag{3}$$

In the following, we will assume that Ω is a closed convex subset of R^n which guarantees that the nearest point projection P_Ω is well defined. In the special case of Ω being a box-shaped, that is, $\Omega = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_n, b_n]$ with a set of real numbers $a_i \leq b_i$ ($i = 1, 2, \dots, n$), we easily see that P_Ω can be specified componentwisely by

$$P_\Omega(x) = (g_1(x_1), g_2(x_2), \dots, g_n(x_n))^T \tag{4}$$

where

$$g_i(s) = \begin{cases} a_i, & \text{when } s \in (-\infty, a_i) \\ s, & \text{when } s \in [a_i, b_i] \\ b_i, & \text{when } s \in (b_i, +\infty) \end{cases} . \tag{5}$$

Likewise, when $\Omega = R_+^n := \{x = (x_1, x_2, \dots, x_n)^T : x_i \geq 0, i = 1, 2, \dots, n\}$, we can check that

$$P_\Omega(x) = \frac{1}{2}(x + |x|) \tag{6}$$

where $|x| = (|x_1|, |x_2|, \dots, |x_n|)^T$. In Eq.(3), the parameter $\Lambda(t)$ is always assumed to be of the form $\alpha(t)I$ whenever P_Ω is nonseasonal (namely, it can not be defined componentwisely), otherwise it is positively diagonal. In any case, however, we assume

$$\inf_{t \geq 0} \left\{ \alpha(t), \inf_{1 \leq i \leq n} \{ \alpha_i(t) \} \right\} = \alpha > 0. \tag{7}$$

With the above notion and notations, we now establish several useful lemmas that are needed for proof of our main theorem. The proof of the lemmas will be omitted due to the limitation of the space.

Lemma 1. The set Ω is invariant under the system (3), i.e., the trajectory $x(t, x_0)$ of the system (3) is in Ω whenever $x_0 \in \Omega$.

Lemma 2. Restricted to the set Ω , the system (3) is a strict generalized gradient system. In particular, $E(x)$ is a strict energy function of the system (3) on Ω .

Lemma 3. If E is a twice differentiable and convex function, then $\partial^0 E(\cdot)$ is uniformly anti-monotone on Ω in the following sense:

$$\langle \partial^0 E(x) - \partial^0 E(y), x - y \rangle \geq \frac{1}{2} \|\partial^0 E(x) - \partial^0 E(y)\|^2, \forall x, y \in \Omega. \tag{8}$$

Remark 1. Let $\Lambda = \text{diag}\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ be any positive definite matrix. Under the assumptions of Lemma 3, we can further show that

$$\langle \partial^0 E(x) - \partial^0 E(y), \Lambda(x - y) \rangle \geq \frac{1}{2} \|\partial^0 E(x) - \partial^0 E(y)\|^2, \forall x, y \in \Omega. \tag{9}$$

Actually, introducing a new norm defined by $\|x\|_* = \|\Lambda^{1/2}x\|$, we can easily prove the conclusion completely as that in the proof of Lemma 3.

We now prove our main theorem.

Theorem 1. Assume that E is a twice differentiable and convex function, $\Omega_0 \subset \Omega$ be a bounded subset containing the trajectory $x(t, x_0)$ of (3). Assume further that $\Lambda(t) = \text{diag}\{\alpha_1(t), \alpha_2(t), \dots, \alpha_n(t)\}$ satisfies $\alpha_i(t) < \frac{\sqrt{2}}{2}, i = 1, 2, \dots, n$. Then the trajectory $x(t, x_0)$ converges to an optimal solution of (COP) starting from any point x_0 in Ω .

Proof. Let $F(x) = -x + P_\Omega(x - \Lambda\partial^0 E(x))$ and $G(x) = x - \Lambda\partial^0 E(x)$. For any fixed $x^* \in F^{-1}(0)$ (therefore, $x^* = P_\Omega(G(x^*))$), define

$$V(x, x^*) = \frac{1}{2}(x - x^*)^T(x - x^* + 2\Lambda\partial^0 E(x^*)) - \Lambda E(x). \tag{10}$$

Clearly $\partial^0 V(x, x^*) = x - x^* + \Lambda\partial^0 E(x^*) - \Lambda\partial^0 E(x) = G(x) - G(x^*)$, and consequently,

$$\begin{aligned} \tau \frac{dV}{dt} &= \langle G(x) - G(x^*), F(x) \rangle \\ &= -\langle G(x) - G(x^*), x - x^* \rangle + \langle G(x) - G(x^*), P_\Omega(G(x)) - P_\Omega(G(x^*)) \rangle. \end{aligned} \tag{11}$$

By Lemma 3, we have

$$\begin{aligned} &\langle G(x) - G(x^*), x - G(x) - x^* + G(x^*) \rangle \\ &= \langle G(x) - G(x^*), \Lambda[\partial^0 E(x) - \partial^0 E(x^*)] \rangle \\ &= \langle \Lambda[G(x) - G(x^*)], \partial^0 E(x) - \partial^0 E(x^*) \rangle \\ &= \langle \Lambda[x - x^*] - \Lambda[\Lambda\partial^0 E(x) - \Lambda\partial^0 E(x^*)], \partial^0 E(x) - \partial^0 E(x^*) \rangle \end{aligned}$$

$$\begin{aligned}
 &= \langle \Lambda[x - x^*], \partial^0 E(x) - \partial^0 E(x^*) \rangle - \|\Lambda[\partial^0 E(x) - \partial^0 E(x^*)]\|^2 \\
 &\geq \frac{1}{2} \|\partial^0 E(x) - \partial^0 E(x^*)\|^2 - \|\Lambda[\partial^0 E(x) - \partial^0 E(x^*)]\|^2 \\
 &\geq [\frac{1}{2} - \max_{1 \leq i \leq n} \{\alpha_i^2\}] \|\partial^0 E(x) - \partial^0 E(x^*)\|^2 \geq 0.
 \end{aligned}$$

Combined with (11), it thus follows that

$$\begin{aligned}
 \tau \frac{dV}{dt} &\leq -\langle G(x) - G(x^*), G(x) - G(x^*) \rangle + \\
 &\quad \langle G(x) - G(x^*), P_\Omega(G(x)) - P_\Omega(G(x^*)) \rangle \\
 &\leq -\|G(x) - G(x^*)\|^2 + \|G(x) - G(x^*)\| \|P_\Omega(G(x)) - P_\Omega(G(x^*))\| \\
 &\leq 0
 \end{aligned}$$

where the last inequality is deduced from the nonexpansiveness of the nearest point projection P_Ω . This shows that $V(x(t, x_0), x^*)$ is monotonically decreasing in t and hence, the limit $\lim_{t \rightarrow \infty} V(x(t, x_0), x^*)$ exists. Let

$$V_1(x, x^*) = \frac{1}{2}(x - x^*)^T(x - x^* + 2\Lambda\partial^0 E(x^*)). \tag{12}$$

Then $V(x(t, x_0), x^*) = V_1(x(t, x_0), x^*) - \Lambda E(x(t, x_0))$. By Lemma 2, it is known that the limit $\lim_{t \rightarrow \infty} E(x(t, x_0))$ exists. So does $\lim_{t \rightarrow \infty} V_1(x(t, x_0), x^*)$.

By assumption, all $x(t, x_0)$ are included in a bounded set Ω_0 . So the ω -limit set $\omega(x(t, x_0))$ is nonempty. Furthermore, from Lemma 2, $\omega(\Gamma(x_0)) \cap F^{-1}(0)$ is also nonempty. Let us fix an $x^* \in \omega(\Gamma(x_0)) \cap F^{-1}(0)$. Then there is a sequence $\{t_n\} \subset (0, \infty)$ such that $\lim_{n \rightarrow \infty} x(t_n, x_0) = x^*$. Thus we can conclude that $\lim_{t \rightarrow \infty} V_1(x(t, x_0), x^*) = 0$. Since x^* is also a solution of the variational inequality

$$\partial^0 E(x^*)^T(x - x^*) \geq 0, \forall x \in \Omega, \tag{13}$$

there then holds the following

$$V_1(x(t, x_0), x^*) \geq \frac{1}{2} \|x(t, x_0) - x^*\|^2. \tag{14}$$

Consequently, we deduce that $\lim_{t \rightarrow \infty} x(t, x_0) = x^*$ and Theorem 1 follows. This finishes the proof of Theorem 1.

3 Specification

In this section we specialize the generic theorem proven in the last section to several important concrete cases, demonstrating the generality and powerfulness of the theorem. Due to the limitation of space, all the details of the proof for the following specification will be omitted.

3.1 Bounded Constrained Optimization Problem

Consider the **(COP)** with the constraint set Ω defined by the box

$$B = \{x = (x_1, x_2, \dots, x_n)^T \in R^n : a_i \leq x_i \leq b_i, i = 1, 2, \dots, n\}. \tag{15}$$

The projection $P_\Omega := P_B$ is then defined by (4) in this case. Specializing Theorem 1 to this particular case leads to the following corollary.

Corollary 1. If E is convex and twice differentiable on B , $x(t, x_0)$ is the unique trajectory of the projection neural network

$$\tau \frac{dx}{dt} = -x + P_B(x - \Lambda \nabla E(x)), x(0) = x_0, \tag{16}$$

then $x(t, x_0)$ converges to an optimal solution of **(COP)** from any x_0 in B .

Remark 2. Xia and Wang [8][9] recently considered the neural network

$$\frac{dx}{dt} = A\{-x + P_B(x - A(x))\}, x(0) = x_0 \tag{17}$$

where $A : R^n \rightarrow R^n$ and they proved that if A is monotone and continuously differentiable, the trajectory $x(t, x_0)$ of (17) will converge to an equilibrium point x^* of (17) from any x_0 in B in the sense that

$$\inf_{t \rightarrow \infty} \|x(t, x_0) - x^*\| = 0. \tag{18}$$

Notice that $A = \nabla E(x)$ is monotone and differentiable if and only if E is convex and twice differentiable, and $\inf \lim_{t \rightarrow \infty} \|x(t, x_0) - x^*\| = 0$ if and only if $x(t, x_0)$ has a subsequence that converges to x^* (in other words, $x(t, x_0)$ itself does not necessarily converge). So Corollary 1 sharpens Xia and Wang’s result in the sense that when applied to convex programming, the “weak” global convergence of their result is upgraded to global convergence in normal sense.

3.2 Linear Constrained Convex Programming Problem

Consider the following **(COP)** with linear constraints

$$\min f(x) \text{ subject to } Ax = b, x \geq 0 \tag{19}$$

where $f(x)$ is continuously differentiable and convex from R^n to R , $A \in R^{m \times n}$, and $b \in R^m$. It is known (cf., e.g.,[10]) that x^* is a solution of (19) if and only if there exists $y^* \in R^m$ such that (x^*, y^*) satisfies

$$\begin{cases} (x - \alpha \nabla f(x) + \alpha A^T y)^+ - x = 0 \\ Ax - b = 0 \end{cases} \tag{20}$$

where α is a positive constant, $(x)^+ = [(x_1)^+, \dots, (x_n)^+]^T$, and $(x_i)^+ = \max\{0, x_i\}$. So the problem (19) can be solved by means of the neural network

$$\frac{d}{dt} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} -x + (x - \alpha \nabla f(x) + \alpha A^T y)^+ \\ \alpha(Ax - b) \end{pmatrix} \tag{21}$$

with any positive scaling parameter λ . Let $z = (x, y)^T \in R^n \times R^m$. Define

$$\Omega = \{z = (x, y)^T : x \in R_+^n, y \in R^m\} = R_+^n \times R^m \tag{22}$$

and

$$P_\Omega(z) = P_\Omega \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} P_+(x) \\ y \end{pmatrix}, \forall z \in R^n \times R^m \tag{23}$$

where P_+ is the nearest point projection from R^n to R_+^n . We easily show that the so defined operator P_Ω is a nearest point projection of $R^n \times R^m$ onto Ω , and in light of such defined projection, we can rewrite the neural network (21) into

$$\begin{aligned} \frac{dz}{dt} &= \frac{d}{dt} \begin{pmatrix} x \\ y \end{pmatrix} \\ &= \lambda \left\{ - \begin{pmatrix} x \\ y \end{pmatrix} + P_\Omega \begin{pmatrix} x - \alpha \nabla f(x) + \alpha A^T y \\ y - \alpha(-Ax + b) \end{pmatrix} \right\} \\ &= \lambda \{-z + P_\Omega(z - \alpha \nabla L(z))\} \end{aligned} \tag{24}$$

where

$$L(z) = L(x, y) = f(x) - y^T(Ax - b), \forall z = (x, y)^T \in R^n \times R^m. \tag{25}$$

Now, applying Theorem 1, we can get the following corollary.

Corollary 2. Assume that $x(t, x_0, y_0)$ is the unique solution of (21) and the level set $L_0 = \{(x, y) : f(x) - y^T(Ax - b) \leq f(x_0) - y_0^T(Ax_0 - b)\}$ is bounded. If $f(x)$ is convex and twice differentiable, then $x(t, x_0, y_0)$ converges to an optimal solution of **(COP)** from any (x_0, y_0) in $R_+^n \times R^m$.

Remark 3. To solve the problem (19), Xia and Wang ([10]) proposed the subsequent neural network

$$\frac{d}{dt} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} -x + (x - \alpha \nabla f(x) + \alpha A^T y)^+ \\ \alpha(-Ax + b) \end{pmatrix}. \tag{26}$$

They proved that the network (26) is globally convergent to the stationary point $z^* = (x^*, y^*)^T$ of the system (26), starting from any initial point $(x_0, y_0) \in R_+^n \times R^m$, whenever $f(x)$ is strictly convex and twice differentiable. His model differs from our model (21) only in the second equation of the system (actually only differs in the sign). However, this difference makes it possible for us to justify the very desirable property — global convergence of the model without restriction of strict convexity of f , while Xia and Wang claimed the global convergence of their model with the restriction. This shows potential of the model (21) and generality of Theorem 1 presented in the present talk.

4 Conclusion

Under the general assumptions on convexity of f and closeness and convexity of Ω , we have established a generalized theory of global convergence/asymptotic

stability of the projection-type neural networks (1) for constrained optimization problems (**COP**). It is developed mainly by making a novel use of dynamic system analysis and application of a special type of monotone operators. The main value of the theory consists in showing that the projection-type neural networks can assuredly converge to an optimal solution of the optimization problem from any point x_0 in Ω , as long as the objective function f is twice differentiable and convex. Thus, we conclude that the projection-type neural networks can be applied to solving the broader class of constrained optimization problems, not necessarily restricted to the strictly convex functions. The obtained results extend and sharpen the existing knowledge on applicability of the projection-type neural networks.

There are many problems deserving further research. Some problems, for instance, are: how to develop a more generalized global convergence/ asymptotic stability theory for the system (1) when $A(x)$ takes as a feasible direction operator, and if the projection-type neural networks can be used as a unified model to unify the existing neural network models. All these problems are under our current investigation.

References

1. T. Yoshikawa: Foundations of Robotics: Analysis and Control. MIT Press, Cambridge, MA, (1990)
2. J.J. Hopfield: Neural networks and physical systems with emergent collective computational ability. Proc. Natl. Acad. Sci. USA. **79** (1982) 2554-2558
3. J.J. Hopfield, D.W.Tank: Neural computation of decisions in optimization problems. Biological Cybernetics. **52** (1985) 141-152
4. D.W.Tank, J.J. Hopfield: Simple neural optimization networks: An A/D converter, signal decision circuit, and a linear programming circuit. IEEE Transactions on Circuits and Systems. **33** (1986) 533-541
5. M.P. Kennedy, L.O. Chua: Neural networks for nonlinear programming. IEEE Transactions on Circuits and Systems. **35** (1988) 554-562
6. A. Bouzerdoum, T.R. Pattison: Neural network for quadratic optimization with bound constraints. IEEE Transactions on Neural Networks. **4** (1993) 293-304
7. X.B.Liang, J.Wang: A recurrent neural network for nonlinear optimization with a continuously differentiable objective function and bound constraints. IEEE Transactions on Neural Networks. **11** (2000) 1251-1262
8. Y.S.Xia, H. Leung, J.Wang: A projection neural network and its application to constrained optimization problems. IEEE Transactions on Circuits and Systems. **49** (2002) 447-458,
9. Y.S.Xia, J.Wang: A general projection neural network for solving monotone variational inequalities and related optimization problems. IEEE Transactions on Neural Networks. **15** (2004) 318-328
10. Y.S.Xia, J.Wang: A recurrent neural network for solving nonlinear convex programs subject to linear constraints. IEEE Transactions on Neural Networks. **16** (2005) 379-386
11. Z.B.Xu, C.P.Kwong: Associative Memories, in "Neural Networks Systems Techniques and Applications" (ed. by C. T. Leondes). Academic Press, New York. **3** (1998) 183-258

Hierarchical Recognition of English Calling Card by Using Multiresolution Images and Enhanced Neural Network

Kwang-Baek Kim¹ and Sungshin Kim²

¹ Department of Computer Eng.,
Silla University, Busan 617-736, Korea
gbkim@silla.ac.kr

² School of Electrical and Computer Eng.,
Pusan National University, Busan, Korea
sskim@pusan.ac.kr

Abstract. In this paper, we proposed a novel hierarchical algorithm to recognize English calling cards. The algorithm processes multiresolution images of calling cards hierarchically to extract characters and recognize the characters by using an enhanced neural network method. Each processing step functions at lower overhead and results improved output. That is, first, horizontal smearing is applied to a 1/3 resolution image in order to extract the areas that only include characters from the calling card image. Second vertical smearing and the contour tracking masking, is applied to a 1/2 resolution image in order to extract individual characters from the character string areas. And last, the original image is used in the recognition step, because the image accurately includes the morphological information of the characters accurately. To recognize characters with diverse font types and sizes, the enhanced RBF network that improves the middle layer based on the ART1 was used. The results of experiments on a large number of calling card images showed that the proposed algorithm greatly improves the character extraction and recognition compared with traditional recognition algorithms.

1 Introduction

Recently, mobile devices such as PDAs, mobile phones, and other electronic communications devices often equipped with camera sensors, and various input devices, which can scan documents via photo sensor. Consequently many advancement in image processing by the scanning functions of mobile devices have been made. Work on calling card recognition by mobile devices is actively being pursued. These works scan calling cards via photo sensor and then recognize and digitalizes personal information.

Generally, the recognition of calling cards consists of two phases; the extraction of significant information from scanned images the extraction phase and the recognizing the resulting information the recognition phase [1]. The extraction phase includes preprocessing to remove noise, the extraction of character strings

and the extraction of individual characters. The recognition phase encodes the extracted characters by using various recognition methods.

This paper proposes a new character extraction method using multiresolution images, which quickly and precisely extracts individual characters without interference from noise or the skew and arrangement of character strings by using smearing and contour tracking algorithms on the original image and images with lower resolution. An enhanced RBF(Radial Basis Function) network that is robust to noises and quickly adapts to new patterns is also proposed. The network was used to recognize individual characters extracted from calling card images.

2 Related Works

Previous works have used a variety of methods to extract characters from calling card images, such as contour tracking algorithms [2], [3], and smearing algorithms [4], etc.

The contour tracking algorithm extracts individual characters by using a mask pattern designed for contour tracking and selects true characters by considering the extracted areas that have more black pixels than the given threshold. While this algorithm precisely extracts characters without interference from noises, skew or arrangement, it has a higher processing overhead and may incorrectly recognize characters like 'i' because the dot is treated as noise.

The extraction methods that use smearing algorithms [4] can be classified as RLSA(Run Length Smoothing Algorithm) and RXYC(Recursive X-Y Cuts) methods. The RLSA method replaces the white pixels between two black pixels with black ones when the number of white pixels is not greater than a given threshold. For example, if the black pixels, white pixels and the threshold are 1, 0 and 3, respectively, the output of the RLSA method for the pixel sequence, "10001100000110010001", becomes "1111110000011111 00001". If the RLSA method is used to input an image vertically and horizontally and the two output images are combined by the logical operand, AND, the original image becomes blocked by areas having meaningful information. The RXYC method scans input images in vertical and horizontal directions and cuts the area having more black pixels than a given threshold in each scan direction. The RXYC method is efficient for extracting large blocks, while the RLSA method is efficient for extracting small blocks. However, the RXYC method may incorrectly extract blocks under noisy conditions.

A variety of methods have been proposed for recognizing individual characters such as template matching method, statistical method and neural network method [5],[6]. One is the RBF network, which uses RBF to perform multivariate analysis and statistical interpolation to structure neural network model [7]. An RBF network consists of three layers, (input, middle and output) a clustering algorithm such as SOM (Self Organizing Map) and applying FCM(Fuzzy C-Means) to the middle layer. This paper proposes an enhanced RBF network, which uses an enhanced ART1 in the learning structure between the input and middle layers, and uses it to recognize the individual characters of calling cards.

3 Recognition of Calling Card by Using Multiresolution Images and Enhanced RBF Network

3.1 The Proposed Recognition System for Calling Cards

This system consists of the extraction phase and the recognition phase. In the extraction phase a significant amount of information is derived from scanned images. In the recognition phase, the extracted information and the general structure of the calling card are recognized. Fig. 1(a) illustrates the system.

In this paper, multi-resolution images are used to extract significant information from calling card images. Images with 1/3 times resolution are used in the extraction of character strings; images with half resolution for individual characters, and the original image for character recognition. The extraction of blocking areas and individual characters from low-resolution images reduces the processing time and it is implement ably in parallel processing in hardware. Also, contour tracking and smearing algorithms are used to prevent interference by skew, character arrangement and image noise for a robust extraction process. The whole structure of proposed recognition system is based on multi-resolution images is like the one in Fig. 1(b).

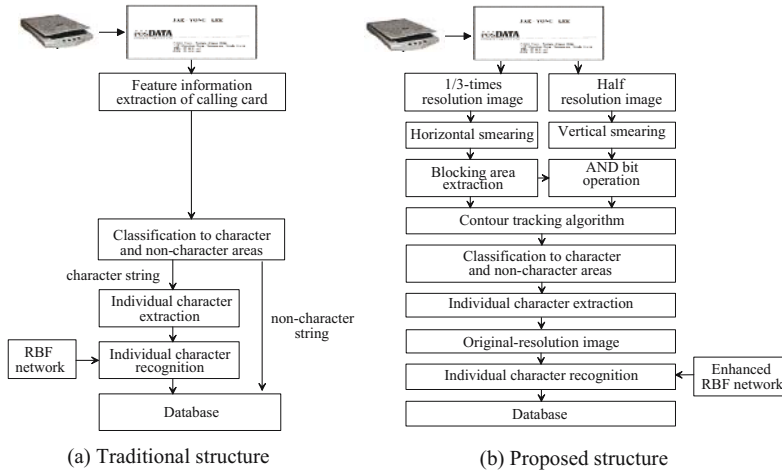


Fig. 1. The structure of calling card recognition system

3.2 Image Preprocessing and Blocking Area Extraction

The proposed method eliminates noise from the input images and extracts blocking areas including picture, logo and character strings from the preprocessed image. In the preprocessing step, the input image is binarized to precisely extract individual characters and the binary image is down-scaled to two low-resolution images. Image binarization and down-scaling implicitly remove noise from the original image.

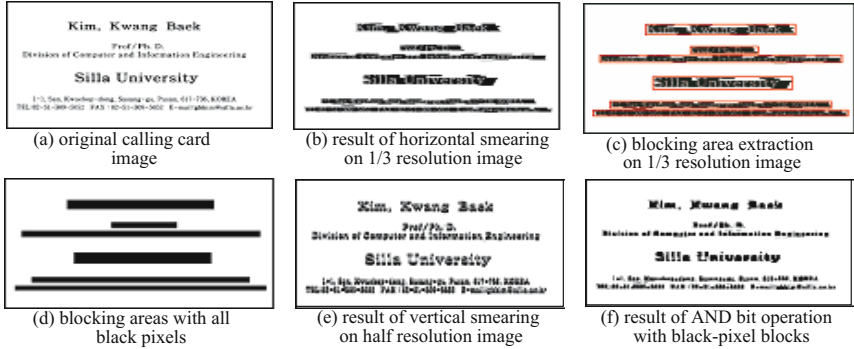


Fig. 2. The extraction procedure of character strings and individual characters

In order to quickly extract blocking areas, the proposed method uses a smearing algorithm on the 1/3 resolution binary image and extracts blocking areas from the results by using the calling card feature of arranging characters of the same size at regular intervals on horizontal lines. The smearing algorithm expands black pixels in the direction in the image and removes the space between characters if it is below a given threshold. As individual characters are arranged in horizontal lines, horizontal smearing connects adjacent characters and generates a blocking area that includes a row of characters. Fig. 2(b) shows the results generated by applying the horizontal smearing algorithm to a 1/3 resolution image of calling card. This paper locates the boundaries of rectangular blocking areas and saves their coordinates.

3.3 Individual Character Extraction

This method uses 1/2 resolution images for individual character extraction to reduce processing time and improve accuracy. In image with half resolution, vertical smearing algorithm is applied to areas corresponding to blocking areas previously extracted, and the AND bit operation is executed between the result of vertical smearing and blocking areas with all black pixels. Fig. 2(c)~2(f) show results of vertical smearing and AND bit operation in the half-resolution image.

Individual characters are able to be extracted by applying contour tracking algorithm to blocking areas generated by a sequence of pixel operations in the half-resolution image. Generally, contour tracking algorithm extracts accurately individual characters along with its shapes. But, in the case that a character like 'i' is separated to two parts, contour tracking algorithm may process small part of the character as noise and remove it. So, this paper makes characters like 'i' in to connected areas by using vertical smearing algorithm and AND bit operation with black-pixel block. Fig. 3(a) shows that previous extraction method fails to extract characters with separated parts like 'i' and Fig. 3(b) shows the proposed extraction algorithm.

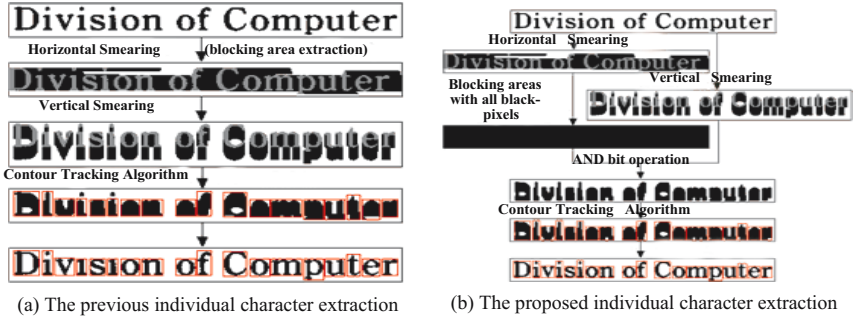


Fig. 3. Comparison of individual character extraction methods

3.4 Character Recognition Using an Enhanced RBF Network

The middle layer of an RBF network clusters the learning vectors by grouping homogenous vectors in the same cluster. In the clustering procedure, the homogeneity between learning vectors is represented as the distance between the vectors. Therefore, if the distances between a learning vector and all vectors in a cluster are smaller than a given constant radius, the learning vector is added to the cluster. However, the usage of a constant radius in clustering is the primary source of errors and therefore decreases the recognition success rate [8]. To improve the recognition success rate, this paper proposes an enhanced RBF neural networks that adapts the ART1 network to the learning structure between the input layer and the middle layer and applies the output layer of the ART1 network to the middle layer.

In the ART1 network, a vigilance parameter inversely determines the degree of mismatch between any input pattern and saved patterns that is allowed [9]. A high vigilance threshold classifies an input pattern as a new category in spite of a few mismatches between the pattern and the expected patterns, and conversely a low threshold value may allow an input pattern to be classified with an existing cluster in spite of a large number of mismatches. However, because the ART1 network bases its vigilance threshold on empirically derived values, the recognition success rate is declines. To correct this defect, the ART1 network has been enhanced by using Yager’s intersection operator, a fuzzy connection operator, to determine the degree of homogeneity between patterns and dynamically adjust the vigilance threshold. Eq. (1) shows the equation applied to the ART1 network in this paper, which dynamically adjusted the vigilance threshold, ρ , by using Yager’s intersection operator [10].

$$\rho(n + 1) = 1 - \min \left(1, \sqrt{(1 - \rho(n))^2 + (1 - \rho(n - 1))^2} \right) \quad (1)$$

This paper enhances the RBF network by applying the modified ART1 network to the middle layer, as shown in Fig. 4. And we call the modified RBF network as an enhanced RBF neural network.

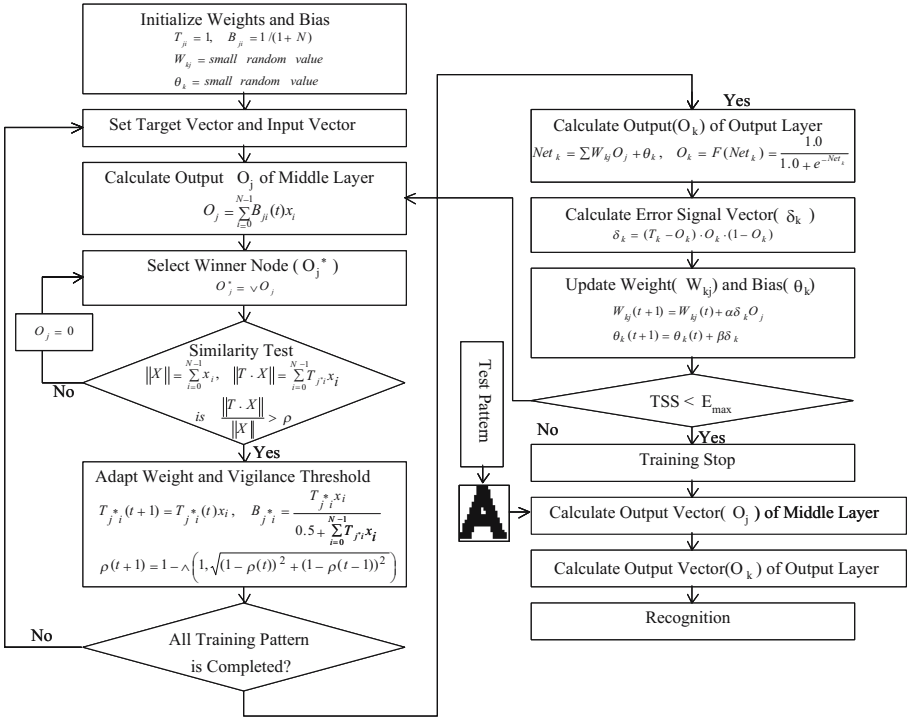


Fig. 4. The learning and recognition algorithm of the enhanced RBF network

4 Performance Evaluations

The proposed recognition system was implemented using C++ Builder on a Pentium-IV PC and performed experiments on 50's calling card images with 1500x800 pixel size. To evaluate the performance of the proposed extraction algorithm, Table 1 compares the number of blocking areas and individual characters extracted by the proposed extraction algorithms in [1] and those from the proposed method. The last column of Table 1 presents the number of characters successfully and unsuccessfully recognized by the proposed enhanced RBF network on all characters extracted by two algorithms.

The first row of Table 1 indicates the total number of blocking areas and individual characters included in the 50 calling card images. The second row shows the number of blocking areas and individual characters extracted by the proposed method in [1] and the number of successfully recognized characters out of the extracted characters. The fourth row shows the experimental results by the proposed method in this paper. As shown in Table 1, the proposed extraction method improves the success rate for individual character extraction as compared to the result in [1].

To evaluate the performance of the enhanced RBF network, the proposed enhanced ART1 and the normal ART1 were used on the middle layer of the

Table 1. The extraction result of blocking areas and individual characters with respect to extraction methods

	Blocking area	Character area	Character recognition
Total number of areas	629	7641	7641
# of of extraction success by Ref.[1]	627	7073	6812
# of of extraction failure by Ref.[1]	2	568	829
# of of extraction success by the proposed method	627	7495	7386
# of of extraction failure by the proposed method	2	146	255

Table 2. The results of learning and recognition using the enhanced RBF network

	# of hidden layer's nodes (# of nodes / # of input patterns)	Recognition performance (# of successes / # of input patterns)
The proposed enhanced RBF network		
Alphabetic char's	380 / 520	3698 / 3723
Numeric char's	121 / 200	2782 / 2821
Special char's	16 / 30	906 / 951
The RBF network based on normal ART1		
Alphabetic char's	498 / 520	3682 / 3723
Numeric char's	175 / 200	2773 / 2821
Special char's	21 / 30	889 / 951

RBF network. Table 2 compares learning and recognition results using the two algorithms on previously extracted individual characters.

For experiments presented in Table 2, all individual characters are separated into three character sets; alphabetic characters, numeric characters and special characters, and the recognition experiments were performed on each character set separately. The initial vigilance parameter in the RBF network based on normal ART1 was fixed to 0.9, and in the enhanced RBF network, the vigilance parameter was initially set at 0.9 and adapted dynamically. For the 520 alphabetic character input patterns, the enhanced RBF network and the RBF network based on the normal ART1 created 380 and 496 middle nodes, respectively, demonstration that the enhanced RBF network is more efficient at learning individual characters in calling card images. By comparing the number of recognition successes between the two algorithms in Table 2, the enhanced RBF network performed better than the RBF network based on normal ART1.

5 Conclusions

This paper proposes a new character extraction algorithm based on multiresolution images that reduces preprocessing time and accuracy in the recognition

of calling cards. The proposed algorithm uses 1/3 resolution images for blocking area extraction, half-resolution images for individual character extraction, and the original images for the recognition of individual characters. For accurate character extraction based on arrangement patterns of character strings and character shapes, the proposed algorithm uses horizontal and vertical smearings combined with AND bit operation with block-pixel blocks and contour tracking algorithms in the extraction of blocking areas and individual characters. To improve the recognition rate for individual characters, this paper proposes an enhanced RBF network that dynamically adapts its vigilance parameters. Character extraction and recognition experiments were performed on over 50 calling card images to evaluate performance. The results showed that the proposed extraction algorithm and the enhanced RBF network perform better than previous algorithms used in calling card recognition.

References

1. Kim, K. B., Kim, C. K., and Kim, J. W.: A study on The Recognition of an English Calling Card by using Contour Tracking Algorithm and Enhanced ART1. *Journal of Intelligent Information Systems*. 8(2) (2002) 105–115
2. Kim, K. B., Jang, S. W., and Kim, C. K.: Recognition of Car License Plate by Using Dynamical Thresholding and Enhanced Neural Networks. *Lecture Notes in Computer Science*. LNCS 2756. Springer-Verlag (2003) 309–319
3. Kim, K. B., Kim, Y. j.: Recognition of English Calling Cards by Using Enhanced Fuzzy Radial Basis Function Neural Networks. *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*. E87-A(6) (2004) 1355–1362
4. Nagy, G., Seth, S. C., and Stoddard, S. D.: Document analysis with an expert system. *Proceedings of Pattern Recognition in Practice II*. Amsterdam. (1985) 19–21
5. Schalkoff, R.: *Pattern Recognition: Statistical, Structural and Neural Approaches*. John Wiley & Sons. (1992)
6. Duda, R. O., Hart, P. E., and Stork, D. G.: *Pattern Classification*. 2nd Edition. John Wiley & Sons. (2001)
7. Kim, K. B., and Kim, C. K.: Performance Improvement of RBF Network using ART2 Algorithm and Fuzzy Logic System. *Lecture Notes in Artificial Intelligence*. LNAI 3339. (2004) 853–860
8. Park, J., and Sandberg, I. W.: Universal Approximation Using Radial Basis Function Network. *Neural Comp.* (1991) 246–257
9. Kim, K. B., and Yun, H. W.: A Study on Recognition of Bronchogenic Cancer Cell Image Using a New Physiological Fuzzy Neural Networks. *Japanese Journal of Medical Electronics and Biological Engineering*. 13(5) (1999) 39–43
10. Zimmermann, H. J.: *Fuzzy set Theory and its Applications*. Kluwer Academic Publishers. (1991)

An Novel Artificial Immune Systems Multi-objective Optimization Algorithm for 0/1 Knapsack Problems

Wenping Ma¹, Licheng Jiao¹, Maoguo Gong¹, and Fang Liu²

¹ Institute of Intelligent Information Processing, P.O. Box 224, Xidian University, Xi'an 710071, China

wpma@mail.xidian.edu.cn, maoguo_gong@hotmail.com

² School of Computer Science and Technology, Xidian University, Xi'an 710071, China

Abstract. Based on the concept of Immunodominance and Antibody Clonal Selection Theory, This paper proposes a new artificial immune system algorithm, Immune Dominance Clonal Multiobjective Algorithm (IDCMA), for multiobjective 0/1 knapsack problems. IDCMA divides the individual population into three sub-populations and adopts different evolution and selection strategies at them, but the update of each sub-population is not carried out all alone. The performance comparisons among IDCMA, SPEA, HPGA, NPGA, NSGA and VEGA show that IDCMA clearly outperforms the other five MOEAs in terms of solution quality.

1 Introduction

Since 1960s, the multiobjective optimization problems have attracted more attentions from researchers in various fields^[1]. With the appearance of evolutionary algorithms and its lucubrating, it attracts comparative attentions as a novel method solving multiobjective optimization problems: Schaffer put forward VEGA.^[2] Horn et al's NPGA and Srinivas et al's NSGA attracted more attentions^[3]. In recent years, a lot of newly improved algorithms were proposed, such as Zitzler's SPEA^[4]. Just like evolutionary algorithms, artificial immune system (AIS) constructs new intelligent algorithms with immunology terms and fundamenta.l^[5]

After describing the multiobjective 0/1 knapsack problem in Section 2, a novel multiobjective optimization algorithm for the multiobjective 0/1 knapsack problem, Immune Dominance Clonal Multiobjective Algorithm (IDCMA), is put forward in Section 3. Section 4 is the simulation analyses.

2 The Multiobjective 0/1 Knapsack Problems

The multiobjective knapsack problem considered here is defined in the following^[4]

Given a set of m items and a set of n knapsacks, with P_{ij} = profit of item j according to knapsack i , w_{ij} = weight of item j according to knapsack i , c_i = capacity of knapsack i . Its mathematic model:

$$\begin{aligned}
 \text{max.} \quad & y = f(x) = (f_1(x), f_2(x), \dots, f_n(x)) \\
 \text{subject to} \quad & \sum_{j=1}^m w_{i,j} x_j \leq c_i, x = (x_1, x_2, \dots, x_m) \in \{0,1\}^m
 \end{aligned} \tag{1}$$

find a vector $x = (x_1, x_2, \dots, x_m) \in \{0,1\}^m$, such that $\forall i \in \{1, 2, \dots, n\}; \sum_{j=1}^m w_{i,j} x_j \leq c_i$

and for which $f(x) = (f_1(x), f_2(x), \dots, f_n(x))$ is maximum, where $f_i(x) = \sum_{j=1}^m p_{i,j} \bullet x_j$ and $x_j = 1$ if item j is selected.

A 0/1 knapsack problem is a typical NP-hard problem. The dynamic programming can not search the satisfactory solutions with the feasible computation time and cost. As a result, we should find effective search method.

3 Solving Multiobjective 0/1 Knapsack Problems Based on IDCMA

Beginning with explanation of some terms used in IDCMA

Antigen: In artificial immune system, antigens refer to problems and its constraints.

Antibody: Antibodies represent candidates of the problem. For the multiobjective 0/1 knapsack problem, every candidate adopts binary coding with the length m , each binary bit represents the value of variable $x_i, i \in [1, m]$.

Antibody-Antigen Affinity: In AIS, it generally indicates values of objective functions or fitness measurement of the problem.

Antibody-Antibody Affinity: The reflection of the total combine power locates between two antibodies. In this paper, we compute the antibody-antibody affinity as reference [6]. Namely, if the coding of an antibody a_i is '1 1 0 0 0 1 0', and the coding of another antibody a_{di} is '1 1 0 1 0 1 1 0', then the antibody-antibody affinity between a_i and a_{di} is $6 + 3^2 + 2^2 = 19$.

Immune Dominance: For the multiobjective 0/1 knapsack problem, the antibody a_i is an immune dominance antibody in antibody population $A = \{a_1, a_2, \dots, a_{n_b}\}$, if there is no antibody $a_j (j = 1, 2, \dots, n_b \wedge j \neq i)$ in population A satisfied

$$(\forall k \in \{1, 2, \dots, n\} f_k(e^{-1}(a_j)) \geq f_k(e^{-1}(a_i))) \wedge (\exists l \in \{1, 2, \dots, n\} f_l(e^{-1}(a_j)) > f_l(e^{-1}(a_i))) \tag{2}$$

So the immune dominance antibodies are the Pareto-optimal individuals in current population.

Clonal Operation: In AIS, the clonal operation to the antibody population $A(k)$ is defined as:

$$Y(k) = T_c^C(A(k)) = [T_c^C(a_1(k)) \quad T_c^C(a_2(k)), \quad \dots, T_c^C(a_{n_b}(k))]^T \tag{3}$$

where $T_c^C(a_{ci}(k)) = I_{ci} \times a_{ci}(k) \quad i = 1, 2, \dots, N, I_{ci}$ is a q_{ci} -dimensional identity row vector.

Immune Differential Degree: The Immune Differential Degree denotes the relative distribution of an immune dominance antibody. Namely, assuming that there are n_d immune dominance antibodies in current population, f_{kl} is the value of the K -th objective function of the l -th antibody. The Immune Differential Degree of the l -th antibody a_l can be calculated as follow:

$$d_l^* = \min \left\{ d_l(m) = \sqrt{\sum_{k=1}^q \left(\frac{\phi(f_{kl}) - \phi(f_{km})}{\phi(f_{kl})} \right)^2} \mid l = 1, 2, \dots, n_d; m = 1, 2, \dots, n_d \wedge m \neq l \right\} \quad (4)$$

where $\phi(\bullet)$ is an incremental function without the value of zero.

Inspired from the immuodominance of the biology immune system and the clonal selection mechanism, IDCMA is based on clonal selection with immune dominance and clone anergy for the multiobjective 0/1 knapsack problems which can be implemented as follows:

Step 1: Give the termination generation G_{\max} , the size of Immune Dominance Antibody population n_d , the size of Generic Antibody population n_b , the size of Dominance Clonal Antibody population n_t , and clonal scale n_c . Set the mutation probability p_m , recombination probability p_c and coding length c . Randomly generate the original antibody population $A(0) = \{a_1(0), a_2(0), \dots, a_{n_b}(0)\} \in I^{n_b}$, $k:=0$;

Step 2: Modify $A(k)$ with the greedy repair method as reported in reference [4], obtain the antibody population $A(k)$ which satisfies the constrained conditions.

Step 3: According to the antibody-antigen affinities of all the antibodies in $A(k)$, constitute population $DT(k)$, if the number of antibodies in $DT(k)$ is no larger than n_d , let Immune Dominance Antibody population $D(k)=DT(k)$, go to Step6; otherwise go to Step4;

Step 4: Compute the Immune Differential Degrees in population $DT(k)$;

Step 5: Sort all the antibodies in $DT(k)$ by descending of their Immune Differential Degrees, and select the first n_d antibodies to constitute $D(k)$;

Step 6: If $k=G_{\max}$, export $D(k)$ as the output of the algorithm, Stop. Otherwise, replace the immune dominance antibodies in $A(k)$ by new antibodies generated randomly. Then marked the antibody population as $B(k)$;

Step 7: Select an immune dominance antibody a_{di} randomly from $D(k)$. Compute the antibody-antibody affinities between the antibodies in $B(k)$ and the antibody a_{di} .

Step 8: Sort all the antibodies in $B(k)$ by descending of their antibody-antibody affinities, select the first n_t antibodies to constitute the Dominance Clonal Antibody population $TC(k)$, and other antibodies to constitute the Immune Anergy Antibody population $NR(k)$.

Step 9: Implement the Antibody Clonal Operation T_c^C at $TC(k)$ and get the antibody population $CO(k)$ after clonal operation.

Step 10: Implement the recombination operation at $CO(k)$ with the probability p_c and get the antibody population $CO'(k)$.

Step 11: Implement the mutation operation at $CO'(k)$ with the probability p_m and get the antibody population $COT(k)$.

Step 12: Combine the populations $COT(k)$, $D(k)$ and $NR(k)$ to form the antibody population $A(k+1)$, $k=k+1$, go to Step 2.

4 Simulations

In order to validate the algorithm, we compare the algorithm with another five algorithms. They are Zitzler's Strength Pareto Evolutionary Algorithm (SPEA)^[4], Schaffer's Vector Evaluated Genetic Algorithm (VEGA)^[7], Hajela's and Lin's genetic algorithm (HLGA)^[8], the niched Pareto genetic algorithm (NPGA)^[9] and Srinivas' and Deb's Nondominated Sorting Genetic Algorithm (NSGA)^[10]. This paper selects the performance measure ζ in [11]. The test data sets are available from the authors[4], where two, three, and four objectives are taken under consideration, in combination with 250, 500, 750 items. The parameters setting of IDCMA are:

The halt generation $G_{\max}=500$, immune dominance antibody population size $n_d=100$, antibody population size $n_b=100$, dominance clonal antibody population size $n_l=50$, clonal scale $n_c=300$, coding length $c=m$ where m is the number of terms, mutation probability $p_m=2/c$, recombination probability $p_c=1$. Thirty independent runs of IDCMA are performed per test problem. The test problems and reported results of SPEA, HLGA, NPGA, NSGA and VEGA are directly gleaned from Zitzler' website: <http://www.tik.ee.ethz.ch/~zitzler/testdata.html/>.

The direct comparisons of IDCMA with the other algorithms based on the ζ measure from 30 runs are depicted using box plots, as shown in Figure 1. A box plot provides an excellent visual result of a distribution. The upper and lower ends of the box are the upper and lower quartiles, while a thick line with the box encodes the median. Dashed appendages summarize the spread and shape of the distribution.

As the Figure 1 shows, IDCMA achieves the best assessments among these multiobjective EAs. It covers 100% of the nondominated solutions found by HLGA with all the nine test problems, and covers 100% of the nondominated solutions found by NPGA with seven of the nine test problems, and covers 100% of the nondominated solutions found by VEGA with eight of the nine test problems, and covers 100% of the nondominated solutions found by NSGA with seven of the nine test problems, and covers 100% of the nondominated solutions found by SPEA with six of the nine test problems; For 2 knapsacks and 500 items 100% are covered, as the same as 2 knapsacks and 750 items. For 4 knapsacks and 250 items at least 81% are covered. And for other six test problems at least 91% are covered. Vice versa, those algorithms cover less than 2% of the IDCMA outcomes in all 270 runs. Therefore IDCMA can find solutions that are closer to the Pareto-optimal front than those produced by other five algorithms.

Figure 1 reveals the superiority of IDCMA over the compared MOEAs in terms of robustness and nondominated solution quality. The experimental results reveal that IDCMA outperforms SPEA, HLGA, NPGA, NSGA and VEGA, especially for problems with a large number of items or a large number of knapsacks. IDCMA also performs well for various numbers of knapsacks.

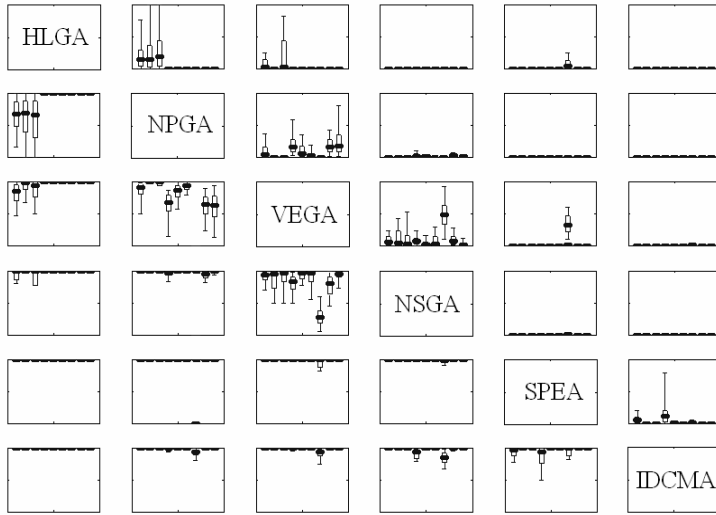


Fig. 1. Box plots based on the ζ measure. Each rectangle contains nine box plots representing the distribution of the ζ values for a certain ordered pair of algorithms; the three box to the left relate to 2 knapsacks and (from left to right) 250,500,750 items; correspondingly the three middle box plots relate to 3 knapsacks and the three to the right to 4 knapsacks. The scale is 0 at the bottom and 1 at the top per rectangle. Furthermore, each rectangle refers to algorithm A associated with the corresponding row and algorithm B associated with the corresponding column and gives the fraction of B covered by A ($\zeta(A, B)$).

5 Concluding Remarks

In this paper, a novel algorithm for the multiobjective 0/1 knapsack problem, Immune Dominance Clonal Multiobjective Algorithm is put forward, which is inspired by the concept of immunodominance and the clonal selection theory. From the numerical results of the metrics, Coverage of Two Sets, we draw a conclusion quantitatively that solutions obtained from IDCMA dominate those obtained from SPEA, HLGA, NPGA, NSGA and VEGA obviously. Especially for problems with a large number of items or a large number of knapsacks, the dominance of IDCMA is more obvious. Namely, for the discontinuous Pareto-optimal fronts or the isolated optimal point, IDCMA can construct and find them while the other algorithms seem incapable sometimes. At the same time, IDCMA is much better than the other five algorithms from the distributions of the solutions.

References

1. Hwang, C. and Yoon, K.: Multiple Attribute Decision Making: Methods and Applications, Springer-Verlage, Berlin, (1981)
2. Schaffer, J.D.: Multiple objective optimization with vector evaluated genetic algorithms. PhD thesis, Vanderbilt University (1984)

3. Abido, M.A.: Environmental economic power dispatch using multiobjective evolutionary algorithms. *IEEE Trans. Power Systems*, Vol.18, No. 4, November (2003)
4. Zitzler, E., Thiele, L.: Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Trans. Evolutionary Computation*. Vol. 3, No. 4, November (1999)
5. Jiao Licheng, Du Haifeng.: Artificial immune system: progress and prospect (in Chinese). *ACTA ELECTRONICA SINICA*, Vol.31, No.10, (2003) 1540-1548
6. Carlos A.Coello Coello and Nareli Cruz Cortss.: An Approach to Solve Multiobjective Optimization Problems Based on an Artificial Immune System. In: Jonathan Timmis and Peter J. Bentley (editors). *Proceedings of the First International Conference on Artificial Immune Systems (ICARIS'2002)*, 212-221, University of Kent at Canterbury, UK, September, 9-11 (2002)
7. J. D. Schaffer.: Multiple objective optimization with vector evaluated genetic algorithms, in *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, J. J. Grefenstette, Ed., Pittsburgh, PA, July 24-26 (1985) 93-100, sponsored by Texas Instruments and U.S. Navy Center for Applied Research in Artificial Intelligence (NCARAI)
8. P. Hajela and C.- Y. Lin.: Genetic search strategies in multicriterion optimal design, *Structural Optimization*, vol. 4, June (1992)99-107
9. J. Horn and N. Nafpliotis.: Multiobjective optimization using the niched pareto genetic algorithm, *IlligAL Report 93005*, Illinois Genetic Algorithms Laboratory, University of Illinois, Urbana, Champaign, July (1993)
10. N.Srinivas and K.Deb.:Multiobjective optimization using nondominated sorting in genetic algorithms, *Evolutionary Computation*, vol.2, no.3, (1994) 221-248
11. Eckart Zitzler. *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. A dissertation submitted to the Swiss Federal Institute of Technology Zurich for the degree of Doctor of Technical Sciences. Diss. Eth No. 13398 (1999)
12. Gong, M.G., Du, H.F., Jiao, L.C., Wang, L.: *Immune Clonal Selection Algorithm for Multiuser Detection in DS-CDMA Systems*. Springer-Verlag, LNCS 3339, (2004) 1219-1225
13. Jiao, L.C., Gong, M.G., Shang, R.H., DU, H.F., Lu, B.: *Clonal Selection with Immune Dominance and Energy Based Multiobjective Optimization*. Springer-Verlag, LNCS 3410, (2005) 474-489
14. Gong, M.G., Jiao, L.C., Liu, F. and Du, H.F.: *The Quaternion Model of Artificial Immune Response*. Springer-Verlag, LNCS 3627, (2005) 207-219

RD-Based Seeded Region Growing for Extraction of Breast Tumor in an Ultrasound Volume

Jong In Kwak¹, Sang Hyun Kim², and Nam Chul Kim¹

¹ Laboratory for Visual Communications, Department of Electronic Engineering,
Kyungpook National University, Daegu, 702-701 Korea
jikwak@vcl.knu.ac.kr, nckim@ee.knu.ac.kr

² Department of Multimedia Engineering,
Yongsan University, Yongsan, 626-847 Korea
ksh50@ysu.ac.kr

Abstract. This paper proposes a rate-distortion (RD) based seeded region growing (SRG) for extracting an object such as breast tumors in ultrasound volumes which contain speckle noise and indistinct edges. In the proposed algorithm, region growing proceeds in such a way that the growing cost is minimized which is represented as the combination of rate measuring the roughness of a region contour and distortion measuring the inhomogeneity of pixels in a region. An input image is first segmented into an initial seed region and atomic homogeneous regions. The seed is next merged with one of adjacent regions which makes the RD cost minimum at each step. Such a merging is repeated until the RD cost averaged over the entire seed contour reaches the maximum. As a result, the final seed holds region homogeneity and has a smooth contour while maximizing inhomogeneity against its adjacent regions. Experiments of extracting breast tumors in four real ultrasound volumes show the proposed method yields the average 40% improvement in error rate with respect to the results extracted manually over some conventional methods.

1 Introduction

Image segmentation [1] is a process of extracting an object of interest from an image, which has been used as a fundamental process for computer vision such as image analysis, target recognition, medical diagnosis, and so forth. Most of image segmentation methods can be largely classified into contour-based and region-based ones.

Among works of extracting diagnostic objects or organs in medical images, various contour-based methods are found as follows. In [2], an active contour algorithm is presented to extract a fetus from an ultrasound (US) image. In [3], an active contour algorithm using the external force called as gradient vector flow, which is computed as a diffusion of the gradient vectors of edge map, is used to extract the left ventricle of a human heart from a magnetic resonance image. A Laplacian of Gaussian (LoG) operator to extract a breast tumor from

a US volume is found in [4]. A series of adaptive neighborhood filtering, Sobel operation, watershed transform, and binarization are used in [5] to detect a follicle wall from a US image. In using a contour-based method to extract a diagnosis object from a US image which has speckle noise and indistinct edges, the estimated object contour often converges into local maxima or minima, which may result in a false object boundary.

Region-based segmentation methods applied for medical application are listed as follows. A SRG using pixel intensity and texture features is used in [6] to extract heart regions from a high-frequency intracardiac echocardiography image. In [7], an unseeded region growing using adaptive anisotropic diffusion filters is performed to extract the contour of a heart from a US image. In case that an object has weak edges caused by noise or fuzzy edges, the object region grown based only on region homogeneity may encroach through the frail edges on background, which in the end yields an incorrect object boundary. If such constraints on region contour as in contour-based methods are added to such considerations on region homogeneity as in region-based methods, the object boundary can be better extracted and more robust to noise.

In this paper, we propose a RD-based SRG considering contour smoothness and region homogeneity jointly, which is thus robust to speckle noise and proper for extracting a breast tumor which has an ambiguous boundary and relatively high intensity variation. In section II, we explain the proposed RD-based SRG. The extraction of a breast tumor from a US volume using the proposed method is given in section III. In section IV, we evaluate the performance of the proposed algorithm for an artificial and four real US data and present our conclusion.

2 Proposed RD-Based Seeded Region Growing

Preliminary: When an arbitrary region R_i adjacent to the seed R_0 is merged with the seed as in Fig. 1, the contour of the merged region, updated as a new seed, is expressed using a concatenation operation \oplus on the contours of the two prior regions as follows:

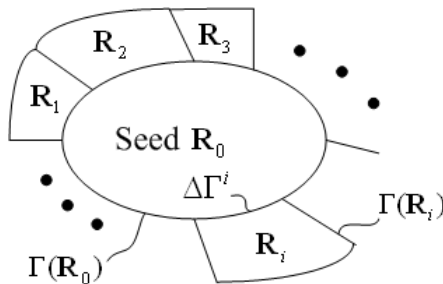


Fig. 1. Contour relation between the seed and its adjacent regions

$$\begin{aligned} \Gamma(\mathbf{R}_0 \cup \mathbf{R}_i) &\triangleq \Gamma(\mathbf{R}_0) \oplus \Gamma(\mathbf{R}_i) \\ &= (\Gamma(\mathbf{R}_0) \cup \Gamma(\mathbf{R}_i)) - (\Gamma(\mathbf{R}_0) \cap \Gamma(\mathbf{R}_i)) \end{aligned} \quad (1)$$

where Γ denotes the contour of each region. Letting $\Delta\Gamma^i$ be the common contour between the two prior regions, it is expressed as

$$\Delta\Gamma^i \triangleq \Gamma(\mathbf{R}_0) \cap \Gamma(\mathbf{R}_i) \quad (2)$$

where \mathbf{R}_i denotes one of adjacent regions of \mathbf{R}_0 and satisfies the condition of $\Delta\Gamma^i \neq \phi$.

Initialization: An input image \mathbf{I} is segmented into an initial seed $\mathbf{R}_0(0)$ and the remaining atomic regions $\mathbf{R}_i(0), 0 < i < N$. That is, we get

$$\mathbf{I} = \bigcup_{i=0}^{N-1} \mathbf{R}_i(0) \quad (3)$$

where N denotes the total number of segmented regions including the seed and all the regions are disjoint, that is, $\mathbf{R}_i(0) \cap \mathbf{R}_j(0) = \phi$ for all combinations of i and j . The initial seed is a homogeneous core region which is set manually and the atomic regions are small base regions which are partitioned automatically so as to hold region homogeneity.

Region growing: Let Γ_t be the contour of the seed $\mathbf{R}_0(t)$ at the step t .

$$\Gamma_t \triangleq \Gamma(\mathbf{R}_0(t)) \quad (4)$$

Then, merging with the adjacent region which minimizes a cost required to growing at this state, the seed is grown at the next step as follows:

$$\begin{aligned} \Gamma_{t+1} &= \Gamma_t \oplus \Gamma(\mathbf{R}_m(t)) \\ &= \Gamma_t \oplus \Gamma_t^m \end{aligned} \quad (5)$$

where $\mathbf{R}_m(t)$ denotes the adjacent region which minimizes the growing cost at the step t . The minimum-cost region is represented as

$$\mathbf{R}_m(t) = \min_{\{\mathbf{R}_i(t)\}} J(\Delta\Gamma_t^i) \quad (6)$$

where $\mathbf{R}_i(t)$'s are atomic regions, that is, $\mathbf{R}_i(t) \in \{\mathbf{R}_i(0), 0 < i < N\}$, adjacent to the seed $\mathbf{R}_0(t)$, and satisfy the condition $\Delta\Gamma_t^i \neq \phi$. The growing cost J means the cost which is required to merging of the two regions which share a common contour. To avoid the merging of the seed with any non-adjacent region, its growing cost is set $J(\phi) \triangleq \infty$.

Stop condition: The region growing is repeated until the growing cost $\bar{J}(\Gamma_t)$ averaged over the seed contour Γ_t reaches the maximum of

$$\max_t \bar{J}(\Gamma_t) = \max_t \frac{1}{|\Gamma_t|} J(\Gamma_t) \tag{7}$$

where $|\Gamma_t|$ denotes the length of the seed contour Γ_t . Since the seed contour Γ_t is obtained as the concatenation of common contours between the seed and its adjacent regions, that is, $\Gamma_t = \bigoplus_{i=1}^{K(t)} \Delta\Gamma_t^i$, let the total cost of the entire contour be given as the sum of costs of common contours. That is, we have

$$\begin{aligned} J(\Gamma_t) &= J\left(\bigoplus_{i=1}^{K(t)} \Delta\Gamma_t^i\right) \\ &\triangleq \sum_{i=1}^{K(t)} J(\Delta\Gamma_t^i) \end{aligned} \tag{8}$$

where $K(t)$ is the number of adjacent regions of the seed at the step t . Equation (7) and (8) mean that the region growing stops when the averaged growing cost between the seed and its adjacent regions becomes the maximum.

Selection of growing cost: As a growing cost, we choose a RD-based cost which jointly considers the rate indicating the roughness of a region contour and the distortion indicating the inhomogeneity of pixels in a region, which is given as

$$J(\Delta\Gamma_t^i) = D(\Delta\Gamma_t^i) + \lambda_t R(\Delta\Gamma_t^i) \cong -\frac{D(\Delta\Gamma_t^i)}{R(\Delta\Gamma_t^i)} \tag{9}$$

where $D(\Delta\Gamma_t^i)$ and $R(\Delta\Gamma_t^i)$ denote the distortion and the rate that occur in merging the seed $\mathbf{R}_0(t)$ with an adjacent region $\mathbf{R}_i(t)$. The parameter λ_t is the Lagrange multiplier which is introduced to combine the rate and distortion. For computational simplicity, the RD-based cost is approximated with the RD slope that is shown in [8] to yield the similar performance. Using such a RD-based cost, the seed is grown so as to merge with the adjacent region that simultaneously minimizes the roughness of the resulting seed contour and the inhomogeneity of the resulting seed region and, in the end, comes to have good region homogeneity and a smooth contour while maximizing inhomogeneity against its adjacent regions.

If we further select the mean of a region among various features for region homogeneity, then the distortion $D(\Delta\Gamma^i)$ in Equation (9) is given as [8]

$$D(\Delta\Gamma^i) = D(\mathbf{R}_0 \cup \mathbf{R}_i) - [D(\mathbf{R}_0) + D(\mathbf{R}_i)] = \frac{(\bar{\mathbf{R}}_0 - \bar{\mathbf{R}}_i)^2}{|\mathbf{R}_0|^{-1} + |\mathbf{R}_i|^{-1}} \tag{10}$$

where $|\mathbf{R}|$ and $\bar{\mathbf{R}}$ denote the size and mean of a region, respectively, and $D(\mathbf{R})$ the sum of squared approximation error of \mathbf{R} . As a contour smoothness, we estimate the rate $R(\Delta\Gamma^i)$ in Equation (9) simply with the length of the common contour, which is given as [8]

$$R(\Delta\Gamma^i) \cong -\{\alpha|\Delta\Gamma^i| + \beta\} \tag{11}$$

where α denotes the estimated slope, $|\Delta\Gamma^i|$ the length of the common contour, and β the bias parameter. The growing cost chosen here is completely specified by Equation (9) \sim (11).

3 Extraction of Breast Tumor in an Ultrasound Volume

Fig. 2 shows the block diagram of extraction of a breast tumor in a US volume using the proposed SRG method. In the cutting planes generation, a set of cutting planes is obtained by equiangular revolution of a cross sectional plane on a reference axis for 3D volume data [9]. In the preprocessing, median filtering is performed in each cutting plane to reduce speckle noise and scattering [10], [11]. In a material such as an organ, tissue, and so on, the farther the distance from an ultrasonic transducer to a target point is, the more severe its attenuation is [12]. Such attenuation is linearly compensated after median filtering. An elliptic seed is subsequently set in a breast tumor by locating the two points on the tumor boundary by a user, which is based on the fact that the shape of a breast tumor is close to an ellipse.

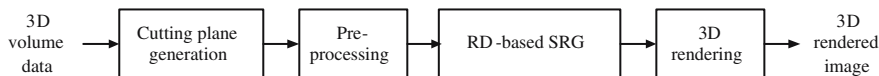


Fig. 2. Block diagram of extraction of a breast tumor using the proposed SRG method

As mentioned in Chapter 2, in the RD-based SRG, region growing proceeds in such a way that the RD growing cost is minimized and the contour of the final seed becomes close to the contour of a breast tumor. Fig. 3 shows the results of the RD-based SRG for a 2D cutting plane of a breast tumor. Fig. 3(a) shows a 2D cutting plane of a breast tumor and Fig. 3(b) the contours of the initial seed and its atomic regions

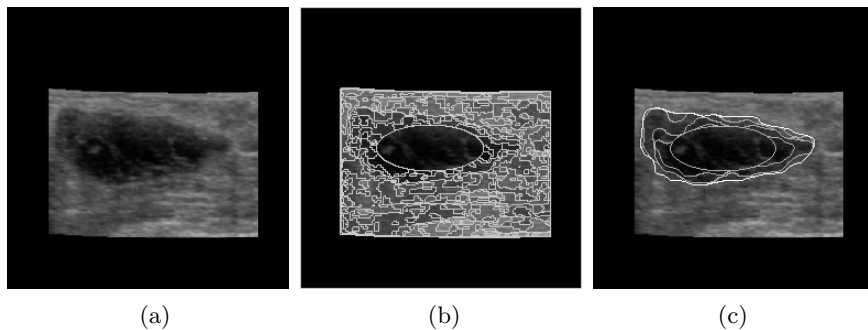


Fig. 3. Results of the RD-based SRG for a 2D cutting plane of a breast tumor: (a) 2D cutting plane; (b) contours of the initial seed and its atomic regions; (c) contours of the initial, intermediate, and final seeds

and its atomic regions. Fig. 3(c) shows the contours of the initial, intermediate, and final seeds. It is shown that the extracted tumor contour is so smooth and consistent well with the actual boundary. In the 3D rendering, a fixed number of vertices are first extracted from the tumor contour in each cutting plane and a 3D wire frame [13] is formed using polygons composed of these vertices. Finally a tumor volume is rendered from the 3D wire frame by using a surface-based method [14].

4 Experimental Results and Discussion

An artificial data and four real US volume data of a breast tumor are used for performance evaluation of the proposed tumor extraction method. Cutting planes of all the test US volumes are composed of 200×200 pixels. The artificial data is generated as a 3D Markov random field, based on the convolution model [15]. As a performance measure, the error rate of a tumor contour extracted automatically against the corresponding one extracted manually is used which is defined as

$$\text{Error rate}[\%] = \frac{N_{FA} + N_{FD}}{N_M} \times 100 \quad (12)$$

where N_M , N_{FA} , and N_{FD} denote the size of a tumor measured manually, the number of false alarm pixels, and the number of false dismissal pixels for each cutting plane, respectively.

Fig. 4 shows the average error rates over all cutting planes according to various segmentation methods for the artificial data. In the figure, the distortion-based and mean-difference based SRGs mean the segmentation methods using the distortion and mean-difference instead of the RD cost in the proposed method, respectively. Fig. 4(a) shows a cutting plane of the artificial data. We can see in Fig. 4(b) that the average error rate of the proposed RD-based SRG is lower by a half than that of the mean-difference based SRG. Shown in Fig. 5 are the

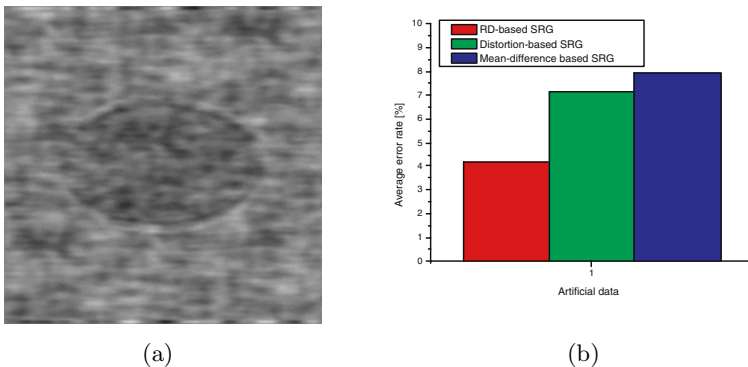


Fig. 4. Average error rates according to various segmentation methods for the artificial data: (a) 2D cutting plane; (b) average error rates

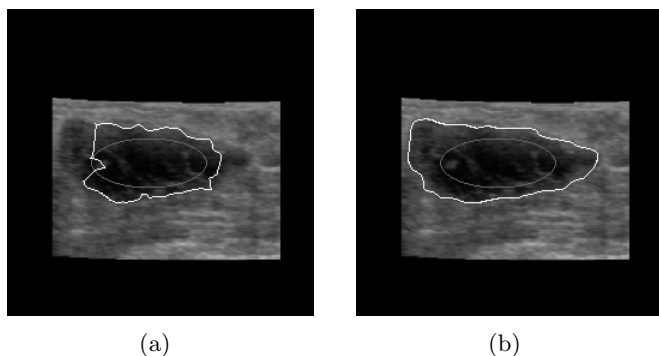


Fig. 5. Results of the proposed method and the active contour algorithm using GVF for a cutting plane of a real US volume: (a) contour extracted by the GVF-based active contour algorithm; (b) contour extracted by the proposed method

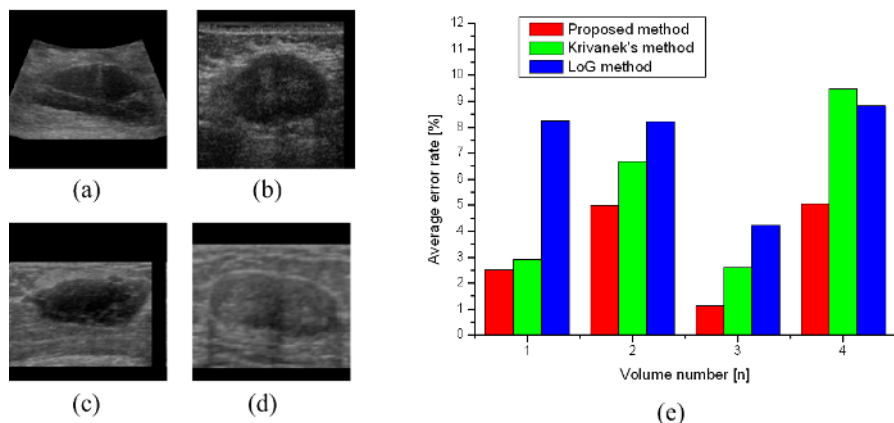


Fig. 6. 2D cutting planes and average error rates: (a),(b),(c), and (d) are 2D cutting planes; (e) average error rates

results of the proposed method and active contour algorithm using GVF [3] for a cutting plane of a real US volume. Fig. 5(a) shows the initial contour identical to the contour of the initial seed in (b) and the final contour that is extracted by the GVF-based active contour algorithm. In Fig. 5(a), the estimated tumor contour converges into local maxima, which results in the false tumor boundary. Fig. 5(b) shows the contour of the initial seed and the final contour that is extracted by the proposed method. In Fig. 5(b), the estimated tumor contour overcomes local maxima, which results in the contour near to the real tumor boundary.

Fig. 6 shows 2D cutting planes and average error rates for four real US volumes. Fig. 6(a) ~ (d) show the 2D cutting planes and Fig. 6(e) shows the average error rates of the proposed method and Krivanek's method [5]. We can see in

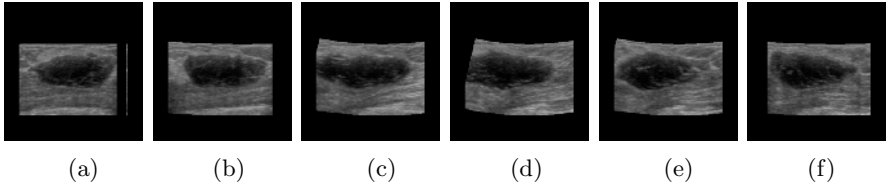


Fig. 7. 2D cutting planes obtained by equiangular revolution of a cross sectional plane for a 3D breast tumor volume: (a) Frame #0; (b) frame #1; (c) frame #2; (d) frame #3; (e) frame #4; (f) frame #5

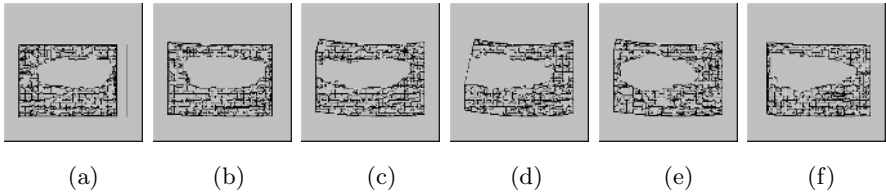


Fig. 8. Results of the RD-based SRG for a real breast tumor: (a) Frame #0; (b) frame #1; (c) frame #2; (d) frame #3; (e) frame #4; (f) frame #5

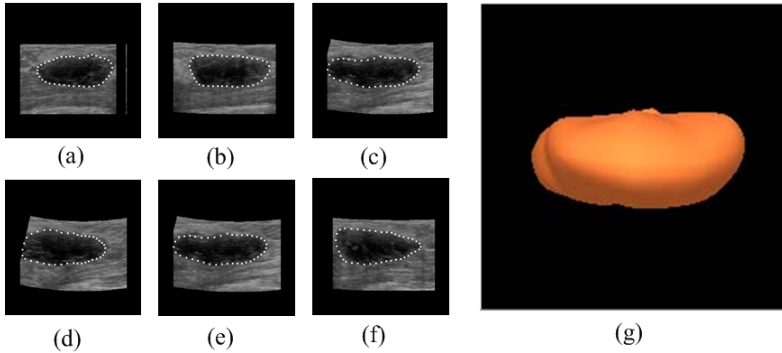


Fig. 9. Vertices extracted from tumor contours and 3D rendered tumor volume: (a) Frame #0; (b) frame #1; (c) frame #2; (d) frame #3; (e) frame #4; (f) frame #5; (g) 3D rendered tumor volume

Fig. 6(e) that the proposed method yields better average error rates of 1.1 ~ 5% over Krivanek’s method which yields 2.7 ~ 9.5%. Fig. 7 ~ Fig. 9 show the intermediate results of extraction of a 3D tumor using the proposed method. Fig. 7 shows 2D cutting planes obtained by equiangular revolution of a cross sectional plane for a 3D breast tumor volume.

Fig. 8 shows the results of the RD-based SRG for a real breast tumor. Fig. 9(a) ~ (f) show the vertices extracted from tumor contours in cutting planes. Fig. 9(g) shows a 3D tumor volume rendered by linearly interpolating the vertices obtained in Fig. 9(a) ~ (f). We can see that the tumor contours extracted by the

proposed SRG coincide nearly with those extracted by human vision. Through the experimental results, we can see that the proposed method is an efficient method for extracting a breast tumor from US volumes.

5 Conclusion

We have proposed a RD-based SRG whose growing cost measures contour smoothness and region homogeneity jointly so as to be robust to speckle noise and proper for extracting a breast tumor containing an ambiguous boundary and relatively high intensity variation. In the experimental results, the breast tumor contours extracted by the proposed method are shown to coincide nearly with those extracted by human vision. The 3D rendering of breast tumor volumes confirms the efficiency of the proposed method.

Acknowledgement

This research was done by the University Research Program supported by Medison Co., Ltd. Korea.

References

1. R. C. Gonzalez and R. E. Woods, Digital Image Processing, Addison-Wesley (1993)
2. M. Martin, E. Rodriguez: Energy functions for the segmentation of ultrasound volume data using active rays, Proc. of the IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP'00) 4(2000)2274-2277
3. C. Xu, J. L. Prince: Snakes, shapes, and gradient vector flow, IEEE Trans. on Image Processing 7(1998)359-369
4. S. Ogawa and K. Itoh: Three dimensional ultrasonic imaging for diagnosis of breast tumor, IEEE Ultrasonics Symp. (1998)1677-1680
5. A. Krivanek, M. Sonka: Ovarian ultrasound image analysis: Follicle segmentation, IEEE Trans. Medical Imaging 17(1998) 935-944
6. X. Hao, C. Bruce, C. Pislaru, J. Greenleaf: Segmenting high-frequency intracardiac ultrasound images of myocardium into infarcted, ischemic, and normal regions, IEEE Trans. Medical Imaging 20(2001)1373-1383
7. Z. Lin, J. Jin, H. Talbot: Unseeded region growing for 3D image segmentation, ACM International Conference Proceeding Series, Selected papers from the Pan-Sydney workshop on Visualization 2(2000)31-37
8. C. W. Lim, N. C. Kim, S. C. Jun, C. S. Jung: Rate-distortion based image segmentation using recursive merging, IEEE Trans. Circuits Syst. Video Technology 10(2002)1121-1134
9. J. I. Kwak, M. N. Jung, S. H. Kim, N. C. Kim: 3D segmentation of breast tumor in ultrasound images, Proc. in Biomedical Optics and Imaging: Medical Imaging 2003 Ultrasonic Imaging and Signal Processing 4(2003)193 - 200
10. R. Entrekin, P. Jackson, J. R. Jago, B. A. Porter: Real time spatial compound imaging in breast ultrasound: Technology and early clinical experience, Medical Mundi 43(1999)35-43

11. P. N. T. Wells: Absorption and dispersion of ultrasound in biological tissue, *Ultrasound in Medicine and Biology* (1975)369-376
12. G. Xiao, M. Brady, A. Noble, Y. Zhang: Segmentation of ultrasound B-Mode images with intensity inhomogeneity correction, *IEEE Trans. on Medical Imaging* 21(2002)48-57
13. C. S. Choi, H. Harashima, T. Takebe: Analysis and synthesis of facial expressions in knowledge-based coding of facial image sequences, *Proc. of ICASSP* (1991)2737-2740
14. A. Watt: *Fundamental of Three-Dimensional Computer Graphics*, Addison Wesley (1990)
15. T. Lango, T. Lie, O. Husby, J. Hokland: Bayesian 2-D deconvolution: Effect of using spatially invariant ultrasound point spread function, *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control* 48(2001)

Improving Classification for Microarray Data Sets by Constructing Synthetic Data

Shun Bian and Wenjia Wang

School of Computing Sciences, University of East Anglia, Norwich, NR4 7TJ, UK
{bs, wjw}@cmp.uea.ac.uk

Abstract. Microarray technology has been widely used in biological and medical research to observe a large number of gene expressions. However, such experiments are usually carried out with few replica or instances, which may lead to poor modelling and analysis. This paper suggests an approach to improve classification by using synthetic data. A new algorithm is proposed to estimate synthetic data value and the generated data are labelled by ensemble methods. Experiments with artificial data and real world data demonstrate that the proposed algorithm is able to generate synthetic data on uncertain regions of classifiers to improve effectiveness and efficiency of classification on microarray data sets.

1 Introduction

Microarray Technology uses DNA chips to carry out a single experiment on massive genes and generates large quantity of gene expression data. It has been applied to many fields including identifying genetic mutations and variation, finding DNA-based drug candidates, and rapidly discovering or measuring genetic predisposition to disease *etc.*

Microarray data are characterised with a large number of features (genes) but a small number of instances due to the cost in microarray experiments. There currently exist an arrange of techniques, such as dimensionality reduction, to reduce the number of features. Nevertheless, there are few virtually effective techniques to deal with the shortage of instances for improving machine learning. In this paper, a synthetic data construction is proposed as a potential solution to the latter problem. Based on a concept “uncertain region” [1], an algorithm (LARGE) is devised to estimate values of synthetic data. Ensemble is applied for labelling the generated new data and evaluating performance. Experiments are performed on both artificial and real world microarray data sets to test the new algorithm and ensemble method.

2 Background

2.1 Constructing Synthetic Data in Machine Learning

Synthetic data construction is a technique to generate artificial data when the real data are insufficient for modelling. The use of well-designed synthetic data

offers several benefits, such as meeting the demand of the size of training sets, determining actions for adjusting parameters of a data generator and understanding the distribution of examples [2][3]. Although a number of studies [3][4][5][6] have been applied in machine learning and data mining domains in recent decades, a common problem is that new data that are generated from the whole problem domain maybe unuseful and also result in computationally intensive problem.

2.2 Uncertain Region Theory and Selective Sampling

For improving a learner's generalisation ability, Cohn [1] in 1992 proposed a concept of "region of uncertainty", which is the area in a problem domain where the learner's misclassification is more likely to occur, and can be described as:

$$R(S^m) = \{x : \exists_{c_1, c_2} \in C, c_1(x) \neq c_2(x)\}$$

where, C is a concept class and S^m is a set of m examples. Classifiers c_1, c_2 are consistent with all $x \in S^m$. $R(S^m)$ serves as an envelope for consistent concepts. A data point that falls outside will leave it unchanged; a point inside will further restrict the region. Hence any disagreement between those concepts must lie in $R(S^m)$.

Cohn believes that as we draw more and more samples from a region of uncertainty of a learner, the region can be reduced without decreasing efficiency. This process is called selective sampling, and usually consists of two steps: *drawing*—taking an unclassified example from the current data distribution and *querying*—asking for the classification of that data point [1].

Several algorithms have been suggested for sample selection [7][8][9]. Seung [7] proposed *query by committee* (QBC), in which a committee of students is trained on the same data set. Query samples are selected according to the *principle of maximal disagreement*. Abe and Mamitsuka [9] extended Seung's proposal by using Boosting and Bagging as query learning strategies.

Summarily, the committee filter tends to select examples that split the set of hypotheses into two subsets of comparable size, because if one subset contains most of the hypotheses, then the probability that the two subsets disagree will be very small. This can be shown below:

$$x^* = \operatorname{argmin}_{x \in C} |\{t \leq T | h_t(x) = 1\}| - |\{t \leq T | h_t(x) = 0\}|$$

where C is a set of samples whose distribution is in a given domain; x^* is a sample picked from C ; T is number of times the algorithm is called; h is the hypothesis.

In data labelling (querying), QBC can also be used for labelling unclassified samples by using a hypothesis randomly picked from the version space. Melville *et. al.* [4] proposed an algorithm, DECORATE, for labelling. A diverse ensemble is built by training a new classifier on the union of the original training data and artificial data. If adding this new classifier to the current ensemble increases the ensemble training error, then this classifier is rejected, otherwise it is added to the current ensemble.

3 Methodologies – LARGE and ECS

We believe that generating synthetic data on the ambiguous area of a classifier is a key issue to improve effectiveness and efficiency of synthetic data. A new synthetic data construction algorithm shown in the Figure 1, called Learner’s uncertain ReGion Estimator, abbreviated as “LARGE”, is proposed based on the concept of “region of uncertainty”. A new sample is generated by gradually increasing or (decreasing) the feature’s values of an original training sample as well as predicting its category by a trained classifier. If its prediction keeps changing on the predicted category, that means the new data is approaching or already in the uncertain region of the classifier, then it can be selected as a new synthetic data for the classifier.

Labelling is made by Ensemble Classification System (ECS), a data mining software program developed by the authors based on diversity and ensemble theories. Rather than using single kind of classifiers like in DECORATE, ECS employs different kind of algorithms as base classifiers to create a diverse ensemble, including decision trees, neural network, Bayesian network, SVM and rules, *etc.* Besides, it uses feature selection to further increase the diversity between classifiers. ECS consists of Data Process, Feature Selection, Classifier Selection, Feature-classifier Pair Selection and Combination Modules [10].

Table 1. Summary of Performance of ECS. Columns indicate the names of data sets, number of total generated synthetic data, number of the data whose probability estimations are above threshold, size of comparison data set, changing of performance when adding given comparison samples, initial correct rates of combination, final results using all synthetic data and using the certain sets only and changes of performance respectively. (↑) means increase, (↓) decrease, and (→) no-change.

Data sets	New data	Higher conf.	Random data	Comparison (↑)/(↓)	Start	Final by whole	Final by certainty	System (↑)/(↓)
15-T	12	11	15	(↓)	96.67	96.67	94.81	(↓)
20-T	17	8	20	(↓)	95.0	95.0	96.0	(↑)
25-T	22	11	25	(↓)	98	98.4	98.0	(→)
30-T	27	23	30	(↓)	97.67	98.67	99.0	(↑)
40-T	37	24	40	(↓)	98.5	98.5	98.75	(↑)
50-T	47	23	50	(→)	98.2	98.6	98.6	(↑)

4 Experiment

4.1 Experiment with Artificial Data

A simple artificial dichotomy classification problem with one feature is defined to evaluate the effectiveness of LARGE [11]. It is clear that the ensemble method produced better or equivalent results when using synthetic data. Figure 2 depicts

```

Main
  Read  V, C, N, K, F, X, B, η
  Set   a ← X, v ← ηV, F(a) ← C, k ← 0, r ← 0
  Repeat
    While a ≤ B And F(a) = C
      a ← a + v, k ← k + 1
      If F(a) = -C Then Do FIND_VALUE
    End While
    If a > B Then Set a ← X, v ← -v, r ← r + 1
    Else Output a
    End If
  Until r = 2
  Output a
-----
Find_Value
  Set n ← 0
  Repeat
    Set v ← ηV, k ← 0
    While F(a) ≠ -C And k ≠ K
      a ← a + v, k ← k + 1
    End While
    Set C ← -C, v ← -ηV, k ← 0, n ← n + 1
  Until n = N
  Return a

```

Fig. 1. Pseudo-code of LARGE. In the code, V is the standard variance of an feature; C is the class type usually 0 or 1 for a two-class problem; N is the times of changing direction; K is the times of changing value; F is a decision function of a classifier; X is a sample chosen from the training data set; B is the boundary of a feature; and η is a changing rate. Parameters can be adjusted by user to obtain “optimal” result.

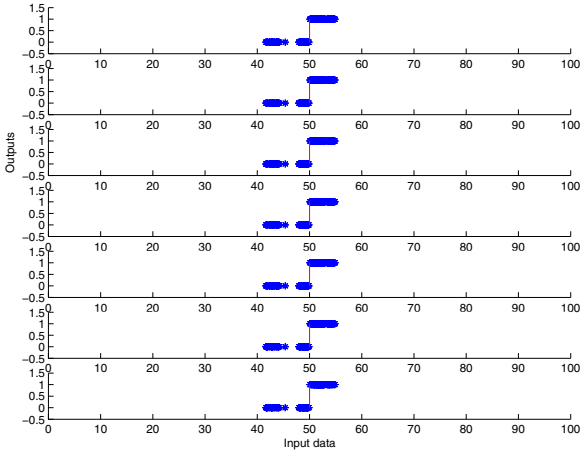


Fig. 2. Plot of new synthetic data generated by 7 base classifiers based on an artificial data set. The vertical line on each x-axis is the true decision line of the decision function.

Table 2. Partial properties and evaluation results on microarray data sets. Names and original feature numbers of each set are listed in the first column. The second and third columns list the total number of base classifiers with increased or decreased accuracy, while the fourth and fifth columns present the system performance when comparison samples or synthetic samples are included in the training set. (\uparrow) means increase, (\downarrow) decrease, and (\rightarrow) no-change.

Data sets	No. of Train	No. of Test	No. of down	No. of improved	Comparison Results	Ensemble Results
Leukemia (7130)	34	38	1 (C_1)	3 (C_2, C_3, C_4)	(\downarrow) (91.2 to 88.5)	(\rightarrow) (91.2)
LungCancer (12534)	32	149	0	5 (C_1, C_2, C_3, C_5, C_7)	(\downarrow) (81.4 to 75.8)	(\uparrow) (81.4 to 87.3)
Lymphoma (2985)	22	27	1 (C_1)	3 (C_3, C_5, C_7)	(\downarrow) (75.6 to 72.8)	(\uparrow) (75.6 to 76)

the location of synthetic data points. It shows that newly generated synthetic data appear in the regions of uncertainty around the true decision line of the function (presented by a vertical line). Furthermore, most of the data are labelled correctly by the ensemble.

4.2 Experiment with Microarray Data

Data Sets. To further test the performance of LARGE, we used 3 microarray data sets in this study, *i.e.* AMLALL_Leukemia and Lung Cancer that are taken from Kent Ridge Bio-medical Data Set Repository and Lymphoma taken from the UCI Repository of Machine Learning Database. Due to their high dimensionality, a feature selection procedure, the Correlation-based Feature Selection (CFS) and Best First Search Strategy, was applied to reduce the data to a possible minimum sign without losing too much useful information [10]. The number of the training and test samples of each data set are listed in Table 2. For checking consistency, 10 comparison data sets were generated by randomly selecting values within the same range of training data. Each comparative sample was labelled randomly.

Results and Discussion. The evaluation results are listed in Table 2. As shown in the table, the performance of classification system is improved for Lung Cancer and Lymphoma data, and same accuracy is produced for Leukemia data when synthetic data are included. In contrast to the comparison data, which cause a clear decrease in the classification accuracy, the use of carefully generated synthetic data in the training process demonstrated to be effective for improving classifier’s learning performance. Furthermore, all but one of (C_1) the base classifiers yield better or equal accuracy on test data.

5 Conclusion and Further Work

The experiments presented in this paper have demonstrated that LARGE, though applied only to one dimensional data, achieves promising performance in producing useful synthetic data that locate in or near the uncertain region of base classifiers. Ensemble methods from ECS provide a relatively effective and practical way for labelling new synthetic data. To construct synthetic data by the presented approach is shown potential in improving the performance of classification systems on microarray data sets, which usually suffer from insufficient samples and huge number of features.

LARGE can also be extended for constructing multi dimensional data if some factors, such as co-relation among features, are considered. This will be apart of future work.

References

1. Cohn, D.: Improving generalization with active learning. *Machine Learning* **15** (1992) 201–221
2. Aha, D.: Tolerating noisy, irrelevant and novel attributes in instance-based learning algorithms. *International Journal of Man Machine Studies* **36** (1992) 267–287
3. Hunniford, T.J., Hickey, R.J.: A simulated annealing technique for generating artificial data to assess concept learning algorithms. *Intelligent Data Analysis* **3** (1999) 177–189
4. Melville, P., Mooney, R.: Constructing diverse classifier ensembles using artificial training examples. *Proc. Inter. Joint Conference on Artificial Intelligence* (2003)
5. Varga, T., Bunke, H.: Comparing natural and synthetic training data for off-line cursive handwriting recognition. *Proc. of the 9th International Workshop on Frontiers in Handwriting Recognition* (2004)
6. Potharst, R., van Wezel, M.: Generting artificial data with monotonicity constraints. *Econometric Institute Report EI2005-06* (2005)
7. Seung, H., Oppor, M., Sompolinsky, H.: Query by committee. *Proc. Fifth Annual Workshop on Computational Learning Theory* (1992)
8. Freund, Y., S., S.H., Shamir, E., Tishby, N.: Information, prediction, and query by committee. *Advances in Neural Information Processing Systems* **5** (1993) 483–490
9. Abe, N., Mamitsuka, H.: Query learning strategies using boosting and bagging. *Proc. Fifteenth International Conference on Machine Learning* (1998)
10. Bian, S., Wang, W.: Developing diverse ensemble for classification of gene microarray data. *Proc. IEEE Workshop on Life Science Data Mining, ICDM'04* (2004)
11. Bian, S., Wang, W.: A preliminary investigation on generating synthetic data by active learning for improving ensemble performance. *Proc. of the Twenty-fifth Annual International Conference of the British Computer Society's Specialist Group on Artificial Intelligence (SGAI)* (2005)

A Method to Locate the Position of Mobile Robot Using Extended Kalman Filter

Ping Wei¹, Chengxian Xu¹, and Fengji Zhao²

¹ School of Science, Xi'an Jiaotong University, Xi'an, 710049, Shaanxi, P.R. China
weiping@mail.xjtu.edu.cn, mxxu@mail.xjtu.edu.cn

² Tohoku University, Sendai, Japan
zhao@fxc.jp

Abstract. A method to estimate long distance navigation of a mobile robot is proposed. The method uses the dead reckoning, sonar and infrared sensors to detect the landmarks. A corridor environment with equal spaced convex edges is applied as the mobile robot's moving space and the convex edges are used as landmarks for the robot mounted with the combined sensor system to estimate its position. The robot detects the convex edges using combined sensor system, and navigates in this corridor by using the information obtained from dead reckoning and combined sensor system based on the Extended Kalman. Experiment result show the effectiveness of the method.

1 Introduction

Many works in the field of the mobile robot navigation have been conducted but only in laboratory environments [1,2,3]. One important aspect with these works is how to scale up them into more complex environments covering much longer distance [4]. Some methods for navigating in long distance environments have been proposed, and one of them is called "wall following" [5]. The method assumes an environment in which there is no wall to follow but pillars with convex shapes are available for the robot to navigate.

In the paper, the mobile robot mounted with a combined sensor system (sonar and infrared) is tested in a real corridor environment. The size of the corridor environment has several tens meters, and it is called "long distance navigation". In the corridor environment, convex edges are a special type of landmark that can reliably be detected in successive sensor measuring. Moreover, the convex edges are stable, have features of natural environment, and hence they are useful landmarks for robot navigation.

The paper is organized in three parts. In the first part, we introduce the method that uses the combined sensor system mounted on the mobile robot to detect the convex edges in the corridor environment. The second part of the paper presents the method for the robot navigating in the corridor environment by using the dead reckoning and detecting the convex edges based on the Extended Kalman Filter (EKF); The last part is used to give experimental results of the navigation using the mobile robot.

2 Estimating the Trajectory

A mobile robot is assumed to move on the horizontal (X, Y) -plane at all the time. The robot's position and orientation are specified by the coordinates (x, y, θ) , where (x, y) are the coordinates of the robot body's center, and θ is the robot's orientation relative to the X -axis. The robot-centered coordinate $X'O'Y'$ is defined based on the center of the robot on the $X - Y$ plane, with the X' -axis pointing to the exact front of the robot, Y' -axis to the right, r is the distance from the robot to the POE of the convex edge and α is the angle between the direction of r and the X' -axis.

At the starting point, the robot's location in the X - Y plane is defined to be $x = 0, y = 0, \theta = 0$, and the $X'-Y'$ plane coincides with the X - Y plane.

2.1 The System Model

The position and orientation of the mobile robot at time k is denoted by the state vector $\mathbf{x}(k) = [x(k), y(k), \theta(k)]^T$. After a time period ΔT , the position and orientation of the robot is updated as

$$\begin{pmatrix} x(k+1) \\ y(k+1) \\ \theta(k+1) \end{pmatrix} = \begin{pmatrix} x(k) \\ y(k) \\ \theta(k) \end{pmatrix} + \begin{pmatrix} \Delta T v(k) \cos(\theta(k) + \alpha(k)) \\ \Delta T v(k) \sin(\theta(k) + \alpha(k)) \\ \alpha(k) \end{pmatrix}. \quad (1)$$

The state $\mathbf{x}(k)$ of the robot changes with time in response to an input $\mathbf{u}(k)$, and a disturbance noise $\mathbf{w}(k)$, and the translation equation of the state can be expressed as

$$\mathbf{x}(k+1) = \mathbf{f}[\mathbf{x}(k), \mathbf{u}(k)] + \mathbf{w}(k), \quad (2)$$

where $\mathbf{x}(k)$ is the state of the robot at time k , $\mathbf{u}(k) = [v(k), \alpha(k)]^T$ represents the external input to control the system at time k , and $v(k), \alpha(k)$ are the steering angle and the velocity of the driven wheel. $\mathbf{w}(k)$ is a zero-mean process noise and is defined by the covariance matrix $\mathbf{Q}(k)$. \mathbf{f} is the transition matrix, which relates the state at time k to the state at time $k+1$.

2.2 Estimation Errors Using the Dead Reckoning

There exist errors when equation (2) is used to estimate the state of the robot from a time step to then next time step, due to the facts that the dead reckoning measurements from the wheels are inexact and wheels slippage. Let $\hat{x}(k)$ be the estimated value of $\mathbf{x}(k)$, $\Delta\hat{\mathbf{x}}(k)$ be the error in $\hat{\mathbf{x}}(k)$, $\hat{\mathbf{u}}(k)$ the measured value of $\mathbf{u}(k)$, and $\Delta\hat{\mathbf{u}}(k)$ the error in $\hat{\mathbf{u}}(k)$. Then

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{f}[\hat{\mathbf{x}}(k) + \Delta\hat{\mathbf{x}}(k), \hat{\mathbf{u}}(k) + \Delta\hat{\mathbf{u}}(k)] + \mathbf{w}(k) \\ &\cong \mathbf{f}[\hat{\mathbf{x}}(k), \hat{\mathbf{u}}(k) + \mathbf{J}_1(k)\Delta\hat{\mathbf{x}}(k) + \mathbf{J}_2\Delta\hat{\mathbf{u}}(k)] + \mathbf{w}(k) \\ &= \hat{\mathbf{x}}(k+1) + \Delta\hat{\mathbf{x}}(k+1) \end{aligned} \quad (3)$$

and the errors appeared in dead reckoning are

$$\Delta\hat{\mathbf{x}}(k+1) = \mathbf{J}_1(k)\Delta\hat{\mathbf{x}}(k) + \mathbf{J}_2(k)\Delta\hat{\mathbf{u}}(k) + \mathbf{w}(k), \quad (4)$$

where

$$\begin{aligned} \mathbf{J}_1(k) &= \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k), \hat{\mathbf{u}}(k)), \\ \mathbf{J}_2(k) &= \frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\hat{\mathbf{x}}(k), \hat{\mathbf{u}}(k)), \end{aligned}$$

The estimation accuracy then can be derived from the error ellipsoid [8], that is estimated using the dead reckoning and the error ellipsoid on XY -plane. It can be observed from the trajectory that the uncertainty in the estimation using the dead reckoning system increases as the robot moves. Thus, it is not reliable for a robot to navigate a long distance if it only relies on a dead reckoning system.

3 The Calculation Using EKF of Trajectory

In this subsection, we present the method with which the robot’s position estimated from the dead reckoning can be corrected by the measurements made by the landmark detection sensor. The following nonlinear model relates the state vector $x(k)$ and the landmark p_i .

$$\mathbf{z}(k) = \mathbf{h}[\mathbf{x}(k), \mathbf{p}_i] + \mathbf{v}(k), \tag{5}$$

where $\mathbf{z}(k)$ is the observation vector, \mathbf{p}_i is the location of the convex edge, $\mathbf{p}_i = [x_i, y_i, \alpha, r]^T$. \mathbf{h} is the matrix that transforms the vector in state space into the observation space,

$$\mathbf{h}[\mathbf{x}(k), \mathbf{p}_i] = [(x(k) - x_i) + r \cos(\theta + \alpha)(y(k) - y_i) + r \sin(\theta + \alpha)]$$

and $\mathbf{v}(k)$ is the noise in measurement defined by the covariance matrix $\mathbf{R}(k)$. It is assumed that the process noise $\mathbf{w}(k)$ and the measurement noise $\mathbf{v}(k)$ are Gaussian, and uncorrelated white sequence.

The exact states of the robot are unknown, and can be estimated. Since the process and observation models are disturbed by random processes, the estimation errors are also random variables. However, we can approximate the state and observation vectors by omitting $w(k)$ and $v(k)$

$$\tilde{\mathbf{x}}(k + 1) = \mathbf{f}[\tilde{\mathbf{x}}(k), \mathbf{u}(k)], \tag{7}$$

$$\tilde{\mathbf{z}}(k) = \mathbf{h}[\tilde{\mathbf{x}}(k), \mathbf{p}_i], \tag{8}$$

where $\hat{\mathbf{x}}(k)$ is an approximation of the state $x(k)$ at time k .

When the non-linearity of the system is not “too serious”, equations (7)and(8) can linearly approximated using the linear Kalman filter equations

$$\mathbf{x}(k + l) \approx \hat{\mathbf{x}}(k + 1) + \mathbf{A}[\mathbf{x}(k) - \hat{\mathbf{x}}(k)] + \mathbf{B}\mathbf{w}(k), \tag{9}$$

$$\mathbf{z}(k) \approx \hat{\mathbf{z}}(k) + \mathbf{C}[\mathbf{x}(k) - \hat{\mathbf{x}}(k)] + \mathbf{D}\mathbf{v}(k), \tag{10}$$

where $\mathbf{x}(k+1)$ and $\mathbf{z}(k)$ are the actual states and measurement vectors, $\hat{\mathbf{x}}(k+1)$ and $\hat{\mathbf{z}}(k)$ are the approximation state and measurement vectors, $\hat{\mathbf{x}}(k)$ is an estimate for the state $x(k)$, $\mathbf{A}, \mathbf{B}, \mathbf{C}$ and \mathbf{D} are Jacobian matrices,

$$\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k), \mathbf{u}(k)),$$

$$\mathbf{B} = \frac{\partial \mathbf{f}}{\partial \mathbf{w}}(\hat{\mathbf{x}}(k), \mathbf{u}(k)),$$

$$\mathbf{C} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}}(\hat{\mathbf{x}}(k), \mathbf{p}_i),$$

$$\mathbf{D} = \frac{\partial \mathbf{h}}{\partial \mathbf{v}}(\hat{\mathbf{x}}(k), \mathbf{p}_i),$$

Let

$$\Delta \mathbf{x}(k) = \mathbf{x}(k) - \hat{\mathbf{x}}(k) \quad (11)$$

$$\Delta \mathbf{z}(k) = \mathbf{z}(k) - \hat{\mathbf{z}}(k) \quad (12)$$

be the prediction error and the measurement residual at time k , then the errors at time $k+1$ can be expressed as

$$\Delta \mathbf{x}(k+1) \approx \mathbf{A}(x(k) - \hat{\mathbf{x}}(k)) + \mathbf{w}'(k), \quad (13)$$

$$\Delta \mathbf{z}(k) \approx \mathbf{H}\Delta \mathbf{x}(k) + \mathbf{v}'(k), \quad (14)$$

where $\mathbf{w}'(k)$ and $\mathbf{v}'(k)$ are new independent random variables with zero-mean and covariance matrices \mathbf{BQB}^T and \mathbf{DRD}^T as a white noise. Notice that Eqs. (13) and (14) are linear.

Now we can use the actual measurement residual $\Delta \mathbf{z}(k)$ in Eq. (12) and the second Kalman filter to estimate the prediction error $\Delta \mathbf{x}(k)$ given by Eq. (13). Let $\hat{\mathbf{x}}(k)$ be the estimation, then

$$\hat{\mathbf{x}}(k+1) = \hat{\mathbf{x}}(k) + \Delta \mathbf{z}(k) = \hat{\mathbf{x}}(k) + \mathbf{K}(k)(\mathbf{z}(k) - \hat{\mathbf{z}}(k)). \quad (15)$$

This equation can now be used for the measurement update in the Eq. (9), with $\hat{\mathbf{x}}(k)$ and $\hat{\mathbf{z}}(k)$ coming from Eqs. (7) and (8). The matrix \mathbf{K} is the gain that minimizes the posteriori error covariance,

$$\mathbf{K} = \mathbf{P}(k)\mathbf{H}^T(k)(\mathbf{H}(k)\mathbf{P}(k)\mathbf{H}^T(k) + (\mathbf{V}(k)\mathbf{R}\mathbf{V}^T(k))^{-1})^{-1}. \quad (16)$$

The covariance estimate from time step k to step $k+l$ is given by

$$\mathbf{P}(k+1) = \mathbf{A}(k)\mathbf{P}(k)\mathbf{A}^T(k) + \mathbf{B}(k)\mathbf{QB}^T(k), \quad (17)$$

where $\mathbf{A}(k)$ and $\mathbf{B}(k)$ are process Jacobian matrices at time k , and \mathbf{Q} is the process noise covariance at time k . The estimation accuracy now comes out by calculating the covariance estimates at a suitable interval.

4 Navigation Experiments

Experiments are made in a corridor with some convex edges, which are along to the corridor with distances 8 meters. When the robot moves, it can detect the convex edge using its dead reckoning system. The region is called sensing zones whose locations are known in advance as a map information. The robot moves along the planned path according to the measuring information from the dead reckoning from start to a sensing zone. Then the robot rotates the sensors at an angle to get the information from sensors as it moves.

The navigation is divided into the following steps:

1) The planned path from the current point to the next sensing zone is a straight line. Navigating the robot between these two points is performed based on the dead reckoning;

2) Detecting a landmark (a convex edge) by the combined sensor. The robot receives the position and the direction information of a convex edge as a landmark from an environmental map, and then calculates the expected distance and direction from the robot to the convex edge. Using these information, the sensors on the robot that are used to detect the convex edge tries to face the convex edge and measure the distance and direction.

3) After detecting the convex edge, the robot uses the measured values and the dead reckoning information to correct its estimated position. When the estimated position is generated, the robot updates the dead reckoning, and then has a new plan path regarding the next sensing point as a new goal.

Let the sensed convex edge be located at (x_i, y_i) , r the distance and α the angle. Then we have

$$r = \sqrt{(x(k) - x_i)^2 + (y(k) - y_i)^2}, \tag{18}$$

$$\theta = \tan^{-1} \frac{y(k) - y_i}{x(k) - x_i} - \alpha. \tag{19}$$

Referring to Eq. (6) the information from the combined sensor can be rewritten as

$$\mathbf{h}[\mathbf{x}(k), \mathbf{p}_i] = [(y(k) - y_i) - (x(k) - x_i) \tan(\theta(k) + \alpha)]. \tag{20}$$

According to Eq. (8) we can calculate the estimated position and the covariance matrix with the proposed method. The parameters used in the equation are obtained from the following equations,

$$\tilde{\mathbf{z}}(k) = (\tilde{y}(k) - y_i) - (\tilde{x}(k) - x_i) \tan(\tilde{\theta}(k) - \tilde{\alpha}), \tag{21}$$

$$\mathbf{B} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} = \left[-\tan(\tilde{\theta}(k) + \tilde{\alpha}) \quad 1 \quad \frac{\tilde{x}(k) - x_i}{\cos^2(\tilde{\theta}(k) - \tilde{\alpha})} \right], \tag{22}$$

$$\mathbf{D} = \frac{\partial \mathbf{h}}{\partial \mathbf{v}} = \left[\tan(\tilde{\theta}(k) + \tilde{\alpha}) \quad -1 \quad \frac{\tilde{x}(k) - x_i}{\cos^2(\tilde{\theta}(k) - \tilde{\alpha})} \right]. \tag{23}$$

By combining two estimations from dead reckoning and external sensor, the robot position becomes more accurate than the original estimation. Then the

robot updates the dead reckoning using the result of Eq. (9). The dead reckoning is updated at every sensing point.

In the experiments, the robot is controlled to follow the given path. By correcting the estimated position, the robot changes its path in the following order: The robot moves at a straight line at a distance 47.7meters along with four convex edges as landmarks. Then, the robot turns back along an arc with 3 meters radius, and then goes back to the start point using the same landmarks. Thus the robot is able to return back to the starting print within about 10cm, the 0.2% of the total distance traveled.

5 Conclusions

The method to use the dead reckoning to estimate the position of a robot is described. Then the estimated position of the robot is corrected by the observations of landmarks based on Extend Kalman Filter. Experiments are made with EKF in a corridor environment at a long distance, where the mobile robot was controlled to follow a rout of over 100m in length. The system uses convex edges as landmarks and combined sensors for self-guiding. The results show that natural landmarks such as convex edges are useful to correct the robot position by the proposed method.

References

1. Kidono, K., Miura, J., Shirai, Y.: Autonomous navigation of a mobile robot using a Human-Guided experience. *Proc.Ssian conf.on Computer Vision*. (2000) 449–454.
2. Moon, I., Miura, J., Shirai, Y.: Automatic extraction of visual landmarks for a mobile robot under uncertainty of vision and motion. *Proc.2001 IEEE Int. Conf.on Robotics and Automation*. (2001) 1188–1193.
3. Roumeliotis, S., Bekey, G.: Bayesian estimation and Kalman filtering: A unified framework for mobile robot localization. In *Proc.2000 IEEE International Conference on Robotics and Automation, San Francisco, CA, Spril 22-28*. (2000) 2985–2992.
4. Goel, P., Dedeoglu, G., Roumeliotis, S.: Fault detection and identification in a mobile robot using multiple model estimation and neural network. In *Proc.2000 IEEE International Conference on Robotics and Automation, San Francisco, CA, April 22-28*. (2000) 2302–2309.
5. Fengji, Zhao., Hai-jiao, Guo., Ken-Ichi, ABE.: Position location for a mobile robot using sonar sensors. *IEEE International Conference on Intelligent Engineering Systems Vienna, Austria*. (1998) 130–135.
6. Beom, H., Cho, H.: Mobile robot localization using a single rotating sonar and two passive cylindrical beacons. *Robotica*. **13** (1995) 243–252.
7. Fengji, Zhao., Hai-jiao, Guo., Ken-Ichi, ABE.: Precise localization for a mobile robot with sonar and infrared sensors. *IASTED International Conference Robotics and Manufacturing Banff, Canada*. (1998) 8–12.

Simulated Annealing with Injecting Star-Alignment for Multiple Sequence Alignments

Hongwei Huo and Hua Ming*

School of Computer Science and Technology,
Xidian University, Xi'an 710071, China
hwhuo@mail.xidian.edu.cn

Abstract. We present a novel algorithm SASAlignSimulated annealing with star-alignment. In the SASAlign, instead of starting with an initial solution chosen at random, we use the results formed by star-alignment to give a good starting point as the initial solution to the SA for further refinement. The time required by the algorithm scales linearly with the number of sequences in S , linearly with the number of iterations, and cube with the length of the sequences, that is $O(Nnl^3)$. Experiments on the BALiBASE benchmark database also show that the proposed algorithm is efficient, and prove to be competitive with and better than the other method HMMT.

1 Introduction

The multiple sequence alignment is one of the challenging tasks in bioinformatics. It plays an essential role in finding conserved region among a set of sequences and inducing the evolutionary history and common properties of some species[1,2]. It is known to be NP-complete and the current implementations of multiple alignment algorithms are heuristics owing to the high complexity on space and time needed when performing alignments[3].

Most existing algorithms for the multiple sequence alignment are classified into three categories[4]. The first class is those algorithms that use high quality heuristics very close to optimality[5,6]. They can only handle a small number of sequences with length less than 20 and limited to the sum-of-pairs objective function. The second class is those algorithms using progressive alignments strategy and is by far the most widely used heuristic to align a large number of sequences[7,8]. The third class is those algorithms using iterative refinement strategies to improve an initial alignment until convergence or reaching the maximum user-defined number of iterations[9].

In this paper, we present a novel algorithm SASAlignSimulated annealing with star-alignment, in which star-alignment was injected into simulated an-

* Supported by the National Natural Science Foundation of China under Grant Nos.69601003.

nealingSAalgorithm for multiple sequence alignment to enhance the ability to search optimal solutions to the problem. Experiments on the BALiBASE benchmark database show that the proposed algorithm is efficient, and prove to be competitive with and better than the other method HMMT in the regions of low similarity.

2 Preliminaries

2.1 Multiple Sequence Alignments

Given a finite alphabet set Σ and a family $S = (s_1, s_2, \dots, s_n)$ of n sequences of various length l_1 to l_n : $s_i = s_{i1}s_{i2} \dots s_{il_i}$, $s_{ij} \in \Sigma, 1 \leq j \leq l_i, 1 \leq i \leq n$, where for DNA sequences, Σ consists of 4 characters {A,C,G,T}, and for protein sequences, Σ consists of 20 characters of amino acids. An alignment of S is a matrix $A = (a_{ij})$, where $(1 \leq i \leq n), (1 \leq j \leq l), \max(l_i) \leq l \leq \sum_{i=1}^n l_i$ and satisfying: (1) $a_{ij} \in \Sigma \cup \{.\}$, here "." denoting the gap letters; (2) each row $a_i = a_{i1}a_{i2} \dots a_{il}(1 \leq i \leq n)$ of A is exactly the corresponding sequence s_i if we eliminate all gap letters; (3) A has no column which only contains gaps.

2.2 Objective Function

Evaluation of the alignments is made using an objective function(OF) which is a measure of multiple alignment quality. We use one OF related to the sums of pairs score(SPS) with affine gap penalties. The object function of an alignment A can be expressed as follows:

$$SCORE(s_i, s_j) = \sum_{k=1}^{\min(l_i, l_j)} d(s_{ik}, s_{jk}) + G \cdot num_{ij}$$

$$SCORE(A) = \sum_{i=2}^n \sum_{j=1}^{l-1} SCORE(s_i, s_j)$$

where $d(s_{ik}, s_{jk})$ is the score for each aligned pair of residues, G is gap penalty, num_{ij} is the number of gaps in the pairwise alignment of rows s_i and s_j . If an alignment A' of S satisfying:

$$SCORE(A') = \max_A(SCORE(A))$$

we call A' the optimal alignment.

2.3 Simulated Annealing

The simulated annealing (SA for short) for the combinatorial optimization problems first presented for the TSP problem in Science by Kirkpatrick. It develops a general strategy for solving discrete optimization problems. The basic idea

behind simulated annealing is that the search process can escape local minima if we allow some uphill moves. Starting with an initial solution s_0 we choose at random a neighbor of b . If that move improves the cost function we perform it; otherwise we move to that solution with probability $p = e^{-\Delta/t}$, where Δ is the deterioration in the cost function. The parameter t , which is called temperature, is used in order to control the acceptance ratio.

3 Algorithms

We present a novel algorithm SASAlign, in which star-alignment was injected into simulated annealingSAalgorithm for multiple sequence alignment to enhance the ability to search optimal solutions to the problem. In the SASAlign, instead of starting with an initial solution chosen at random, we use the results formed by star-alignment to give a good starting point as the initial solution to the SA for further refinement.

3.1 Star Alignment

The idea of the star alignment (SAlign) algorithm is to first determine the optimal pairwise alignment of each sequence in S to a sequence s in S selected at random using dynamic programming, and then to construct a multiple sequence alignment consistent with these pairwise alignments. The algorithm is described in Fig. 1. where s can be any sequence in S , $ns_i(k)$ be the number

```

SAlign(S)
1  for  $i \leftarrow 1$  to  $n$  do
2    Find the optimal pairwise alignment  $a_i$  of  $s_i$  and  $s$ 
3  for  $k \leftarrow 0$  to  $length(s)$  do
4     $ns(k) \leftarrow \max_{i=1..n} ns_i(k)$ 
5  Construct a new matrix  $A'$  for each pairwise alignment  $a_i$ ,  $1 \leq i \leq n$ .
6  Insert  $ns(0) - ns_i(0)$  column of spaces in front of  $a_i$ .
7  Insert  $ns(k) - ns_i(k)$  column of spaces immediately after character  $k$  of  $a_i$ , for  $0 < k \leq length(s)$ .
8  return  $A'$ 
    
```

Fig. 1. The star alignment (SAlign) algorithm

of gaps inserted into a_i corresponding to s between characters k and $k + 1$ of s for $0 \leq k \leq length(s)$, for each pairwise alignment a_i , $i = 1 \dots n$. $ns_i(0)$ be the number of spaces inserted in front of s and let $ns_i(length(s))$ be the number of spaces inserted at the end of s . Lines 1-2 find the optimal pairwise alignment a_i of s_i and s with respect to the function $SCORE(s_i, s)$ using dynamic programming, for $1 \leq i \leq n$. Lines 3-4 compute $ns(k)$ which is defined by $\max_{i=1..n} ns_i(k)$. Line 5-7 construct a new matrix A' for each pairwise alignment a_i .

3.2 Simulated Annealing with Star-Alignment for MSA

In the algorithm of SASAlign, instead of starting with an initial solution chosen at random, we use the multiple sequence alignment formed by SAlign to give a good starting point as the initial solution to the SA for further refinement. Then by substitution operation progressively a sequence was obtained, which had the maximum sum of scoring value. The algorithm is described in Fig. 2. where the

```

SASAlign(S)
1 Initialize  $t_0$  and  $\alpha$ 
2 Produce  $A^{(0)}$  by invoking SAlign
3 for  $k \leftarrow 0$  to  $N$  do
4   for  $j \leftarrow 1$  to  $n$  do
5     for  $i \leftarrow 1$  to  $length(a_j^{(k)})$  do
6       for each character  $c \in \Sigma$  do
7         Construct a sequence  $s(c)$  by substituting letter  $c$  for the  $i$ th character of  $a_j^{(k)}$ 
8          $Val(s(c)) \leftarrow \sum_{i=1}^n SCORE(s_i, s(c))$ 
9        $Val(a_j^{(k)}) \leftarrow \sum_{i=1}^n SCORE(s_i, a_j^{(k)})$ 
10      Pick a sequence  $s^*$  in  $\{s(c) | c \in \Sigma\}$  with the maximum scoring value
11      if  $Val(s^*) > Val(a_j^{(k)})$ 
12        then  $a_j^{(k)} \leftarrow s^*$ 
13      else  $a_j^{(k)} \leftarrow s^*$  with probability  $\exp(-(Val(a_j^{(k)}) - Val(s^*)) / t_k)$ 
14       $t_{k+1} \leftarrow \alpha t_k$ 
15 return  $A^{(N)}$ 

```

Fig. 2. Algorithm SASAlign

temperature for iteration k be t_k . At the end of each iteration, the temperature is decreased by a factor a in the range of $0 < a < 1$. N is the number of iterations. We take $t_0 = 1/a = 0.95$ in the experiments.

$SCORE(s_i, a_j)$ is evaluated using the dynamic programming, which is proved as an efficient method for pairwise alignment, for each aligned pair of residues based on the substitution BLOSUM matrix. We use BLOSUM30 as substitution matrix for the similarity of residues less than 30% and BLOSUM45 as substitution matrix for the similarity of residues greater than 30%.

The length of a sequence is limited to $1.2l_{max}$, if one sequence's length exceeds the limitation, then discard it. The choice of 1.2 as a scaling factor was based on the observation that solutions to common alignment problems rarely contained more than 20 percent gaps.

3.3 Analysis of SASAlign

To predict the complexity of the overall algorithm, consider firstly the star-alignment algorithm. Each computation of optimal pairwise alignment a_i requires an edit distance to be computed for each i , $1 \leq i \leq n$. This can be done

in time proportional to the product of the lengths of the two sequences by dynamic programming. Thus, the **for** loop of lines 1-2 in SAlign takes $O(\sum_{i=1 \dots n} l_i \cdot length(s)) = O(n \cdot length(s) \max_{i=1 \dots n} l_i)$ time. Obviously, the **for** loop of lines 3-4 takes $O(n \cdot length(s))$ time. The **for** loop of lines 5-7 takes $O(n \cdot length(s))$ time. Thus, the overall time of the algorithm SAlign is $O(n \cdot length(s) \cdot \max_{i=1 \dots n} l_i)$.

Next, we analyze the complexity of the algorithm SASAlign. The SASAlign procedure contains four nested **for** loops. Lines 5-8 require $O(N \cdot n \cdot length(a_j) \cdot |\Sigma| \cdot l_i \cdot length(s(c)))$ time, line 9 takes $O(N \cdot n \cdot l_i \cdot length(a_j))$ time, and line 10 takes $O(N \cdot n \cdot |\Sigma|)$ time, lines 11-13 take $O(N \cdot n)$ time, line 14 takes $O(N)$ time. Adding the time required in line 2 (its analysis in the last paragraph), the overall time of the algorithm SASAlign is $O(N \cdot n \cdot length(a_j) \cdot |\Sigma| \cdot l_i \cdot length(s(c)))$ time, where we have ignored the lower order terms.

If we further assume that l is the maximum length of the sequences and $|\Sigma|$ is a constant, then this expression can be simplified to $O(N \cdot n \cdot l^3)$.

4 Experimental Results

The algorithm was tested on the data of benchmarks from the BALiBASE2.0. The results of the experiment applying SASAlign and HMMT to the same sequences were shown in Table 1. The first column gives the family name of being tested sequences, the second and third respectively contain the maximum length and average percent similarity of sequences being tested. We validated this algorithm from the alignment it formed, and compared it with the model of HMMT which uses simulated annealing algorithm in the iterative training procedure. The accuracy was evaluated by the sum-of-pairs score (SPS), which indicates the ratio of pairs correctly aligned.

Table 1. Results of two methods of SASAlign and HMMT on the same data

family sequence name	max sequence length	average similarity	SASAlign	HMMT
cytochrome c	87	27	0.478	0.346
cytochrome e	127	29	0.433	0.217
ribosomal protein l1	213	30	0.524	0.314
anthranilate isomerase	284	32	0.533	0.193
1pkm	449	34	0.709	0.748
reverse transcriptase	541	42	0.748	0.856
serine protease	245	43	0.757	0.742
plastocyanin	99	46	0.737	0.797
papain	220	46	0.786	0.856
Triose phosphate isomerase	254	49	0.890	0.941
Cold shock protein	70	51	0.805	0.776
phosphoglucomutase	567	51	0.825	0.844
Ribonuclease	103	57	0.902	0.832

The results from the experiments show that our method is better than HMMT on the sequences with similarity 27%, 29%, 30%, 32%, 43%, 51%, and 57%, especially for the sequences with the lower similarities. The results indicate that SASAlign works efficiently, because the distantly related sequences often have only short conserved motifs in long regions of low overall similarity and hardly to identify.

So we conclude that the proposed algorithm is efficient, and prove to be competitive with and better than the other method HMMT as a whole, especially for the sequences with low similarities.

5 Conclusion

In this paper, a novel algorithm SASAlign was presented, in which star-alignment was injected into simulated annealing (SA) algorithm for multiple sequence alignment to enhance the ability to search optimal solutions to the problem. In the SASAlign, instead of starting with an initial solution chosen at random, we use the results formed by star-alignment to give a good starting point as the initial solution to the SA for further refinement. Other heuristics algorithms can be used as the seed to improve the algorithm. In our future work, we will not only use the substitution operation to construct a new sequence as described in the paper, but also introduce Insertion and Deletion operations. Therefore good search capacity will intensify the performance of simulated annealing.

References

1. Neil C. Jones, Pavel A. Pevzner: An Introduction to Bioinformatics Algorithms. MIT Press (2004)
2. D. Gunsfield: Algorithms on Strings, Trees and Sequences. Cambridge University Press(1999)
3. Hongwei Huo: Exercises and Solutions on Algorithms. Higher Education Press, China(2004)
4. Cedric Notredame: Recent progresses in multiple sequence alignment: a survey. Pharmacogenomics, 3(2002) 131-144
5. Carrillo H., Lipman,D.J.: The multiple sequence alignment problem in biology. SIAM Appl. Math, 48(1988) 1073-1082
6. Pearson,W.R., Lipman,D.J.: Improved tools for biological sequence comparison. Proceedings of the National Academy of Sciences of the USA, 4(1988) 2444-2448
7. D.F. Feng, R.F. Doolittle: Progressive sequence alignment as a prerequisite to correct phylogenetic trees. Molecular Evolution, 25(1987) 351-360
8. Arun Siddharth Konagurthu, James Whisstock, Peter J. Stuckey: Progressive multiple alignment using sequence triplet optimizations and three-residue exchanges costs. Journal of Bioinformatics and Computational Biology, 2(2004) 719-745
9. Notredame C., Higgins D.G.: SAGA: sequence alignment by genetic algorithm. Nucleic Acids Research,24(1996) 1515-1524

A Noise-Insensitive Hierarchical Min-Max Octree for Visualization of Ultrasound Datasets

Sukhyun Lim, Kang-hee Seo, and Byeong-Seok Shin

Inha University, Dept. Computer Science and Information Engineering,
253 Yonghyun-dong, Nam-gu, Incheon, 402-751, Rep. of Korea
{slim, PlasticFlower}@inhaian.net, bsshin@inha.ac.kr

Abstract. There are two important factors to visualize ultrasound datasets for volume ray casting method. The first is an efficient method to skip over empty space and the second is an adequate noise filtering method. We propose a noise-insensitive hierarchical min-max octree. In preprocessing stage, we generate a filtered dataset and make a hierarchical min-max octree from the dataset. In rendering step, we exploit the hierarchical min-max octree only when rays skip over transparent region. If rays reach meaningful object, color and opacity values are computed from the original volume dataset. By adaptively using two datasets, our method increases image quality while reducing rendering time.

1 Introduction

The visualization of ultrasound datasets is a technology for imaging ultrasonic echo information. It is mostly used in obstetrics and gynecology [1], and for visualization of vessels and tumors in soft tissue. The main advantage is that they are non-radiative and inexpensive. In addition, the acquisition procedure is faster than other medical imaging methods such as CT and MR. However, the visualization is difficult since they typically contain a lot of speckle noises. To solve the problem, several filtering techniques are used to separate useful information from the noises [2], [3], [4]. However, those methods influence not only speckle noises but also interesting object. Therefore, images are distorted since the color values are computed from the filtered dataset.

We propose a noise-insensitive hierarchical min-max octree to skip over empty space efficiently without distortion of image quality. In preprocessing step, we generate a filtered dataset by applying a filtering operation from the original volume. Then, we make a hierarchical min-max octree based on the filtered dataset. In rendering step, we exploit the hierarchical min-max octree to skip over transparent region without affecting noises. When a ray reaches interesting object, the original dataset is used to acquire color and opacity values. By using two datasets (filtered and the original dataset) adaptively, we generate natural results while reducing rendering time.

In Sect. 2, we look into the problems of the hierarchical min-max octree when we use it to visualize ultrasound datasets. Then, we explain our data structure in detail in Sect. 3. Experimental results are shown in Sect. 4. Finally, we conclude our work.

2 Problems of a Hierarchical Min-Max Octree for Visualizing Ultrasound Datasets

We exploit the volume ray casting method and a hierarchical min-max octree (HMMO). A HMMO is a data structure composed of several min-max blocks generated during preprocessing stage. Each block holds the minimum and the maximum voxel values representing a block. We call the “transparent block” as a block of which the minimum and the maximum values are confined within transparent range. When a sample point lies on transparent block, a ray jumps over the block.

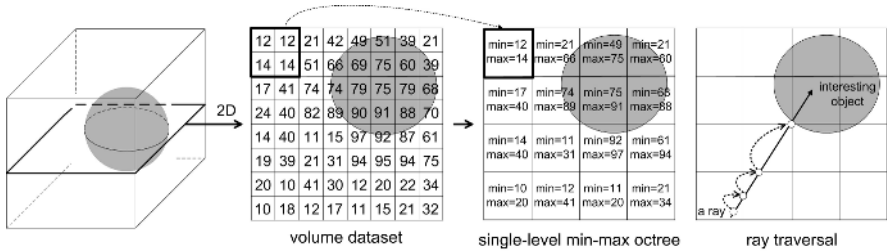


Fig. 1. A structure of the single-level min-max octree. Each block has the minimum and the maximum values for representing the block. Dotted arrow represents how to jump over empty space using a HMMO.

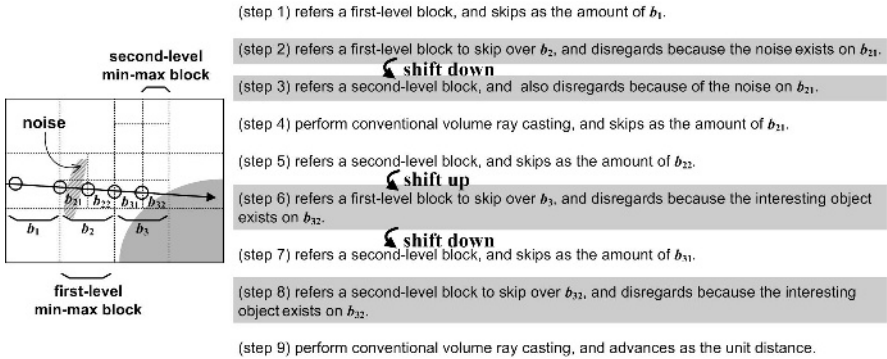


Fig. 2. When we apply a HMMO for space-leaping, unnecessary comparison (shaded box) and level shift (thick arrow) between blocks are increased due to the noises

One of the main obstacles to visualize ultrasound datasets is speckle noises. When the noises exist on a ray path, two problems may arise. At first, when an octree hierarchy is increased, unnecessary comparison and level shift between blocks increases due to the noises. Since a ray does not have prior information of the noises, it interprets them as interesting objects. Therefore, rendering time may increase (see Fig. 2).

The second problem occurs when the density value of noises on a ray path is confined within opaque range. In this case, the color computation step of a pixel is terminated in the noisy region because the opacity value can reach 1.0. Although the

density value of the noises is confined in translucent range, color and opacity values are affected by the noises. Fig. 3 (b) and (c) depicts each problem. The frequency of level shift is increased since ultrasound datasets contain lots of noises.

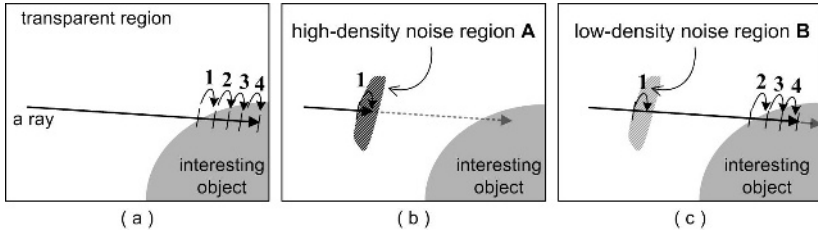


Fig. 3. (a) Assume that four times compositions are performed from an interesting object boundary if there are no noises. (b) When the density value of the noise is confined in opaque range, the color composition can be ended in the high-density noise region **A**. (c) Even if the density value is confined within translucent range, we acquire unexpected color values since the accumulated values are already affected by the low-density noise region **B**.

3 A Noise-Insensitive Hierarchical Min-Max Octree

In order to solve two problems of conventional HMMO, we exploit a filtered dataset generated in preprocessing step. When we use the 5^3 average filter, one voxel searches neighboring 124 ($=5^3-1$) voxels. This computation performs for all voxels. If the speckle noises exist locally, the amount of noises is reduced because they are spread over 125 voxels.

This approach efficiently reduces the noises. However, it can affect to the entire volume dataset. Therefore, we obtain deteriorated images since even interesting objects are computed from the filtering dataset. If we apply the noise filtering not to entire volume but to noisy regions, the final image are not affected. However, since the density values of them are nearly identical [2],[3], rays cannot distinguish them.

After firing a ray from each pixel, the minimum and the maximum values of a min-max block generated from a filtered dataset are referenced to determine whether to skip over the block or not. When the values are confined within transparent range, the ray jumps over the block. Since a HMMO is entirely exploited to skip over empty space, we use it only when a ray traverses transparent region. To skip over transparent blocks efficiently, we use the *distance template* scheme [5],[6] proposed by Lim and Shin. It quickly and reliably leaps to the boundary of a transparent block since it can directly access the distance value to reach the boundary. If the ray arrives at interesting object, we exploit the original volume dataset to compute color and opacity values. This approach introduces accurate rendering result compared with the previous filtered methods because the calculations of color and opacity values are accomplished from the original dataset. As a result, we exploit two datasets adaptively.

Fig. 4 shows a ray-traversal example of our method. The traversal condition is identical to Fig. 2 and 3. In transparent region, a ray path through without

compositions since the noise is estimated as a transparent object by a filtered dataset (see Fig. 4(b)). When the ray reaches interesting object, we exploit the original volume dataset (see Fig. 4(c)). Fig. 5 depicts a ray-traversal procedure after applying a filtering operation. Compared with the case of Fig. 2, total traversal steps are reduced from nine to six.

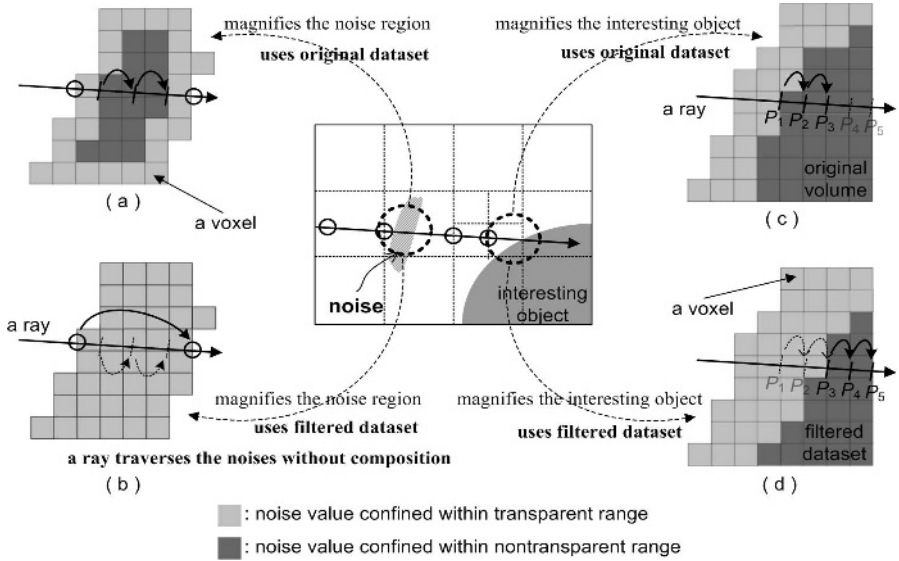


Fig. 4. (a) Since a noise value is confined to nontransparent range, the ray starts compositions in the noise region. In this case, we acquire unexpected result. (b) After the filtering operation, the noise value is confined within transparent range. The ray traverses the noise region without compositions. (c) It computes color and opacity values from accurate point P1 since it refers to the original volume dataset (d) When we exploit the filtered dataset, the ray starts a composition from inaccurate point P3 because the filtering is applied to entire volume dataset. Although the voxels P1 and P2 are meaningful sample point, the ray skips those samples.

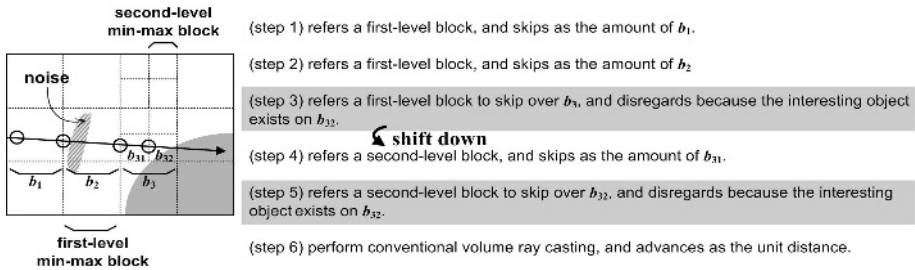


Fig. 5. After applying noise filtering, the number of unnecessary comparison (shaded box) is reduced (4→2), and the number of shift between blocks is reduced from (3→1)

4 Experimental Results

Our method was implemented on a PC equipped with a Pentium™ IV 3.06 GHz CPU and 1 GB of main memory. Volume dataset was obtained by scanning a fetus with an ultrasound of which the resolution is $256 \times 256 \times 128$. We set the maximum level of HMMO as three. We experimented on the Euclidean distancemap method [7], [8], [9], to compare with other space-leaping methods.

Table 1 shows the rendering time. Rendering speed of our method is about four times faster than that of the conventional volume ray casting method. Compared with the distancemap-based method, our method is about 10% slow. However, since the distancemap-based approach requires huge amount of memory and long preprocessing time, it is not suitable for application using ultrasound datasets.

Table 1. Comparison of rendering time. Image size is 256×256 pixels

Method	preprocessing time (sec)	rendering time (sec)
conventional volume ray-casting	-	3.45
Euclidian distancemap	17.23	0.64
our method	0.24	0.72

Fig. 6 shows image quality without change of viewpoint and viewing direction. The images are the results when we apply the 5^3 average filter, our method, and the 7^3 average filter from left to right. Of course, in the case of using the 5^3 and the 7^3 filter, the color values are computed from the filtered dataset. As you can see, the left column image contains a lot of noises. The skin is represented roughly and bumpily. As the filter size increases, the result appears smoothly. However, the image generated by our method shows natural and clear result while reducing noises.

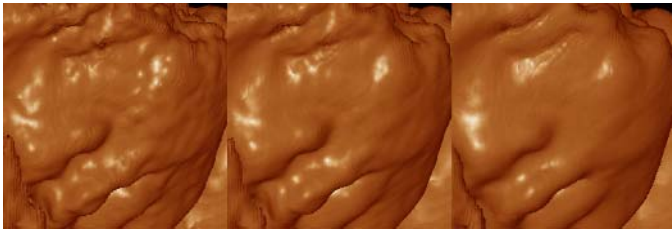


Fig. 6. Comparison of image quality. The images show the rendering results using the 5^3 filter, our method, and the 7^3 filter, from left to right.

To verify that our method is still efficient compared with other filtering approaches, we implement the [4]. This method also exploits the average filter, and they use two filtered datasets for the re-sampling and for the gradient estimation. That is, small-sized filter is used for re-sampling, and large-sized filter is exploited to estimate gradient vectors. Fig. 7 shows the comparison of the results with our method (left) and [4] (right). The rendering result is nearly identical. However, since the [4] requires two filtered dataset, preprocessing time is much longer than that of our method.

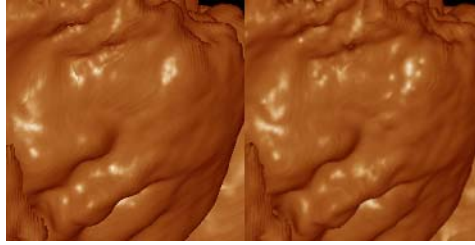


Fig. 7. The left images are generated by our method (exploit the 7^3 filter to generate a HMMO, and the original dataset to compute color value), and the right one is the result generated by [4] (exploit the 5^3 filter for space leaping, and the 7^3 filter for gradient estimation)

5 Conclusion

To achieve interactive speed for ultrasound datasets contained lots of speckle noises while preserving image quality, we present a noise-insensitive HMMO method. When a ray traverse transparent region, it use a HMMO generated from a filtered dataset. This can accelerate rendering speed without affecting noises. When it reaches interesting object, the original dataset is exploited to acquire color and opacity values. This introduces accurate results because it refers the original volume dataset not filtered. By using two datasets, our method produces natural images and takes less time for rendering. (All data are supported by Medison, co., ltd.)

References

1. Baba, K., Jurkovic, D.: Three-Dimensional Ultrasound in Obstetrics and Gynecology. Progress in Obstetric and Gynecological Sonography Series. The Parthenon Publishing(1997)
2. Sakas, G., Walter, S.: Extracting Surfaces from Fuzzy 3D-Ultrasound Data. In R. Cook, editor. Proc. SIGGRAPH 1995 (1995) 465-474
3. Sakas, G., Schreyer, L.A., Grimm, M.: Preprocessing and Volume Rendering of 3D Ultrasonic Data. IEEE Computer Graphics and Applications, Vol. 15, No. 4 (1995) 47-54
4. Shamdasani, V., Bae, U., Managuli, R., Kim, Y.: Improving the Visualization of 3D Ultrasound Data with 3D Filtering. Proc. SPIE, vol. 5744 (2005) 455-461
5. Lim, S., Shin, B.: Reliable Space Leaping Using Distance Template. Lecture Notes in Computer Science, Vol. 3337 (2004) 60-66
6. Lim, S., Shin, B.: RPO: A Reverse-Phased Hierarchical Min-Max Octree for Efficient Space-Leaping. Proc. Pacific Graphics 2005, to be published
7. Zuiderveld, K., Koning, A., Viergever, M.: Acceleration of Ray-casting Using 3D Distance Transforms. Visualization in Biomedical Computing 1992 (1992) 324-335
8. Yagel, R. and Shi, Z.: Accelerating Volume Animation by Space-leaping. Proc. IEEE Visualization 1993 (1993) 62-69
9. Sramek, M., Kaufman, A.: Fast Ray-tracing of Rectilinear Volume Data Using Distance Transforms. IEEE TVCG, vol.6, no.3 (2000) 236-252

A Novel Fusing Algorithm for Retinal Fundus Images

Bin Fang, Xinge You, and Yuan Yan Tang

College of Computer Science, Chongqing University, Chongqing 400044, China
{fb, xyou, yytang}@cqu.edu.cn

Abstract. In this paper, a novel fusing method for fundus retinal images based on robust registration techniques is proposed. In order to construct precise fusion map, we apply a ‘coarse-to-fine’ mapping strategy to accurately align pairs of identified vascular trees of retinas. A coarse mapping algorithm that exploits rigid model is first performed to maximize the goodness of fit between the vascular features over two time periods. However, the results suffer from local misalignment due to the inherent imprecise characteristics of the simplified model. A fine mapping algorithm is employed to eliminate ‘ghost vessels’ based on a local elastic matching technique. The transformed vectors for pixels in the registered fundus image are conveniently calculated by combining the local move vector and the global model transformed vector. Experiment results demonstrate nearly perfect fusion maps of several retinal fundus images in terms of visual inspection.

1 Introduction

Diabetic retinopathy is a serious complication of diabetes mellitus and a major cause of blindness worldwide. Early detection and timely treatment of diabetic retinopathy is possibly able to halt or even reverse the progress of the disease and prevent blindness [1]. For this reason, a patient is often screened at regular intervals where a sequence of digital fundus images are produced. By comparing these images taken at different time periods, physicians can evaluate the progress of the diseases and decide on the appropriate treatments to be taken. Fusing fundus retinal image is helpful to provide valuable information in this context.

Fusing a number of images is usually based on the correct registration between different pairs of images. Alignment methods using the full retinal fundus image content and mutual information as the similarity measure fail to deal with the registration of the total surface of eye fundus images [2]. Since vascular tree is more reliable to be extracted and more representative of the eye surface, it is expected to use tree structure of blood vessels as object features for retinal fundus registration [4,5,6]. Vascular trees are typically incorporated into well-established transformation models such as rigid [4], affine [5] and second order polynomial transformations [3]. The process of registration is equivalent to solving the problem of optimizing a function that measures the goodness of fit between the reference and the transformed images.

The sphere-like shape of human eye suggests a quadratic surface model for registration. Nevertheless, the computation complexity of model parameters may be a

disaster and the search strategy design is not trivial. Therefore, in this paper, we bring forward the idea of employing a ‘coarse-to-fine’ mapping strategy. First, extracted vascular trees are globally aligned using rigid model of translation and rotation. The adoption of comparative simple model enables us to compute the optimal transformation effectively and efficiently. However, one major demerit of employing the rigid model is that local alignment errors inevitably exist because of the inherent imprecise characteristics of the simplified model. In order to rectify the pitfall of local ‘ghost vessels’, we propose a structure-deformed elastic matching algorithm to improve registration accuracy.

2 A Coarse-to-Fine Mapping Strategy

It has been shown that employing a rigid registration model is able to greatly reduce the computation load without losing too much accuracy except some local misalignments for retinal registration. Following this important observation, our idea to solve the problem of fusing retinal images is to adopt a coarse-to-fine, two stage process algorithm. First, we adopt the simplified rigid model of translation and rotation for globally mapping two vascular trees. Second, a local elastic matching technique is used for fine mapping to eliminate local misalignments.

One of the two vascular features to be registered is called the *Template* and the other the *Input*. Thinning is performed for both the *Template* and the *Input* so that the resulting patterns consist of lines with one pixel width only. A sequential distance transformation (DT) is applied to create a distance map for the *Template* by propagation local distances [7]. The *Input* at different positions with respect to the corresponding transformations is superimposed on the *Template* distance map. A centreline mapping error (CME) to evaluate matching accuracy is defined as the average of feature points distance of the *Input* as follows:

$$CME = \frac{1}{N} \sum_{p(i,j) \in Input} DM_{Template}(p(i,j))^2 \quad (1)$$

N is the total number of feature points in the *Input*, $p(i,j)$ are the transformed positions of the original feature points in the *Input* and DM is the distance map created for the *Template* vascular features. It is obvious that a perfect match between the *Template* and *Input* images will result in a minimum value of CME.

The final optimal match is determined by the transformation which has the smallest centreline mapping error. Once the relative parameters for the global transformation model have been computed, the registration between two retinal images is ready. Refer to the following pseudo-code of search algorithm.

```

Algorithm Search
Begin
FOR each initial position of the template
 $X$  = X coordinate of initial position;
 $Y$  = Y coordinate of initial position;
 $\theta$  = rotation angle of initial position;
 $X''$  = step-length of X translation parameter;
 $Y''$  = step-length of Y translation parameter;

```

```

 $\theta''$  = step-length of rotation parameter  $\theta$ ;
dist_min = value of distance function at initial position;
LOOP
FOR x =  $X' - X''$ ,  $X' + X''$ , step =  $X''$ 
FOR y =  $Y' - Y''$ ,  $Y' + Y''$ , step =  $Y''$ 
FOR  $\theta = \theta' - \theta''$ ,  $\theta' + \theta''$ , step =  $\theta''$ 
    dist_current = value of distance function at position (x,y,  $\theta$ );
    IF ( dist_current < dist_min ) THEN
         $X' = x$ ;  $Y' = y$ ;  $\theta' = \theta$ ;
        LOOP_FLAG = 0;
        EXIT FOR x;
        EXIT FOR y;
        EXIT FOR  $\theta$ ;
    ELSE LOOP_FLAG = 1;
    ENDIF
ENDFOR //  $\theta$ 
ENDFOR // y
ENDFOR // x
IF (LOOP_FLAG = 1) EXIT LOOP
ENDLOOP
Optimal position = ( $X'$ ,  $Y'$ ,  $\theta'$ );
ENDFOR

```

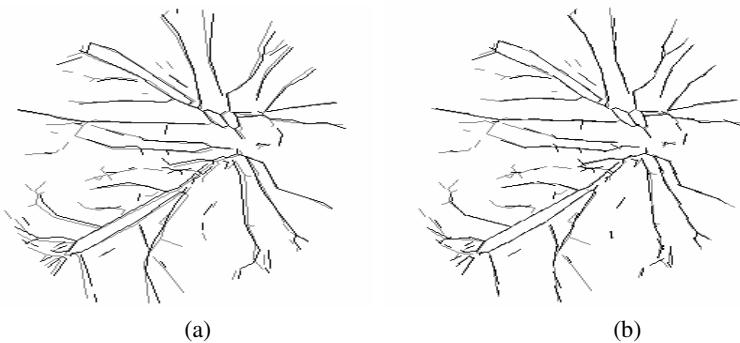


Fig. 1. (a) Overlapped image of *Input* pattern (black lines) and *Template* pattern (gray lines) before matching. (b) Overlapped image of patterns after matching.

While the coarse mapping approach is able to align retinal images globally, the local alignment errors inevitably exist because of the inherent imprecise characteristics of the simplified model. The phenomenon of ‘ghost vessels’ that is more obvious around boundaries of overlapped region is clearly perceptible. In order to rectify the pitfall of the rigid model, we propose a local elastic matching algorithm to further improve matching accuracy by eliminating the existence of ‘ghost vessels’.

Let *Template* and *Input* be two binary vascular trees. The *Template* is elastically deformed in order to match the *Input* locally until the corresponding elements of both *Input* and *Template* meet, as illustrated in Figure 1. The objective is to achieve local

alignment while to maintain the regional structure as much as possible. We elaborately create an energy function whose original format can be found in [8] to guide the deformation process.

$$E_1 = -K_1^2 \sum_{i=1}^{N_i} \ln \sum_{j=1}^{N_r} \exp\left(-|\mathbf{T}_j - \mathbf{I}_i|^2 / 2K_1^2\right) f(\theta_{T_j, I_i}) \tag{2}$$

$$+ \sum_{j=1}^{N_r} \sum_{k=1}^{N_r} w_{jk} (d_{T_j, T_k} - d_{T_j, T_k}^0)^2$$

For details of how to implement the local deform matching process, refer to [8].

3 Fusing Retinal Fundus Images

Once having identified correspondence of each feature point by local elastic matching method, we are able to calculate the transformed positions for registered fundus image. Assuming mid-point and two end points of each *Template* element as feature points, the elastic local move vector for each feature point is ready to calculate directly by referring to location in the *Template* and the matched location in the *Input*.

$$\mathbf{P}_j = \mathbf{T}_j - \mathbf{I}_{i(j)} \tag{3}$$

where \mathbf{T}_j is the move vector of feature point j of the *Template* and $\mathbf{I}_{i(j)}$ is the position vector of the feature point of *Input* which is matched to j . Therefore, local transformed position for each pixel of the registered retinal fundus can be computed as follows.

$$\Delta M = \sum_{j=1}^N w_j \mathbf{P}_j \tag{4}$$

N is the size parameter of the Gaussian window which establishes neighborhoods of correlation, w_j are the correlation weights in the Gaussian correlation window where $w_j = \exp\left(- (j-1)^2 / (2 \times N^2)\right)$ and $j = 1, 2, \dots, N$. \mathbf{P}_j is the local move vector of feature point j in the Gaussian window sorted in the order of increasing distance from that pixel.

Since we have applied the ‘coarse-to-fine’ strategy to achieve accurate alignment, the final transformed vectors for pixels in the input fundus image which is used to be registered to the reference fundus image are conveniently to be calculated by combining elastic local move vectors and global model-transformed vectors. Let the registered image I_q be matched to the reference image I_p . If the transformed position of a certain pixel P_q in I_q is overlapped with a pixel P_p in the retinal region of I_p , the gray density of pixel of the fusion map which has the same position as P_p is to be calculated by merging the gray densities of P_q and P_p . The merging strategy adopted in the experiment is the arithmetic average operation of the gray intensities. For those pixels of I_q whose transformed positions are not located inside the retinal region of

image I_p , they augment the fusion map with their original gray intensities to the transformed positions in the fusion map.

The construction of the fusion map starts with a reference sample which can be conveniently identified by human operator from a number of retinal images. Once a new image has been matched and fused with the reference, an extended vascular tree is built from registered vascular trees and is kept for the use for fusing the next image since it provides more information for accurate and reliable fusing.

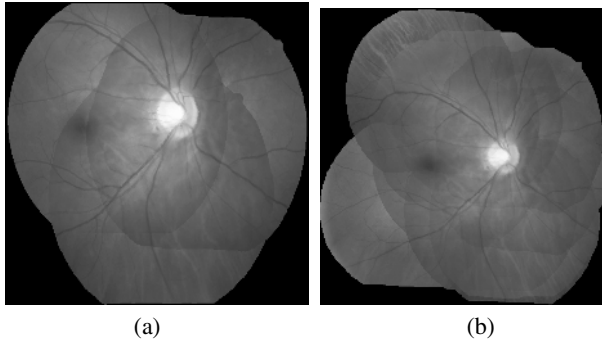


Fig. 2. Fusion maps of (a) three and (b) nine gray level retinal fundus images

4 Experiment Results

The image database that we use to evaluate the performance of the proposed registration algorithm consists of 115 gray level fundus images of both left and right eyes from eleven patients. The image size is $512 \times 512 \times 8$ bits. We randomly pair retinal fundus images captured at different times from the same eye of the same person resulting 268 pairs.

We have 7×7 translation positions and 5 equidistant rotation angles. The step-length for the translation parameters X or Y in vertical or horizontal coordinate directions is one pixel shift. The step-length for the rotation angle is $\Delta\theta = 180 / (\sqrt{2} \times \pi \times 512)$ degrees.

Initial parameter values used for calculation of local move vectors in the elastic matching algorithm are carefully determined. Each line or curve is approximated by fitting a sequence of short straight lines ('elements') of about 20 pixels long in terms of the image size with which the local elastic matching begins (512×512).

Figure 2 shows successful fusion maps consisting of several retinal images. These maps are helpful to provide a broad view of the retina for a better understanding of the anatomical structure of eyes.

5 Conclusion

Fusing fundus retinal images is fundamental to reveal helpful information of the eyes in order to track the progress of possible diseases. For successful fusing, it is crucial to

design an accurate method to register pair of vascular tree features. We employ a ‘coarse-to-fine’ mapping strategy to register pairs of identified vascular trees of retinas. A coarse matching algorithm is able to achieve global alignment but suffering from local ‘ghost vessels’. Therefore, we propose the use of local elastic matching algorithm for fine local alignment. The transformed vectors for pixels in the registered fundus image are conveniently calculated by combining the local move vector and the global model-transformed vector. We build fusion maps by merging gray intensities of pixels with arithmetic averaging. Experiment results show that successful mapping can be achieved within improved accuracy and the fusion maps seem perfect in terms of visual inspection.

References

1. D. E. Singer, D. M. Nathan, H. A. Fogel, A. P. Schachar: Screening for diabetic retinopathy, *Ann. Intern. Med.*, no. 116, (1992) 660-671
2. N. Ritter, R. Owens, J. Cooper, R. H. Eikelboom and P. P. V. Saarloos: Registration of stereo and temporal images of the retina, *IEEE Trans. Med. Imag.*, vol. 18, (1999) 404-418
3. A. Can, C.V. Stewart, B. Roysam and H.L. Tanenbaum: A feature-based, robust, hierarchical algorithm for registering pairs of images of the curved human retina, *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 24, no. 3, March (2002) 347-364
4. A. Pinz, S. Bernogger, P. Datlinger and A. Kruger: Mapping the Human Retina, *IEEE Trans. Med. Imag.*, vol. 17, no. 4, Aug. (1998) 606-619
5. G. K. Matsopoulos, N. A. Mouravliansky, K. K. Delibasis, and K. S. Nikita: Automatic retinal image registration scheme using global optimization techniques, *IEEE Trans. Info. Tech. Biomed.*, vol. 3, no. 1, Mar. (1999) 47 -60
6. B. Fang, W. Hsu, M. L. Lee: Reconstruction of vascular structures in retinal images, in *Proceedings. ICIP'2003*, Barcelona, Spain, September (2003)
7. G. Borgesfors: Hierarchical Chamfer Matching: a parametric edge matching algorithm, *IEEE trans. Pattern Analysis Machine Intelligence*, vol. 10, no. 6, (1988) 849-865
8. C. H. Leung and C. Y. Suen: Matching of Complex Patterns by Energy Minimization, *IEEE Transactions on Systems, Man and Cybernetics*, Part B, vol. 28, no. 5, (1998) 712-720

Improving PSO-Based Multiobjective Optimization Using Competition and Immunity Clonal

Xiaohua Zhang¹, Hongyun Meng², and Licheng Jiao¹

¹ Institute of Intelligent Information Processing, Xidian University, Xian, China
mzhangh@hotmail.com

² Dept.of Applied Math. Xidian University, Xian, China
mhyxdmath@hotmail.com

Abstract. An Intelligent Particle Swarm Optimization (IPSO) for MO problems is proposed based on AER (Agent-Environment-Rules) model, in which Competition and Clonal Selection operator are designed to provide an appropriate selection pressure to propel the swarm population towards the Pareto-optimal front. Simulations and comparison with NSGA-II and MOPSO indicate that IPSO is highly competitive.

1 Introduction

Particle Swarm Optimization (PSO) is a relatively recent heuristic algorithm inspired by the choreography of a bird flock developed by Eberhart and Kennedy [1, 2]. There have been several recent proposals to extend PSO to deal with multi-objective problem [3-5]. In this paper, an Intelligent Particle Swarm Optimization (IPSO) for MO problems is proposed based on AER (Agent-Environment-Rules) model to provide an appropriate selection pressure to propel the swarm population towards the Pareto-optimal front. Simulations and comparison with MOPSO [4] and NSGA-II [6] show the proposed algorithm can achieve a good convergence and diversity of solutions.

A general minimization MO problem including n decision variables, l objective functions and m constraints can be defined as

$$\begin{aligned} \min y = F(x) &= (f_1(x), \dots, f_l(x)) \\ \text{s.t. } g = g(x) &= (g_1(x), \dots, g_m(x)) \leq 0 \end{aligned} \quad (1)$$

$X = \{x \in R^n \mid g_i(x) \leq 0, i = 1, \dots, m\}$ is feasible solution space.

Let $U = (u_1, \dots, u_l)$, $V = (v_1, \dots, v_l)$ be two vectors, then U dominates V denoted as $U \preceq V$ if and only if $u_i \leq v_i (i = 1, \dots, l)$ and $u_i < v_i$ for at least one component. Thus, a solution x of the MO problem is said to be Pareto Optimal if and only if there does not exist another solution y such that $F(y)$ dominates $F(x)$. The set of all Pareto Optimal solutions of MO problem is called Pareto Optimal Set and we denote it with P^* and the set $PF^* = \{F(x) \mid x \in P^*\}$ is called Pareto Front.

2 A Modified Particle Swarm Optimization

It is well-known that the update of particle in the standard PSO is very monotone and invariable. In addition, there exists no sharing of information with other particles in the population. While for multiobjective optimization, sharing of information among the individuals or particles in a population or swarm is crucial in order to introduce the necessary selection pressure to propel the population moving towards the true Pareto-optimal front. A modified model for PSO is given based on the local perception of each particle as following:

$$\begin{cases} v = v + c_1 r_1 (pbest - x) + c_2 r_2 (nbest - x) + c_3 r_3 (gbest - x) \\ x = x + wv \end{cases} \quad (2)$$

In this new model, each particle can not only remember its best position *pbest* of the current particle found so far and the best positions *gbest* of the swarm found so far, but also the local best position *nbest* of the current particle's neighbors found. It needs to be pointed out here that *gbest* is used to speed up convergence and the *nbest* is used to escape the local optima. The Constriction factor *w* depends on the energy of each particle or its distance to the optimal solution *gbest*.

3 Intelligent Particle Swarm Optimization

AER (Agent-Environment-Rules) model is a multiagent system model [7] based on the theory of artificial life. The word "agent" firstly comes from the book of "Society of Mind" edited by Minsky, where an agent is a physical or virtual entity that essentially has the following properties: It is able to live and act in the environment;

It is able to sense its local and global environment; it is driven by certain purposes; it has some reactive behaviors. As can be seen, the meaning of an agent is very comprehensive, and what an agent represents is different for different problem. Agent, Environment and Interactive Rule are three main concepts in it. In recent years, AER model has been widely used in NP-hard problem and achieved good performance.

3.1 New Rules for Agent Particles

Using the adaptability and cooperation relation between particles, in this paper each particle is taken as an agent particle with the ability of memory, communication, response, cooperation and self-learning. We design each agent particle in this paper with following properties. It has its initial position, velocity and energy, here the energy is related to fitness; it has the ability of global sensing and that of local sensing; There exists competition between agent particles, and positions of died agent particles will be inhabited by the particle with higher energy in their neighborhood; the particle has self clone ability, scale depend on the energy of particle.

Here, the agent particle lattice can be represented in two dimension lattice $P(i, j)$, $i=1, \dots, M, j=1, \dots, N$. Each point of intersection $P(i, j)$ represents an agent particle, the neighborhood of $P(i, j)$ can be defined as

$NP(P(i, j)) = \{P(i-m, j-n) | m, n = -d, \dots, 0, \dots, d, |m| + |n| \neq 0\}$ and the radius d shows its local perception range.

3.2 Competition Operator

In our AER model, each particle has a local sensing neighbor with radius of $d = 1$, for each agent particle P , If its energy is not less than the other's, then particle P will survive and produce new agent particle by equation (2); else, it will be abandoned or died forcedly. Here two ways are taken to replace the position.

Assume the agent $P = (x_1, \dots, x_n)$ to be died forcedly and $P_{\max} = (y_1, \dots, y_n)$ be the agent with the maximum energy in its neighborhood. In the first way, new agent $P^{new} = (x_1^{new}, \dots, x_n^{new})$ is given according to the equation $P^{new} = P + rand(0,1) (P_{\max} - P)$. While in the second way, let $range_i = (u_i - l_i) f(t)$, where $[l_i, u_i]$ is the range of design variable of the problem, and t is the number of generation and $f(t)$ satisfies $1 > f(1) > f(2) > \dots$. Let

$$l'_i = \begin{cases} x_i - range_i & \text{if } x_i - range_i > l_i \\ l_i & \text{otherwise} \end{cases}, u'_i = \begin{cases} x_i + range_i & \text{if } x_i + range_i < u_i \\ u_i & \text{otherwise} \end{cases},$$

Then the new agent particle is produced by: $x_i^{new} = l'_i + rand(0,1) (u'_i - l'_i)$.

3.3 Clonal Selection Operator

The famous antibody clonal selection theory [8] was put forward by Burnet in 1958. It establishes the idea that the antigens can selectively react to the antibodies.

Clonal Operating. In this paper, general agent particles and latency agent particles will be cloned. Let T^C be clonal operator and for any particle $A_i(k)$, P_{clone} is the outcome of $A_i(k)$ by T^C as $P_{clone} = I_i \times A_i(k)$, where I_i is q_i dimension row vector with entries as one. Easy to see, P_{clone} is also a q_i -dimension vector with the same component $A_i(k)$, as its element. The scale of P_{clone} is q_i , which is defined as $q_i = g(n_c, F(A_i(k)), p_i)$, in which p_i reflects the affinity with other agent particles, and the higher affinity between particles, the smaller p_i is. Generally speaking, q_i is related to the number of generation. Here

$$q_i = Int(n_c * Energy(A_i(k)) / \sum Energy(A_i(k)) * p_i) \tag{3}$$

where $n_c > MN$ is a constant, p_i is the minimize distance between $A_i(k)$ to other particle in it's neighbor and $Int(x)$ rounds x to the least integer bigger than x .

Clonal Mutation. Crossover and Mutation is the main operator in immune genic operating. It needs to be pointed out here that clonal selection operator in this paper is used only to introduce the mechanism of the immunology, rather than to follow the concepts of the immunology completely. For multiobjective problem, Therefore, Clonal Mutation is adopted in our new model. Let T_M be Clonal Mutation operator, $A_i'(k)$ is produced through T_M by (4), where $Rand(A_i(k))$ denotes an agent particle chosen randomly, p_m is a given threshold and $PSO(A_i(k))$ produced by (2).

$$A_i'(k) = \begin{cases} Rand(A_i(k)) & \eta < p_m \\ PSO(A_i(k)) & otherwise \end{cases} \tag{4}$$

Clonal Selection. T_s^C denotes $B = \{A_{iq_j}''(k) \mid \max energy(A_{ij}'') \quad q_j = 1, 2, \dots, q_i\}$ $i = 1, 2, \dots, n$ the probability of B taking place of $A_i'(k) \in \bar{A}(k)$ is:

$$p_s^k(A_i \rightarrow B) = \begin{cases} 1 & F(B) \preceq F(A_i(k)) \\ \exp\left(-\frac{dist(F(A_i(k)) - F(B))}{a}\right) & \text{else if } A_i(k) \neq A_i^{best} \\ 0 & \text{else } A_i(k) = A_i^{best} \end{cases} \tag{5}$$

Where A_i^{best} is the best in the current population and $a > 0$ is a value concerned with diversity of the population. Generally, the better the diversity is, the bigger a is. Otherwise, a is smaller.

4 Experiments and Discussion

In order to know how competitive our approach was, we decided to compare it against NSGA-II and MOPSO that are representative of the state-of-the-art, where the source codes of NSGA-II and MOPSO are from URL <http://delta.cs.cinvestav.mx/~ccoello/EMOO/EMOOsoftware.html>. The test functions used in this paper are all from [9].

In the proposed algorithm, we take population size as 100 and external population size 100, c_1 and c_2 were set to 2.0. w was decreased linearly from 0.6 to 0.2. The NSGA-II was run using a population size of 100, a crossover rate of 0.8, tournament selection, and a mutation rate of $1/n$. MOPSO used a population of 100 particles, a repository size of 100 particles, a mutation rate of 0.05, and 30 divisions for the adaptive grid. In all the test functions, we report the results obtained from performing 30 independent runs of each algorithm compared. The total number of fitness function evaluations for each function is set to 2500, 2000, 10000 and 7200, respectively.

Figure 1 shows the Pareto Front produced by IPSO, NSGA and MOPSO in 30 independent runs. It is easy to see that the Pareto fronts produced by the proposed algorithm are all below those of others.

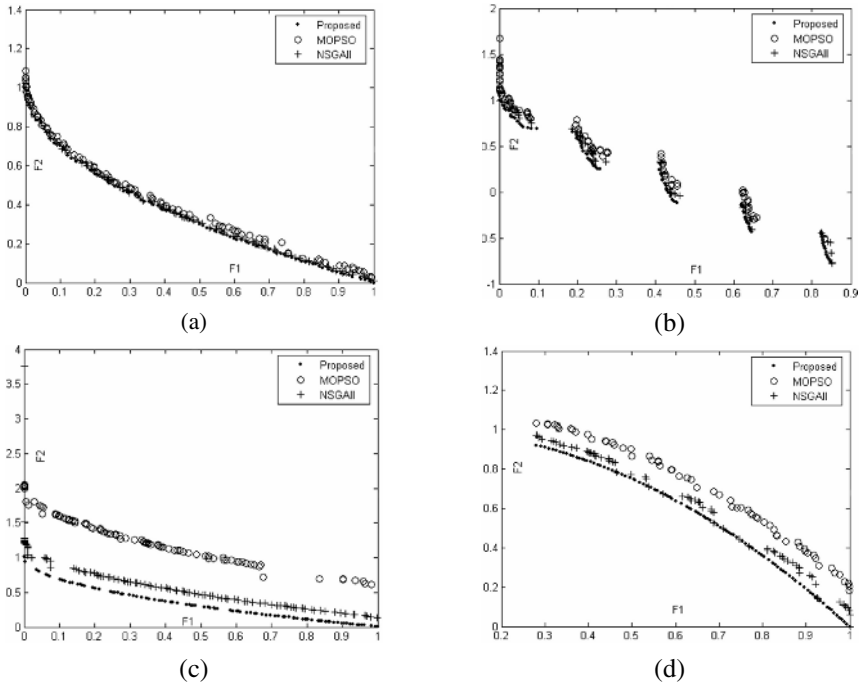


Fig. 1. (a) Pareto fronts for the ZDT1 (b) Pareto fronts for the ZDT3 (c) Pareto fronts for the ZDT4 (d) Pareto fronts for the ZDT6

Table 1 (a). Comparison Results of metrics for the ZDT1

	IPSO			NSGA-II			MOPSO		
	GD	SP	ER	GD	SP	ER	GD	SP	ER
Worst	1.6e-5	0.2922	0.8614	0.0001	0.3056	1.0000	0.0006	0.2684	0.9800
Best	4.0e-5	0.5981	0.6733	0.0033	0.3059	0.9900	0.0040	0.6077	0.9500
Average	2.7e-5	0.4918	0.7875	0.0013	0.3058	0.9937	0.0015	0.4548	0.9640
Median	2.7e-5	0.5653	0.7921	0.0001	0.3059	0.9900	0.0015	0.4943	0.9800
Std. Dev.	5.0e-6	0.1078	0.9546	0.0015	0.0001	0.9951	0.0008	0.1187	0.9953

Table 1 (b). Comparison Results of metrics for the ZDT3

	IPSO			NSGA-II			MOPSO		
	GD	SP	ER	GD	SP	ER	GD	SP	ER
Worst	2.4e-6	0.3394	0.5100	0.0000	0.3271	1.0000	0.0176	0.0858	1.000
Best	3.4e-6	0.5172	0.2800	8.0348	2.7046	0.2000	15.549	3.7311	0.9910
Average	2.8e-6	0.4377	0.3593	0.6331	0.6820	0.9123	1.2488	0.6976	0.9930
Median	2.4e-6	0.5172	0.2800	0.0319	0.4784	1.0000	0.4635	0.5742	0.9940
Std. Dev.	4.4e-7	0.0873	0.9121	1.8238	0.5881	0.8102	2.8277	0.6358	0.9966

Table 1 (c). Comparison Results of metrics for the ZDT4

	IPSO			NSGA-II			MOPSO		
	GD	SP	ER	GD	SP	ER	GD	SP	ER
Worst	0.000	0.2364	0.0000	0.0003	0.2477	1.0000	0.0053	0.2292	1.0000
Best	3.7e-5	0.4927	0.0000	0.0051	0.4627	0.9687	0.0102	0.4828	0.9872
Average	2.6e-5	0.4046	0.0000	0.0021	0.3486	0.9835	0.0076	0.3718	0.9992
Median	2.6e-5	0.4462	0.0000	0.0015	0.3246	0.9868	0.0075	0.3840	1.0000
Std. Dev.	4.5e-6	0.0916	1.0000	0.0016	0.0754	0.9870	0.0011	0.0858	0.9968

Table1 (d). Comparison Results of metrics for the ZDT6

	IPSO			NSGA-II			MOPSO		
	GD	SP	ER	GD	SP	ER	GD	SP	ER
Worst	1.3e-5	0.3966	0.9307	0.0002	0.4401	0.9733	0.0017	0.3107	1.0000
Best	3.8e-5	0.8442	0.7624	0.0151	0.8477	0.9556	0.0470	0.7722	0.9880
Average	2.1e-5	0.6519	0.8399	0.0039	0.6259	0.9636	0.0120	0.5700	0.9993
Median	2.0e-5	0.6258	0.8366	0.0026	0.6072	0.9633	0.0103	0.5261	1.0000
Std. Dev.	6.0e-6	0.1427	0.9508	0.0037	0.1137	0.9961	0.0099	0.1382	0.9971

Table 1 shows the comparison results among the three algorithms considering the metrics GD, SP, ER proposed in [4]. It is easy to see that the average performance of IPSO is the best with respect to GD. To Spacing (SP), IPSO does worse (except for ZDT4) than NSGA-II and MOPSO in the second issue from the list previously provided. Regarding to Error ration(ER), IPSO outperforms the other ones, and it would mean the percentage of solutions generated by IPSO that are members of the true Pareto optimal set is higher than that of NSGA-II and MOPSO in all test functions, To sum up, NSGA-II performs well in Spacing, i.e., a well-extended Pareto front can be found by NSGA-II (except ZDT3), followed by the MOPSO and IPSO. With respect to ER, MOPSO outperforms well in ZDT1 than NSGA-II, while NSGA-II outperforms well in ZDT3, ZDT4 and ZDT6 than MOPSO. In addition, we can also find from the Figure 4 that the performance of the three algorithms are similarity in ZDT1 and ZDT3, while for ZDT4 and ZDT6, IPSO outperforms well than the other two, and in the four test functions, ZDT 4 and ZDT6 are the most difficult to achieve a well distributed nondominated front, since ZDT4 contains 21⁹ local Pareto-optimal sets and has Pareto front with multimodality, while ZDT6 has a non-convex and non-uniform Pareto front. However, IPSO performs well on both of them.

References

1. Eberhart R, Kennedy J.: A new optimizer using particle swarm theory. Proc. 6th Int.Symposium on Micro machine and Human Science. Nagoya(1995) 39-43
2. Kennedy J, Eberhart R.: Particle swarm optimization. Proc. IEEE Int. Conf. On Neural Networks. Perth (1995)1942-1948
3. Coello C.C.,Lechunga M.S.:A proposal for Multiple Objective Particle Swarm Optimization. In Proceedings of the IEEE World Congress on Computational Intelligence, Hawaii, May 12, IEEE Press (2002)

4. Coello C.C., Pulido G.T. and Lechuga M.S.: Handling Multiple Objectives with Particle Swarm Optimization. *IEEE Trans. On Evolutionary Computation*, 8(3), (2004)256-279
5. Ray T., Liew K. M.: A Swarm Metaphor for Multiobjective design Optimization. *Eng. Opt.*, 34(2), (2002)141-153
6. Deb K, Pratap A, Agrawal S, Meyarivan T.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. *IEEE Trans. On Evolutionary Computation*, 6(2), (2002)182-197
7. Liu JM, Jing H, Tang YY.: Multi-Agent oriented constraint satisfaction. *Artificial Intelligence*, (136)1, (2002)101-144
8. Lu D., Ma B.: *Modern Immunology* (in Chinese). Shanghai: Shanghai Scientific and Technological Education Publishing House (1998)
9. Zitzler E.: *Evolutionary Algorithms for Multi-objective Optimization: Methods and Applications*. Ph.D. Thesis, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland (1999)

Clonal Selection Algorithm for Dynamic Multiobjective Optimization

Ronghua Shang, Licheng Jiao, Maoguo Gong, and Bin Lu

Institute of Intelligent Information Processing, P.O. Box 224, Xidian University,
Xi'an 710071, China
shangronghua1980@163.com

Abstract. Based on the clonal selection theory, a new Dynamic Multiobjective Optimization (DMO) algorithm termed as Clonal Selection Algorithm for DMO (CSADMO) is presented. The clonal selection, the nonuniform mutation and the distance method are three main operators in the algorithm. CSADMO is designed for solving continuous DMO and is tested on two test problems. The simulation results show that CSADMO outperforms another Dynamic Evolutionary Multiobjective Optimization (EMO) Algorithm: a Direction-Based Method (DBM) in terms of finding a diverse set of solutions and in converging near the true Pareto-optimal front (POF) in each time step.

1 Introduction

Despite the usefulness of the evolutionary multiobjective optimization (EMO) algorithms, there has been lukewarm interest in extending the ideas to solve dynamic multiobjective optimization (DMO) problems^[1]. Marco Farina et al addressed this issue and suggested a suite of two test problems for the DMO problems and also suggested a baseline algorithm-Dynamic Evolutionary Multiobjective Optimization (EMO) Algorithm: a Direction-Based Method (DBM) for tackling such problems.

The famous antibody clonal selection theory^[2] was put forward by Burnet in 1958. It established the idea that the antigens can selectively react to the antibodies, which are the native production and spread on the cell surface in the form of peptides. So the clonal selection is a dynamic self-adaptive process of the immune system.

Based on the clonal selection theory^[3], a new DMO algorithm-Clonal Selection Algorithm for DMO (CSADMO) is presented. Simulation results of the CSADMO on the two test problems are compared with DBM and much better performance in both the convergence and diversity of obtained solutions of CSADMO is observed.

2 Clonal Selection Algorithm for DMO

In this paper, we propose the algorithm CSADMO which is designed for solving continuous DMO. And decimal coding is used in CSADMO.

2.1 Description of the Algorithm

The Clonal Selection Algorithm for DMO (CSADMO) can be implemented as follow:

Step1: Give the size of antibody population N , the clonal size R , the termination generation $G_{\max}(0)$ and G_{\max} , the initial $t=0$, the maximal time step $T_{\max}=21$. Randomly generate the starting antibody POP, $it:=0$;

Step2: If $t=T_{\max}$, stop. Otherwise, go to Step 3;

Step3: If $t=0$, and if $it:=G_{\max}(0)$, go to Step 10; otherwise go to Step4; If $t>0$, and if $it:=G_{\max}$, go to Step 10; otherwise go to Step4;

Step4: Implement the antibody clonal operation to the antibody population POP and get the new antibody population POP;

Step5: Implement the nonuniform mutation operation to POP and get the new POP;

Step6: Compute the objective values of all the antibodies in the antibody population and get the objective-value matrix $N(t)$;

Step7: Find all of the nondominated solutions in $N(t)$ and delete all of the dominated solutions and the antibodies corresponding to the dominated solutions in POP.

Step8: If the size of $N(t)$ is larger than N , delete the individuals in $N(t)$ and the corresponding individuals in POP with the distance method until the number of the individuals in $N(t)$ is N ; and get the new $N(t)$ and the resulting POP;

Step9: $it:=it+1$; go to Step3;

Step10: $it:=0$; $t:=t+1$; go to Step2;

In order to assure the convergence and the convergent rate, at the time step $t=0$, a randomly distributed starting population POP(0) is generated and the termination generation $G_{\max}(0)=150$; and at the time step $t \neq 0$, the starting population POP(t) is the resulting population POP($t-1$) of the last time step and $G_{\max}=100$.

2.2 Three Main Operators of CSADMO

The clonal selection operator, the nonuniform mutation^[4] and the distance method^[5] are three main operators of CSADMO, which assure the diverse set of solutions and the convergence to the Pareto-optimal front (POF) in each time step.

2.2.1 Clonal Selection

The clonal selection operator is to implement three steps on the antibody population^[6].

Clone $T_c^C(*)$: The clonal operator $T_c^C(*)$ is defined as:

$$Y(k) = T_c^C(A(k)) = [T_c^C(a_1(k)) \quad T_c^C(a_2(k)), \quad \dots, \quad T_c^C(a_n(k))]^T, \quad (1)$$

where $Y(k) = T_c^C(a_i(k)) = I_i \times a_i(k)$, $i=1,2,\dots,n$ and I_i is an n_c dimensional row vector. Here, n_c is a given integer called clonal scale and n_c copies of each antibody will be produced by cloning. After the clonal operator, the antibody population is:

$$Y(k) = \{y_1(k), y_2(k), \dots, y_n(k)\}. \quad (2)$$

Clonal Mutation $T_m^C (*)$: The clonal mutation is not applied to the $a_i \in y_i$ in order to save the information of original population, namely,

$$P(T_m^C(y_i = y'_i)) = \begin{cases} P_m > 0 & y_i \in y_{ij} \\ 0 & y_i \in a_i \end{cases} \quad i = 1, 2, \dots, n, j = 1, 2, \dots, n_c \quad (3)$$

P_m is the mutation probability, after clonal mutation, the antibody population is:

$$Y'(k) = \{y'_1(k), y'_2(k), \dots, y'_n(k)\} = \{a_i(k), y'_{i1}(k), y'_{i2}(k), \dots, y'_{in_c}(k)\} \quad (4)$$

Clonal Selection $T_s^C (*)$: $\forall i = 1, 2, \dots, n$, if

$$b = \{y'_{im} \mid \min f(y'_{ij}), j = 1, 2, \dots, n_c, m \in j\} \text{ and } f(a_i) < f(b), a_i \in y'_n \quad (5)$$

So the information exchanging among the antibody population is realized.

2.2.2 Nonuniform Mutation

In the traditional genetic algorithms, the mutation operator has nothing to do with the evolutionary generations. Z. Michalewicz combined the mutation operator with the evolutionary generation. At the early age of the evolutionary process, mutation operation has a comparatively larger range; but as the evolutionary progresses, it becomes smaller and smaller which has a fine-tuning effect on the evolutionary system.

Let $s = (v_1, v_2, \dots, v_n)$ is a parent solution, and $v_k \in [a_k, b_k]$ is selected to mutate. The resulting solution after mutation is $s' = (v_1, \dots, v_{k-1}, v'_k, \dots, v_n)$.

$$v'_k = \begin{cases} v_k + \Delta(t, b_k - v_k), & \text{if } \text{rnd}(2) = 0 \\ v_k - \Delta(t, b_k - v_k), & \text{if } \text{rnd}(2) = 1 \end{cases} \quad (6)$$

where $\text{rnd}(2)$ is the result of managing a stochastic positive integer m with module 2 and t represents the current generation. The range of function $\Delta(t, y)$ is $[0, y]$ and when t becomes larger, the probability of $\Delta(t, y)$ approaching 0 increases.

$$\Delta(t, y) = y \lfloor (1 - r^{(1-t/T)^\lambda}) \rfloor \quad (7)$$

where $r \in [0, 1]$ is a random number, T is the maximum generation and λ determines the degree of the nonuniformity and tunes the region of search.

2.2.3 The Distance Method

The distance method is used in this paper so as to assure the diversity of the solutions and the crowding-distance-assignment which was proposed by Kalyanmoy Deb et al^[7] is adopted (see [7] for the details of the crowding-distance-assignment).

3 Test Problems and Simulation Results

The test problems are proposed by Marco Farina et al and in order to measure the performance, we use the time-dependent convergence measure $e_f(t)$ ^[1]. CSADMO is compared with DBM. In order to compare the two algorithms impartially, we adopt the simulation results of DBM directly. In the objective space, the known POFs are shown with dots or lines both in CSADMO and in DBM and the obtained time-dependent solutions are shown with stars in CSADMO and with circles in DBM.

Test 1

$$\left\{ \begin{array}{l} f_1(\mathbf{x}) = \sum_{x_i \in \mathbf{x}_I} x_i^{F(t)} \\ f_2(\mathbf{x}) = g(\mathbf{x}) \left(1 - \sqrt{\frac{f_1(\mathbf{x})}{g(\mathbf{x})}} \right) \\ g(\mathbf{x}) = 1 + G(x) + \sum_{x_i \in \mathbf{x}_{II}} (x_i - G(t))^2, G(t) = |\sin(0.5\pi t)|, F(t) = 10^{2\sin(0.5\pi t)} \\ t = \frac{1}{n_\tau} \left\lfloor \frac{\tau}{\tau_T} \right\rfloor, \mathbf{x}_I \in [0, 1], \mathbf{x}_{II} \in [-1, 1] \end{array} \right. \quad (8)$$

In this test problem, we recommend $|\mathbf{x}_I|=5$ and $|\mathbf{x}_{II}|=15$. The density of solutions on the POF varies with t , so a good DMO algorithm is to find a widely distributed set of solutions every time there is a change in t . Theoretically, the POF is $f_2 = (1 + G(t)) \times (1 - \sqrt{f_1})$ every time there is a change in t .

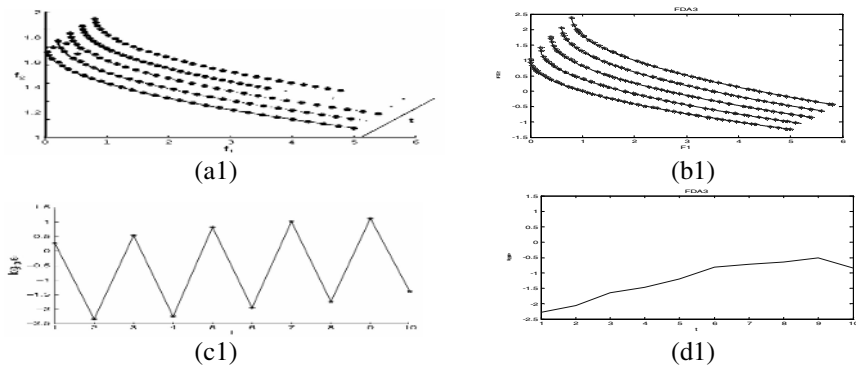


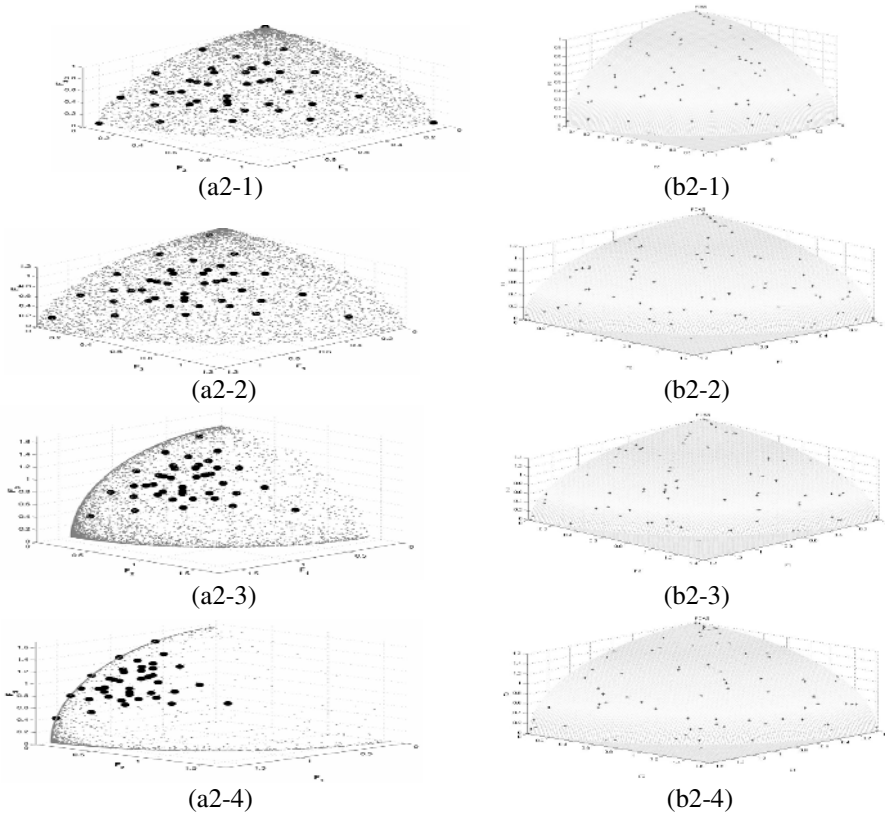
Fig. 1. The simulation results of test 1. (a1) Simulation results of DBM. (b1) Simulation results of CSADMO. (c1) Log of the convergence measure versus time with DBM. (d1) Log of the convergence measure versus time with CSADMO.

(b1) shows that CSADMO can find a widely distributed set of solutions in every time step. However, at the fourth time step, the distribution of the solutions of DBM is poor, as depicted in (a1). Both the algorithms are able to obtain reasonably good convergence after each time change, as depicted in (c1) and (d1).

Test 2

$$\begin{cases}
 \min_{\mathbf{x}} f_1(\mathbf{x}) = (1 + g(\mathbf{x}_{\text{II}})) \prod_{i=1}^{M-1} \cos\left(\frac{y_i \pi}{2}\right) \\
 \min_{\mathbf{x}} f_k(\mathbf{x}) = (1 + g(\mathbf{x}_{\text{II}})) \left(\prod_{i=1}^{M-k} \cos\left(\frac{y_i \pi}{2}\right) \right) \sin\left(\frac{y_1 \pi}{2}\right), k = 2 : M-1 \\
 \min_{\mathbf{x}} f_M(\mathbf{x}) = (1 + g(\mathbf{x}_{\text{II}})) \sin\left(\frac{y_1 \pi}{2}\right) \\
 g(\mathbf{x}_{\text{II}}) = |\sin(0.5\pi t)| + \sum_{x_i \in \mathbf{x}_{\text{II}}} (x_i - |\sin(0.5\pi t)|)^2, y_i = x_i^{(1+100\sin^4(0.5\pi t))} \text{ for } i = 1 : M-1 \\
 t = \frac{1}{n_\tau} \left\lfloor \frac{\tau}{\tau_T} \right\rfloor, \mathbf{x}_{\text{II}} = (x_M, \dots, x_n), x_i \in [0, 1] \quad i = 1 : n
 \end{cases} \tag{9}$$

We recommend $M = 3$ and $n = M + 9$, thereby keeping $|\mathbf{x}_{\text{II}}| = 10$. The density of solutions on the POF changes with t . So the task of a DMO algorithm would be found a good distribution every time there is a change in the density of solutions. The POF is $\sum_{i=1}^M (f_i)^2 = (1 + G(t))^2$ every time there is a change in t .



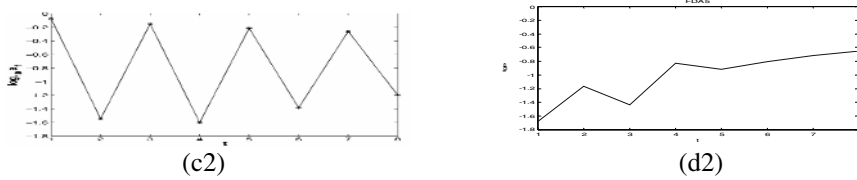


Fig. 2. The simulation results of test 2. (a2-1)~(a2-4) Simulation results of DBM. (b2-1)~(b2-4) Simulation results of CSADMO. (c2) Log of the convergence measure versus time with DBM. (d2) Log of the convergence measure versus time with CSADMO.

DBM has some difficulties in maintaining diversity as depicted in (a2-1)~(a2-4). However, the CSADMO can find a good distribution every time there is a change in the density of solutions as depicted in (b2-1)~(b2-4). Although (d2) shows when t becomes larger, the convergence of CSADMO becomes poorer, both the algorithms are able to obtain good convergence after each time change.

4 Conclusion

In this paper, CSADMO has been proposed and it is tested on two test problems. As has been shown in this study, the proposed algorithm provides a good performance in terms of convergence and diversity and outperforms DBM. CSADMO has the capabilities of accelerating the algorithm's convergent speed and maintaining the diversity of antibody population. But how to extend the algorithm to other DMO problems and how to evaluate a DMO algorithm more comprehensively are our future work.

References

1. Farina, M., Deb, K., Amato, P.: Dynamic Multiobjective Optimization Problems: Test Cases, Approximations, and Applications. *IEEE Trans. Evolutionary Computation.* 8 (2004) 425–442
2. Abbas, A. K., Lichtman, A. H., Pober, J. S.: *Cellular and Molecular Immunology.* 3rd eds. W. B. Saunders Company, New York (1998)
3. Du, H.F., Jiao, L.C., Wang, S.A.: Clonal Operator and Antibody Clone Algorithms. In: *Proceedings of the First International Conference on Machine Learning and Cybernetics, Beijing,* (2002) 506–510
4. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs.* 3rd eds. Springer-Verlag, Berlin Heidelberg New York (1996)
5. Osyczka, A. and Kundu, S.: A Modified Distance Method for Multicriterion Optimization Using Genetic Algorithm, *Computers and Industrial Engineering.* 30(1996) 871-882
6. Jiao, L.C., Du, H.F.: Development and Prospect of the Artificial Immune System. *Acta Electronica Sinica.* 31 (2003) 73–80
7. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Trans. Evolutionary Computation.* 6(2002) 182–197

Key Frame Extraction Based on Evolutionary Artificial Immune Network

Fang Liu and Xiaoying Pan

School of Computer Science and Engineering, Xidian University, Xi'an 710071, China
f631iu@163.com

Abstract. Key frame extraction has been recognized as one of the important research issues in video retrieval. Key Frame Extraction based on Evolutionary Artificial Immune Network (KFE-EAIN) is proposed in this paper. To describe the distribution of video frame data, an artificial immune network is first evolved by video frame data. Then, key frame can be selected by minimal spanning tree of the network. KFE-EAIN does not require the number of clusters to be known beforehand. Otherwise, it can apply to both single shot and video sequence. Experimental results show that KFE-EAIN can effectively summarize content of a video with acceptable complexity.

1 Introduction

In recent years, the use of multimedia information has been increasing rapidly. For all the media types, video is the most challenging one, as it combines all the other media information into a single data stream. However, efficient access to a video is not an easy task due to video's length and unstructured format. Shot boundary detection and key frame extraction are two bases for abstraction and summarization techniques.

Because of its importance, many research efforts have been given in key frame extraction and much progress has been made in this area^[1-3]. In this paper, a new key frame extraction algorithm based on evolutionary artificial immune network (KFE-EAIN) is proposed. KFE-EAIN is based on cluster-validity analysis and is designed to work without any human supervision. And it is a common method for key frame extraction, which can apply not only to a shot but also to the sequence directly.

2 Key Frame Selection Based on Evolutionary Artificial Immune Network

Ideally, key frames should capture the semantics of a shot. However, at current stage, the Computer Vision techniques are not advanced enough to automatically generate such key frames. Instead, we have to base key frame selection on low level visual features, such as color, texture of the salient object in a shot. It is obvious that if a frame is important, the camera will focus more on this frame. This is the basic assumption. We have used in our clustering based key frame extraction technique. In this paper, we only select the color layout descriptor as the character of image frames.

2.1 Color Character

Color character is important information describes video content. A color layout descriptor was proposed in the international criterion MPEG-7. The extraction of it consists of four stages: image partitioning, dominant color selection, disperse cosine transform and non-linear quantization of the zigzag-scanned DCT coefficients.

Firstly, a picture is partitioned into 64 blocks. The size of each block is $W/8 \times H/8$, where W and H denote the width and height of this picture respectively. Secondly, a single dominant color is selected in each block to build a tiny image whose size is 8×8 . We have used simple average colors as dominant colors in the following sections. Thirdly, each of the three components (Y, Cb and Cr) is transformed by 8×8 DCT, and we obtain three sets of DCT coefficients. A few low frequency coefficients are extracted using zigzag scanning and quantized to form the Color layout descriptor.

2.2 Evolutionary Artificial Immune Network Based Approach

Given a shot $X = \{x_1, x_2, \dots, x_n\}$, where $x_i = (x_{i1}, x_{i2}, \dots, x_{i12})^T$ denotes a frame by color layout descriptor. Key frame extraction can be transformed into images clustering.

In the evolutionary immune network theory, the existing cells compete for antigenic recognition and successfully lead to network activation, cell proliferation and clonal selection, while these failed will be eliminated. In addition, antibody - antibody recognition will result in network suppression. In KFE-EAIN, suppression is performed by threshold σ_s . Some definitions have been given firstly:

Definition 2: affinity f_{ij} between antigen x_i and antibody p_j is defined as (1).

$$f(x_i, p_j) = \frac{1}{1 + (x_i - p_j)^T (x_i - p_j)} \quad \begin{matrix} i = 1, 2, \dots, n \\ j = 1, 2, \dots, S \end{matrix} \quad (1)$$

Definition 3: clonal operator T_c^c : If the antibody that clonal operation needed is described as (r_1, r_2, \dots, r_k) . We have formula as equation (2):

$$T_c^c(r_j) = r_j \Theta I_j \quad (j = 1, 2, \dots, k) \quad q_j = \text{Int} \left[Nc * \frac{f(x_i, r_j)}{\sum_{s=1}^k f(x_i, r_s)} \right], \quad (2)$$

Where I_j is a q_j -dimensional row identity vector, $\text{Int}[W]$ is a function returns the minimum integer larger than W , Nc is clone size generated by each stimulated cell.

The network outputs can be taken to be the matrix of memory cells' coordinates (M) and the matrix of inter-cell affinities (S). M represents the network internal images of groups of antigens and S is responsible for determining which cells are connected to each other, describing the general network structure. To achieve problem specific network structure, we will analyze the minimal spanning tree for resulting net.

KFE-EAIN works as follows:

1. $l = 1$, initialize network $A(l)$. Suppression threshold σ_s and natural death threshold σ_d were initialized.
2. For each image x_i (antigen), do: ($i = 1$)
 - 2.1 Determine its affinity $f(x_i, p_j)$ to all the network cells.
 - 2.2 Select the k highest affinity network cells. Then reproduce (clone) these cells. The higher the cell affinity, the larger Nc ;
 - 2.3 Immune gene operations;
 - 2.4 Re-determine the affinity of x_i to all the improved clone cells. Select $q\%$ cells with the highest affinity cells and create a partial M_p memory cell matrix;
 - 2.5 Eliminate those cells whose affinity is inferior to threshold σ_d , yielding a reduction in the size of the M_p matrix;
 - 2.6 Calculate the network cells affinity s_{ij} in M_p matrix, where $s_{ij} = \|p_i - p_j\|$. Eliminate $s_{ij} < \sigma_s$ (clonal suppression) and concatenate $A(l)$ and M_p , ($A(l) \leftarrow [A(l); M_p]$). At the same time, Re-calculate the number of element in $A(l)$;
 - 2.7 $i = i + 1$, if $i \leq n$ then go to 2.1, else go to 3;
3. In $A(l)$, determine s_{ij} and eliminate those cells whose $s_{ij} < \sigma_s$;
4. Add $r\%$ individuals randomly to $A(l)$ and get next initialized network $A(l + 1)$;
5. $l = l + 1$, if $l > 10$ then stop the iterative process and go to 6, else go to 2;
6. Analyze the network by minimal spanning tree (MST). For each key cluster, the frame that is closest to the cluster centroid is selected as the key frame, which captures the salient visual content of the key cluster and thus that of the underlying shot.

In step 2.3, the affinity of the cells with the given antigen i can be improved by the following expression (directed mutation):

$$C = C - \alpha(C - X). \quad (3)$$

Where C is the matrix of network cells, X is the matrix of antigens and α is the learning rate or mutation rate. Equation (3) proposes a biased search, where the antigen - antibody complementarity is increased proportionally to α . By doing so, we guide our search to locally optimize the network cells (greedy search) in order to improve their antigenic recognition capability along the iterations.

3 Experimental Results

To examine the performance of KFE-EAIN on a single video shot, some video sequences including News, Cartoon and Game are tested. Every experiment will repeat 10 times. If a frame is figured as key frame for 6 times or more, it will be key frame. Otherwise, the result is very sensitive to the value of σ_s in KFE-EAIN and user can

control key frames' number by adjusting it. Table 1 shows some experimental results. All shots are chosen randomly. We list 12 or 3 shots in each example.

Table 1. Examples from different test sequence

Sequence	Shot ID (frame)	K-frames ID		
		$\sigma_s=0.25$	$\sigma_s=0.30$	$\sigma_s=0.35$
News	1 (0-34)	4 27	27	27
	2 (35-86)	45	45	45
	3 (87-201)	147 186	147 175	147
	4 (202-256)	219 243	219 239	243
	5 (257-299)	278	286	286
	6 (300-493)	364 469	364	364
	7 (494-714)	507 593 647 701	507 652 701	507 652
	8 (715-796)	715 766	766	766
	9 (797-834)	817 834	820 834	820
	10 (835-1109)	873 894 965 1098	873 965	873 1077
	11 (1110-1200)	1110 1155 1168	1110 1155	1155
	12 (1201-1392)	1201 1293	1293	1293
Cartoon	2 (106-214)	108 113 168 201	127 168 201	176 201
	5 (487-538)	493 527	493 527	527
	8 (837-968)	877 894 933	877 898 933	877
Game	1 (0-126)	64 90 124	16 64 124	64 124
	2 (127-208)	137 189	137 186	137
	7 (724-796)	726 737 775	737 792	737 787

As shown in table 1, the smaller σ_s is, the less compressive of the network and more the number of key frames are. In general, one or two key frames are enough to a single shot. For example, we have $\sigma_s = 0.35$ and some examples are illustrated in Fig. 1. Fig. 1 (1) and (2) show the single key frame from News and Cartoon respectively, while Fig. 1 (3) and (4) show the two key frames from shot-1 of Game.



Fig. 1. Examples for key frame extraction by KFE-EAIN

Thus we know that KFE-EAIN can effectively capture the visual contents of the video shots. And the number of key frames represents the complexity of the shot.

KFE-EAIN can also be applied to the whole video section because it can automatically classify the content of frame, and select the most characteristic frame as the key frame for each classification. In addition, the general cluster algorithm is not strong

enough for a large number of high dimensional data, but KFE-EAIN can obtain good result for a long video section.

In order to verify the performance of KFE-EAIN for a sequence containing several shots, we selected a News video involving 12 shots in Table 1 as a test sequence. The contents of this video mainly express three aspects, newscast, volleyball game and golf game. The results are shown in Fig. 2.



(a) Key frames extracted from every single shot by KFE-EAIN



(b) Key frames extracted from whole sequence directly by KFE-EAIN



(c) Key frames extracted from same sequence by the algorithm in [4]

Fig. 2. Comparing results

As shown in Fig. 2 (a), these key frames can effectively summarize the content of this sequence. The former five frames express volleyball game, the following one frame newscast, the next seven frames golf game and the last frame newscast. There are totally fourteen key frames in it. In the results, the content of volleyball game is expressed well by frame 147. Frame 507 or frame 1077 is about golf game. Frame 364 or frame 1293 is about newscast clearly, but there is redundancy in Fig. 2 (a).

Fig. 2 (b) shows the results of applying KFE-EAIN to the whole sequence directly, which hold the content of this video very well. The first three frames express the content of volleyball game, the following six frames golf game, and the last frame newscast. There are totally ten key frames in (b). Though the number of key frames in (b) is less than that of in (a), these key frames in (b) still clearly summarize the content of this sequence. In Fig. 2 (b), the content of volleyball game is expressed well by frame 146. Frame 507 of six frames about golf game expresses well the content of golf game. Last frame 1307 reflects the newscast with no redundancy. In addition, the results not only reserve the content of the video well, but also reduce the redundancy

compared to the extraction from every single shot by KFE-EAIN. In theory, a video section made up of several shots contains similar frames. And these similar frames can be represented with the same key frame. So the results that we got in KFE-EAIN present the content of video section, but not the content of each shot.

Fig. 2 (c) shows key frames extracted from the same video by algorithm in [4]. Though these key frames can represent the content of the video approximately, the precision is not good. For example, the content of volleyball game is only depended on frame 256 and frame 298 in Fig. 2 (c). But we only know that there are two sportsmen, and not know what game they are playing. Furthermore, the redundancy of (c) is higher than (b), frame 301, 493 and 1377 express the same content, the newscast shot.

4 Conclusions

Key frame extraction based on evolutionary artificial immune network (KFE-EAIN) is proposed in this paper. KFE-EAIN first describes the character of every frame by color layout description and maps these data to the cells in an artificial evolutionary immune network. The artificial immune network is evolved by these frame data. The network structure gotten reflects the distribution of video frame data. At last, minimal spanning tree (MST) is employed to extract the key frames. The experimental results show that KFE-EAIN has a good performance to represent the contents of video. It makes the video condense storage and retrieval much effective. And also, KFE-EAIN is not limited on the color character. We can combine several video characters, such as texture, motion, etc. The extraction of combined characters will be more efficient.

Acknowledgements. This work is supported by the National Natural Science Foundation of China under Grant Nos. 60372045, 60133010 and the National Grand Fundamental Research 973 Program of China under Grant No. 2001CB309403.

References

1. Calic, J., Izuierdo, E.: Efficient key-frame extraction and video analysis. Proceedings of the International Conference on Information Technology. (2002) 28-33
2. Liu, T. M., Zhang, H. J.: A novel video key frame extraction algorithm. IEEE International Symposium on Circuits and Systems. Vol.4. (2002) 149-152
3. Lee, S., Haves, M.H.: A fast clustering algorithm for video abstraction. Proceedings of the International Conference on Image Processing. Vol.2. (2003) 563-566
4. Yueting Zhuang, Yong Rui.: Adaptive key frame extraction using unsupervised clustering. Proceedings of the International Conference on Image Processing. Vol.1. (1998) 866-870
5. Jerne, N. K.: Towards a Network Theory of the Immune System. Ann. Immunol. (Inst.Pasteur) 125C. (1974) 373-389.
6. Nunes de Casto,L., Von Zuben,F.J.: An evolutionary immune network for data clustering. Proceedings of the Sixth Brazilian Symposium on Neural Networks. (2000) 22-25

Clonal Selection Algorithm with Immunologic Regulation for Function Optimization

Hang Yu, Maoguo Gong, Licheng Jiao, and Bin Zhang

Institute of Intelligent Information Processing, P.O. Box 224, Xidian University,
Xi'an 710071, China
Yuhang9551@163.com, maoguo_gong@hotmail.com

Abstract. Based on the Antibody Clonal Selection Theory of immunology, four immunologic regulation operators inspired by immune regulation mechanism of biology immune system are presented in this paper, and a corresponding algorithm, Immunologic Regulation Clonal Selection Algorithm (IRCSA), is put forward. The essential of immunologic regulation operators is to make fine adjustment among the candidates of the algorithm so as to make interrelations between antibodies more complicated and improve the stability, robustness and accuracy of the algorithm. Numeric experiments of function optimization indicate that the new algorithm is effective and useful.

1 Introduction

Artificial Immune Systems (AIS) is a research hot spot of the artificial intelligence after the neural network, fuzzy logic and evolutionary computation, and has been applied to complex problem domains [1], [2]. Immunologic Regulation (IR) is an important, intricate and complex regulative mechanism in biology immune system [4]. However, the relative research is still rare. In this paper, four immunologic regulation operators (IR-operators) are presented in the next section, and a corresponding algorithm, Immunologic Regulation Clonal Selection Algorithm (IRCSA), is put forward. Section 3 provides simulations of function optimization among IRCSA, Genetic Algorithm with elitism preserving strategy (EGA) [5] and Simple Clonal Selection Algorithm (SCSA) [6]. Finally, Section 4 states some conclusions.

2 Algorithms

Artificial Immunologic Regulation (AIR) is a self-learning technique through which the algorithm can accurately and rapidly hunt the global optimum by the interpromotion and interrestriction of the optimization solutions. This kind of regulation mechanism is very complicated, since it not only has positive regulation, but also has negative regulation. Other than the naive appraisal criterion of antigen-antibody affinity, the essential of artificial immunologic regulation is to make fine adjustment among the candidates of the algorithm in order to make interrelations among antibodies more complicated and respond more accurately and quickly. This paper puts forward four novel regulation operators as below.

2.1 Immunologic Regulation Operators (IR-Operators)

2.1.1 Genetic Immunity Operator (IG-Operator)

The main mechanism of genetic immunity is to inject the carriers with antigen codes into the body’s cells where the carriers will be synthesized as corresponding antibody so as to induce immune respond and project the body.

The idea of genetic immunity is mainly realized through the genetic immunity operator (IG-operator) composed of vaccine abstraction and vaccine inoculation. To be exact, they are implemented as tree steps. Firstly, obtain $n_{vaccine}$ best antibodies

$\mathbf{A}_{vaccine}(k) = \{A_{vaccine(1)}(k), A_{vaccine(2)}(k), \dots, A_{vaccine(n_{vaccine})}(k)\}$ as vaccine developing antibodies. Secondly, operate the vaccine developing antibodies to abstract vaccine.

$$V(k) = Y_v^{ad}(\mathbf{A}_{vaccine}(k)) = \bigcap_{i=1}^{n_{vaccine}} A_{vaccine(i)}(k) \tag{1}$$

Here $V(k)$ is namely the vaccine whose essential is schemata [5].

Thirdly, perform the vaccine inoculation that utilizes vaccines to intervene aptly in the variation of genes in individual chromosome so that the modified antibody coincides with $V(k)$.

Given an antibody $A_j(k) (j = 1, 2, \dots, n)$, the vaccine inoculation is

$$\begin{cases} A'_j(k) = Y_s^{ad}(A_j(k)) = A_j(k) \times \bar{V}'(k) + V(k) \\ V_i'(k) = \begin{cases} 1 & V_i \neq 0 \\ 0 & V_i = 0 \end{cases}, \quad (i = 1, 2, \dots, n) \\ \bar{V}'(k) = |1 - V'(k)| \end{cases} \tag{2}$$

Where, $A'_j(k)$ is the wanted antibody after IG-operator.

The essential of IG-operator is to strengthen good schema all along in antibodies and make continuous increment of schema order and defining length so as to gain higher average affinity step by step.

2.1.2 Apoptosis Operator (Apo-Operator)

Apoptosis is important for the immune respond and immune control. Apoptosis is different from death of cells, because it’s an active process. Apoptosis and hyperplasia are all indispensable to life.

According to the mechanism above, this paper designs apoptosis operator (Apo-operator). While operating, firstly, delete the antibodies that have the lowest affinity:

$$\begin{cases} \mathbf{A}'(k) = \mathbf{A}(k) - \mathbf{A}_{bad}(k) \\ \mathbf{A}_{bad}(k) = \{A_{bad_1}(k), A_{bad_2}(k), \dots, A_{bad_{n_m}}(k)\} \end{cases} \tag{3}$$

Now the size of the population $\mathbf{A}'(k)$ is only $(n - n_m)$. In order to maintain the antibody population scale and improve its diversity, we create n_m antibodies from the

remaining antibodies at the second step. In detail, we perform inversing operation to the antibodies based on the probability P_{m2} , and obtain the wanted population.

Apo-operator actively eliminates the antibodies that have fewer or even no matching bits with antigen instead of new antibodies with higher affinity so as to maintain the high efficacies and harmonies of the algorithm. The apoptosis operator has good performance to speed up the convergence.

2.1.3 Affinity Operator (Aff-Operator)

When antigen breaks in the body, the antiserum is composed of the polyclonal antibodies with different affinity to the antigen. As the time goes by, there will be only individuals with high affinity living. That’s because of the affinity regulation.

It’s the Aff-operator that simulates this mechanism. To be exact, it’s mainly realized through the control of reproduction. What is the most different from SCSA is that, the clone scale of the IRCSA is not a constant but self-adjusting based on affinity.

Define

$$\mathbf{D}(k) = \mathbf{Y}_c^{\text{ad}}(\mathbf{A}(k)) = [\mathbf{Y}_c^{\text{ad}}(A_1(k)), \mathbf{Y}_c^{\text{ad}}(A_2(k)), \dots, \mathbf{Y}_c^{\text{ad}}(A_n(k))]^T \tag{4}$$

where, $\mathbf{Y}_c^{\text{ad}}(B_i(k)) = \mathbf{I}_i \times B_i(k)$, $i = 1, 2, \dots, n$, \mathbf{I}_i is q_i dimension row vector and

$$q_i(k) = \text{Int} \left(n_c \times \frac{f(A_i(k))}{\sum_{j=1}^n f(A_j(k))} \right), \quad i = 1, 2, \dots, n \tag{5}$$

n_c is a given value related to the clone scale; $\text{Int}(\bullet)$ rounds the element to the nearest integers towards minus infinity.

Aff-regulator strengthens the local search of the SCSA, and make up the flaws of the SCSA when optimizes the function whose global optimum is in the centre of large-scale flat areas.

2.1.4 Idiotpe and Anti-idiotpe Operator (IAdio-Operator)

It is a very important role that idiotpe and anti-idiotpe regulation plays during immune respond. In immune system, the antibodies, B cell and T cell, have duality. That is, they can not only identify and combine with antigen, but also has the antigenicity that can induce the immune respond. Therefore, idiotpe is antibody, but can perform as antigen which can stimulate antibody.

In order to simulate this mechanism of immunology, this paper designs IAdio-operator. Without loss of generality, consider antibody $A_i(k) = \{a_{i_1}, a_{i_2}, \dots, a_{i_p}, \dots, a_{i_q}, \dots, a_{i_n}\}$ ($i = 1, 2, \dots, n$) which is just obtained from repro-duction. The detail of the IAdio-operator is shown as follows:

1. Select n_{opt} most excellent antibodies based on affinity:

$$\mathbf{B}(k) = \{B_1(k), B_2(k), \dots, B_{n_{opt}}(k)\};$$

2. Choose one antibody $B_i(k) = \{b_{i_1}, b_{i_2}, \dots, b_{i_l}\}$ ($i = 1, 2, \dots, n$) randomly from $\mathbf{B}(k)$, then cross it with $A_i(k)$ based on one-point crossover strategy, and then $A'_i(k)$ and $B'_i(k)$ is obtained;
3. Calculate the distance d_A and d_B between $A'_i(k)$ and $B'_i(k)$ with parental antibody $B_i(k)$ respectively. Here we adopt hamming distance, that is:

$$d_R(k) = \sum_{m=1}^l |r_j(k) - b_m(k)| \tag{6}$$

Where l is the length of the antibody bit string, and $r_j(k)$ stands for the j_{th} gene of the new antibody $A'_i(k)$ or $B'_i(k)$. Now we select the antibody corresponding with the minimum distance between d_A and d_B as next generation.

IAdio-operator is complementary and amendatory to clonal operation, since the searching area is enlarged by joining in the crossover operator. IAdio-operator makes algorithm search with directions and decreases the evaluation number effectively.

2.2 Immunologic Regulation Clonal Selection Algorithm

We introduce these operators above into the Clonal Selection Algorithm [6], and present IRCSA, whose simplification is realized as Figure 1:

-
- Step1** Initialization: Set the parameters, generate n antibodies randomly; $k = 0$;
 - Step2** Condition Judgment: If satisfying the halt condition, then stop the algorithm; otherwise go to step 3;
 - Step3** IR-operation: Perform IG-operator, Apo-operator, Aff-operator and IAdio-operator one by one;
 - Step4** Immune genic operation: Implement mutation by bit;
 - Step5** Clonal selection operation: Obtain the new antibody population; $k = k + 1$;
Turn to step 2.
-

Fig. 1. Immunologic regulation clonal selection algorithm

IRCSA presented in this paper emphasizes on the balance of the convergent speed and population diversity, and can accelerate the convergent speed and improve the accuracy effectively. This will get the confirmation in the section 3.

3 Validation of the Effectiveness

In order to validate the algorithm, we compare the algorithm with another two algorithms. They are Genetic Algorithm with elitism preserving strategy (EGA) [5] and Simple Clonal Selection Algorithm (SCSA) [6]. The six standard functions used

in the test are detailedly showed in reference [3]. The statistic results of T_{best} (the best reaching value), T_{worst} (the worst reaching value), $T_{average}$ (the average reaching value), T_{median} (the median reaching value), T_{av} (the evaluation number of reaching the optimal value) and T_{sd} (the standard deviation) obtained from 30 independent runs are shown in table 1.

Table 1. The performance comparison of EGA, SCSA and IRCSA

T_{sd} (T_{av})			T_{best} (T_{median}) [$T_{average}$] { T_{worst} }		
EGA	SCSA	IRCSA	EGA	SCSA	IRCSA
4.78 $\times 10^{-4}$ (28467)	2.63 $\times 10^{-7}$ (40426)	2.06 $\times 10^{-9}$ (15373)	2.11874779197841 (2.11823023514338) [2.1181561049925] {2.11682951790695}	2.11876342054653 (2.11876338531038) [2.1187632934359] {2.11876229601257}	2.11876342054653 (2.11876342054653) [2.11876342054653] {2.11875775002901}
1.96 $\times 10^2$ (26567)	1.39 $\times 10^{-12}$ (24786)	1.39 $\times 10^{-12}$ (13032)	3597.78375197009 (3445.54657642919) [3396.84368562044] {2892.3412523785}	3599.99999313353 (3599.99999313353) [3599.99999313353] {3599.99999313353}	3599.99999313353 (3599.99999313353) [3599.99999313353] {3599.99999313353}
6.81 $\times 10^{-2}$ (24067)	1.45 $\times 10^{-4}$ (37944)	9.03 $\times 10^{-16}$ (14800)	0.05660401000444 (0.20537874237421) [0.19785672888684] {0.31512425874406}	0.00524300564371 (0.00524300564371) [0.00528324686652] {0.00596442868126}	0.00524300564371 (0.00524300564371) [0.00524300564371] {0.00524300564371}
1.95 $\times 10^{-1}$ (26800)	0 (27030)	0 (11677)	0.00545312997910 (0.09555910487509) [0.17757858679300] {0.88096806226299}	0.00000000946009 (0.00000000946009) [0.00000000946009] {0.00000000946009}	0.00000000946009 (0.00000000946009) [0.00000000946009] {0.00000000946009}
2.06 $\times 10^{-3}$ (26400)	3.12 $\times 10^{-2}$ (12376)	6.52 $\times 10^{-8}$ (36552)	0.00008133676610 (0.00169425495559) [0.00261982592500] {0.00261982592500}	0.00102257126061 (0.02684854366051) [0.03285315973720] {0.13812378426540}	0.0000000003770 (0.00000863681370) [0.00023548083090] {0.00286311659789}
8.73 $\times 10^{-4}$ (23100)	4.52 $\times 10^{-16}$ (28594)	4.52 $\times 10^{-16}$ (14796)	0.99999330039240 (0.99910000000000) [0.99899256750235] {0.99660000000000}	0.9999999995232 (0.9999999995232) [0.9999999995232] {0.9999999995232}	0.9999999995232 (0.9999999995232) [0.9999999995232] {0.9999999995232}

Table 1 shows that, the accuracy of T_{best} , T_{worst} , $T_{average}$ and T_{median} of IRCSA is all much higher than the corresponding ones of EGA and SCSA. The worst one of T_{sd} of IRCSA is only 6.52×10^{-8} much less than 3.12×10^{-2} of SCSA. The evaluation number has been greatly decreased by IR-operators. Except the Function 5 (whose solution precision has been greatly improved by IRCSA), the T_{av} of IRCSA is not more than half of the others. It is IR-operators that improve the solution accuracy and increase the diversity and convergence of the algorithm.

It is noteworthy that, except f_5 , SCSA all performs better than EGA. But while testing f_5 , SCSA not only can't hunt the global optimum but also performs worse at T_{worst} , $T_{average}$ and T_{median} than EGA. So it is confirmed that, when optimizing the function whose global optimum is in the centre of large-scale flat areas, SCSA is difficulty to break away from the local optimal value. In the contrast, IRCSA performs very well. The algorithm employing IR-operators performs much better at ability of avoiding the premature.

The convergent processes of the tree algorithms when testing show that the convergence speed of IRCSA is obviously higher than SCSA and EGA. IR-operators do improve the convergent speed. And so do the solutions' accuracy.

The simulation results above show that IR-operators do carry the point of accelerating the convergence speed, increasing the accuracy and decreasing the evaluation number. IRCSA has the best performance in metrics such as convergence speed, evaluates, accuracy and stability and prevents prematurely most effectively.

4 Conclusions

In this paper, we put forward four IR-operators inspired by immunologic regulation mechanism, and present a corresponding algorithm, Immunologic Regulation Clonal Selection Algorithm (IRCSA). We make great efforts to study the interrelations between antibodies in the hope of taking the use of them to optimize the algorithm and improve the performance of the algorithm. The simulation comparisons among IRCSA, Genetic Algorithm with elitism preserving strategy and Simple Clonal Selection Algorithm and the analysis of the mechanism and parameters show that, IRCSA has the best performance in metrics such as convergence speed, evaluates, accuracy and stability and prevents prematurely most effectively.

References

1. Dasgupta, D., Forrest, S.: Artificial immune systems in industrial applications. In: IPMM '99. Proceedings of the Second International Conference on Intelligent Processing and Manufacturing of Materials. IEEE press, (1999) 257–267
2. Ding, Y.S., Ren, L.H.: Artificial immune system: theory an application. *Pattern Recognition and Artificial Intelligence*, 13, 1 (2000) 52–59
3. Du, H.F., Jiao, L.C., Gong, M.G., Liu, R.C.: Adaptive Dynamic Clone Selection Algorithms. In: Zdzislaw, P., Lotfi, Z. (eds.): *Proceedings of the Fourth International Conference on Rough Sets and Current Trends in Computing*. Uppsala, Sweden, (2004)
4. Yu, S.Q., Wang, H.G.: *Introduction to Immunology*. CHEP, Springer
5. Chen, G.L., Wang, X.F., Zhuang, Z.Q, Wang, D.S, Wang, Z.D.: *Genetic Algorithm and its Applications [M]*. Beijing: Post & Telecom Press, (1996)
6. Du, H.F., Jiao, L.C., Wang, S.A.: Clonal Operator and Antibody Clone Algorithms. In: Shichao, Z., Qiang, Y., Chengqi, Z. (eds.): *Proceedings of the First International Conference on Machine Learning and Cybernetics*. IEEE, Beijing, (2002) 506–510
7. Gong, M.G., Du, H.F., Jiao, L.C., Wang, L.: Immune Clonal Selection Algorithm for Multiuser Detection in DS-CDMA Systems. *Springer-Verlag, LNCS 3339*, (2004) 1219–1225
8. Jiao, L.C., Gong, M.G., Shang, R.H., DU, H.F., Lu, B.: Clonal Selection with Immune Dominance and Anergy Based Multiobjective Optimization. *Springer-Verlag, LNCS 3410*, (2005) 474–489
9. Gong, M.G., Jiao, L.C., Liu, F. and Du, H.F.: The Quaternion Model of Artificial Immune Response. *Springer-Verlag, LNCS 3627*, (2005) 207–219

A Fault-Tolerant and Minimum-Energy Path-Preserving Topology Control Algorithm for Wireless Multi-hop Networks

Zhong Shen, Yilin Chang, Can Cui, and Xin Zhang

National Key Lab of Integrated Service Network, Xidian University, Xi'an, China
{zhshen, ylchang, ccui, xzhang}@mail.xidian.edu.cn

Abstract. In this paper, we propose a topology control algorithm for constructing an energy-efficient spanning subgraph for a wireless multi-hop network. The constructed topology has the following properties: (1) it preserves a minimum-energy path between every pair of nodes; (2) it is biconnected, i.e., it can tolerate any one node failure and avoid network partition. Simulation results show that the constructed topology has a small average node degree, a small average transmission range and a constant power stretch factor.

1 Introduction

The topology of a wireless multi-hop network has a significant impact on the network performance. Since the basic components of multi-hop wireless networks are mostly battery-operated device, power conservation is one of the key issues of such networks. It is not energy efficient to use the communication network G_{max} where each node transmits with its maximum power. Topology control deals with how to assign the transmission power of each node so that the resulting topology satisfies some specified properties [2]. The importance of topology control lies in the fact that it can conserve battery energy, reduce radio interference and increase spatial reuse of wireless bandwidth [5]. In this paper, we propose a distributed topology control algorithm, called Minimum-energy Biconnected Spanning Subgraph (MBSS), for constructing a topology which preserves two desirable properties: *minimum-energy property* and biconnectivity. A topology is said to have the *minimum-energy property* [1] if it maintains a *minimum-energy path* between every pair (u, v) of nodes, one that allows messages to be transmitted with a minimum use of energy among all the paths between u and v in G_{max} . Besides the *minimum-energy property*, the proposed algorithm also preserves biconnectivity which is a desirable property for network fault tolerance.

2 Network Model

It is assumed that all the nodes are distributed in a two-dimensional plane. Each node has a unique ID, an omni-directional antenna, and a low-power GPS

receiver for acquiring its own location information. Further, each node u is able to adjust its transmission power $p(u)$ within the range $0 \leq p(u) \leq p_u^{\max}$, where p_u^{\max} is the maximum transmission power of node u . A transmission between node u and v at least takes power $p(u, v)$. To facilitate discussion of the proposed algorithm, we first define a few terms. The bidirectional reachable neighborhood $NBR(u)$ is the set of nodes that u can reach by using its maximum transmission power and *vice versa*. The weight of node v in $NBR(u)$ is a pair $w_u(v) = \langle p(u, v), ID(v) \rangle$. Further, for $\forall v_1, v_2 \in NBR(u)$, $w_u(v_1) < w_u(v_2) \Leftrightarrow p(u, v_1) < p(u, v_2)$ or $(p(u, v_1) = p(u, v_2)$ and $ID(v_1) < ID(v_2))$. Given $m, n \in NBR(u)$, we say node m precedes node n (denoted by $m \xrightarrow[u]{Precede} n$) if $p_m^{\max} \geq p(m, n)$, $p_n^{\max} \geq p(m, n)$ and $p(u, m) + p(m, n) < p(u, n)$. $m \xrightarrow[u]{Precede} n$ means that for node u , relaying through m to n takes less power than direct transmission to n .

3 MBSS Topology Control Algorithm

MBSS consists of two phases: first, a connected topology with *minimum-energy property* is constructed; second, each cut vertex is *removed* from the network by augmenting some edges around it, which makes the topology biconnected. For clarity, we refer to the procedure of phase 1 in MBSS as MSS (Minimum-energy Spanning Subgraph) and that of phase 2 as BSS (Biconnected Spanning Subgraph).

3.1 MSS Topology Control Algorithm

Initially, each node u broadcasts a HELLO message using p_u^{\max} and collects all ACK messages from $NBR(u)$. The information contained in a message includes

Algorithm 1. MSS topology control algorithm

- 1: For each node u :
 - 2: $Nbrs(u) = NonNbrs(u) = \emptyset$
 - 3: Find the bidirectional reachable neighborhood $NBR(u)$
 - 4: Sort all nodes in $NBR(u)$ in increasing order by weight
 - 5: **for all** $v \in NBR(u)$ in sorted order **do**
 - 6: **if** $\exists a \in Nbrs(u)$, $a \xrightarrow[u]{Precede} v$ **then**
 - 7: $NonNbrs(u) = NonNbrs(u) \cup \{v\}$
 - 8: **else if** $NP(v) \neq \emptyset$, $NP(v) = \{w | w \xrightarrow[u]{Precede} v, w \in NonNbrs(u)\}$ **then**
 - 9: Find the node d which has the minimum weight in $NP(v)$
 - 10: $NonNbrs(u) = NonNbrs(u) - \{d\}$
 - 11: $Nbrs(u) = Nbrs(u) \cup \{d\}$
 - 12: $NonNbrs(u) = NonNbrs(u) \cup \{v\}$
 - 13: **else**
 - 14: $Nbrs(u) = Nbrs(u) \cup \{v\}$
 - 15: **for all** $w \in Nbrs(u)$ **do**
 - 16: send a Symmetry Request Message
 - 17: $p(u) = \max_{a \in Nbrs(u)} p(u, a)$
-

the ID, the position and the maximum power of the sending node. After obtaining the information of $NBR(u)$, node u sorts all the nodes in $NBR(u)$ in increasing order by weight. Then, node u determines its set of neighbors such that for any node n not in $Nbrs(u)$, there exists a node m in $Nbrs(u)$ which precedes n , and the maximum weight in $Nbrs(u)$ is minimized. Note that the neighbor relation may not be symmetric. For instance, $v \in Nbrs(u)$ but $u \notin Nbrs(v)$, which may result in a unidirectional link from u to v . To make the bidirectional links between u and its neighbors, node u sends a Symmetry Request (SR) message which contains its ID and position to each node v in $Nbrs(u)$. Upon receiving the SR from u , v will guarantee its final power not smaller than $p(u, v)$. Finally, node u sets its transmission power required to reach its neighbor of maximum weight.

Let graph G_{max} denote the maximum topology where each node transmits with its maximum transmission power, graph G the topology derived by MSS. The following theorem holds.

Theorem 1. *Given G_{max} is connected, G is connected and has the minimum-energy property.*

Proof. By definition, the topology that has the *minimum-energy property* must be connected. For $\forall u, v \in V(G)$, let path $(u = u_0, u_1, \dots, u_{n-1}, u_n = v)$ be the *minimum-energy path* from u to v in G_{max} . Obviously, it follows $u_{i+1} \in Nbrs(u_i)$, $i = 0, 1, \dots, n - 1$. Thus, G has the *minimum-energy property*.

3.2 BSS Topology Control Algorithm

The topology G derived by MSS servers as the basis for BSS. We can identify the biconnected components and cut vertices in G by using the distributed algorithm given in [7]. It is noted that two different biconnected components should not

Algorithm 2. BSS topology control algorithm

- 1: For each cutpoint c of G :
 - 2: Biconnected components B_1, \dots, B_n connected at c
 - 3: $F_c = \{(u, v) | (u, v) \in E(G_{max}) - E(G), u \in B_i, v \in B_j, i \neq j, u, v \in Nbrs(c)\}$
 - 4: initialize $|Nbrs(c)|$ clusters for the neighbors of c , one per node
 - 5: **for all** $u, v \in B_i, u, v \in Nbrs(c), n \geq i \geq 1$ **do**
 - 6: merge cluster(u) with cluster(v)
 - 7: Sort all edges in F_c in increasing order by length
 - 8: **for all** (u, v) in sorted order **do**
 - 9: **if** cluster(u) \neq cluster(v) **then**
 - 10: **if** $p(u) < p(u, v)$ **then**
 - 11: $p(u) = p(u, v)$
 - 12: **if** $p(v) < p(u, v)$ **then**
 - 13: $p(v) = p(u, v)$
 - 14: merge cluster(u) with cluster(v)
 - 15: **if** the number of clusters is 1 **then**
 - 16: **end**
-

have any common edges (but they can have common vertex), and the common vertex linking two (or more) biconnected components must be a cut vertex of G .

Given any cut vertex c in G , let $Nbrs(c)$ represent the set of neighbors of c , and B_1, \dots, B_n be the biconnected components connected at c , $n \geq 2$. The main idea of BSS is to augment some *feasible* edges to G so as to *remove* this cut vertex from the topology. Formally, a edge (u, v) is a *feasible* edge if $(u, v) \in E(G_{\max}) - E(G)$, $u \in B_i, v \in B_j, i \neq j, n \geq i, j \geq 1$, and $u, v \in Nbrs(c)$. Furthermore, in order to minimize the increment power of neighbors of cutpoint c , we use a lowest-cost way to add *feasible* edges to connect all of the connected components formed by the deletion of c , which is similar to the *minimum cost spanning tree* algorithm.

Let G_B be the topology derived by BSS, the following theorem holds.

Theorem 2. *If G_{\max} is biconnected then G_B is also biconnected.*

Proof. For any given cutpoint c in G derived by MSS, let B_1, \dots, B_n be the biconnected components of G connected at c , and $C_1(c), \dots, C_n(c)$ be the corresponding connected components formed by the deletion of c from G . It follows $B_i - \{c\} \subseteq C_i(c), n \geq i \geq 1$. In addition, we refer to an edge (u, v) as a redundant edge if $(u, v) \in E(G_{\max}) - E(G)$.

Consider redundant edge $(u, v), u \in C_i(c), v \in C_j(c), i \neq j$. Since (u, v) is a redundant edge, by MSS, there must exist a path $r = (u = u_0, u_1, \dots, u_{n-1}, u_n = v)$ from u to v such that $v \in NBR(u_l)$ and $u_{l+1} \xrightarrow[u_l]{Precede} v, l = 0, \dots, n - 1$. Note that the cutpoint c lies on all paths between $C_i(c)$ and $C_j(c)$. Therefore, path r has to contain the cutpoint c and traverse both the B_i and B_j , so there is a node u' which is on the path r and belongs to $B_i - \{c\}$. Since $v \in NBR(u')$, $(u', v) \in E(G_{\max})$. Moreover, (u', v) is a redundant edge; otherwise $C_i(c)$ and $C_j(c)$ are connected. Now we consider edge (u', v) . For node v , there also exists a path $r' = (v = v_0, v_1, \dots, v_{m-1}, v_m = u')$ from v to u' such that $u' \in NBR(v_k)$ and $v_{k+1} \xrightarrow[v_k]{Precede} u', k = 0, \dots, m - 1$, and r' has to contain the cut vertex c and traverse B_i and B_j . Thus, there is a node v' which is on the path r' and belongs to $B_j - \{c\}$. Since $u' \in NBR(v')$, it follows $(u', v') \in E(G_{\max}) - E(G)$ where $u' \in B_i - \{c\}$ and $v' \in B_j - \{c\}$. By the same argument, for redundant edge (u', v') , it is easy to show that there also exists a redundant edge between two neighbors of cutpoint c with one in B_i and the other in B_j .

Since c is a cutpoint in $G, G - c$ is disconnected. Based on the observation that an edge belonging to a same component, say $C_i(c)$, does not contribute to the connectivity of $G - \{c\}$, so there must exist a set of edges $E_a = \{(u, v) | (u, v) \in E(G_{\max}) - E(G), u \in C_i(c), v \in C_j(c), i \neq j, n \geq i, j \geq 1\}$ such that $(G - \{c\}) \cup E_a$ is connected. Similarly, we can also find a corresponding set of *feasible* edges $E''_a = \{(u'', v'') | (u'', v'') \in E(G_{\max}) - E(G), u'' \in B_i, v'' \in B_j, u'', v'' \in Nbrs(c)\}$ such that $(G - \{c\}) \cup E''_a$ is connected. Since $E''_a \subseteq F_c$ and all the edges in F_c have been considered by BSS, hence node c is not a cut vertex in G_B and G_B is biconnected.

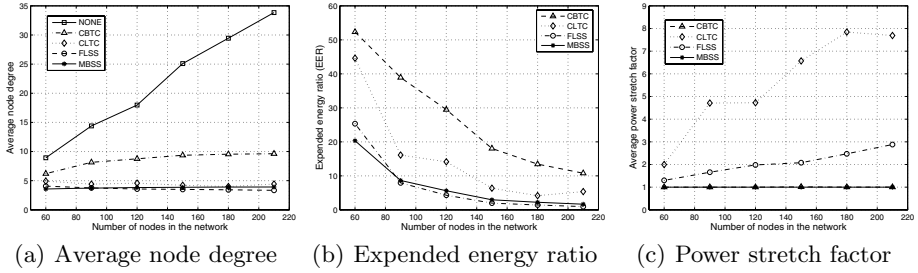


Fig. 1. Performance comparison of CBTC, CLTC, FLSS and MBSS

4 Simulation Results

We evaluate the performance of MBSS against CBTC($\pi/3$) [4], CLTC-A with optimization objective of MINMAX [6] and FLSS₂ [5] with respect to several metrics through simulations. In the simulations, nodes are uniformly distributed in a 2000m \times 2000m region. The maximum transmission range of a node is 500m and the power attenuation constant is set to 4. Each node has an antenna with 0dB gain and 1.5m height. The receive threshold is -94dBW. Each data point is the average of 50 simulation runs.

We compare the average node degree of the topologies derived by different algorithms. A smaller average node degree usually implies less contention and better spatial reuse. Fig. 1(a) shows the average node degree of the topologies derived under CBTC($\pi/3$), CLTC, FLSS₂ and MBSS. The average node degree under NONE (with no topology control) increases almost linearly with the number of nodes. The average degree under CBTC($\pi/3$) is much higher than those of CLTC, FLSS₂ and MBSS. This is mainly due to the fact that for each node u , CBTC($\pi/3$) requires the maximum angle between its two consecutive neighbors is at most $\pi/3$, while the other three algorithms do not have such a constraint. The node degree under MBSS is smaller than that of CLTC and is very close to that of FLSS₂.

Fig. 1(b) shows the average expended energy ratio (EER) [3] of various algorithms. The average transmission power determined by MBSS or FLSS₂ is much smaller than that by CBTC($\pi/3$). It can be seen that MBSS performs slightly better than FLSS₂ in the networks of low density while in the networks of high density, FLSS₂ works better. We also find that the clustering algorithm and the distribution of nodes have an impact on the performance of CLTC. Therefore, how to form optimal clusters under various situations is a challenge for CLTC.

Fig. 1(c) give the average power stretch factor [8]. Since MBSS preserves the *minimum-energy property*, its power stretch factor equals 1. In contrast, CBTC($\pi/3$), CLTC and FLSS₂ do not give any bound on power stretch factor. CLTC has a very large power stretch factor, which indicates that the energy consumption in a transmission between some nodes in the topology derived by CLTC is much higher than that by MBSS. The power stretch factor of CBTC($\pi/3$) is

very close to 1 because each node running CBTC($\pi/3$) attempts to keep neighbors in each cone of $\pi/3$ and the resultant topology is much denser than those of CLTC, FLSS₂ and MBSS. Finally, although FLSS₂ has a small average transmission power, its power stretch factor slowly increases as the node density increases.

5 Conclusion

In this paper, we present a distributed topology control algorithm for constructing a topology satisfying two desirable properties: *minimum-energy property* and biconnectivity. Simulation results show that the constructed topology has a small average node degree, a small average transmission range and a constant power stretch factor.

References

1. L. Li and J.Y. Halpern: A minimum-energy path-preserving topology-control algorithm. IEEE Trans. Wireless Communications, Vol. 3. (2004) 910 - 921
2. E. L. Lloyd, R. Liu, M. V. Marathe, R. Ramanathan, and S. S. Ravi: Algorithmic aspects of topology control problems for ad hoc networks. In: Proc. ACM MOBIHOC(2002) 123–134
3. M. Hajiaghayi, N. Immorlica, and V. S. Mirrokni: Power optimization in fault-tolerant topology control algorithms for wireless multi-hop networks. In: Proc. ACM MOBIHOC(2003) 300–312
4. M. Bahramgiri, M. Hajiaghayi and V. S. Mirrokni: Fault-tolerant and 3-dimensional distributed topology control algorithms in wireless multi-hop networks. In: Proc. Eleventh International Conference on Computer Communications and Networks (ICCCN)(2002) 392–397
5. N. Li and J. C. Hou: FLSS: A fault-tolerant topology control algorithm for wireless networks. In: Proc. ACM MOBIHOC(2004) 275–286
6. C.-C. Shen, C. Srisathapornphat, R. Liu, Z. Huang, C. Jaikaeo and E. L. Lloyd: CLTC: a cluster-based topology control framework for ad hoc networks. IEEE Trans. on Mobile Computing, Vol. 3. (2004) 18 - 32
7. A. Kazmierczak and S. Radhakrishnan: An optimal distributed ear decomposition algorithm with applications to biconnectivity and outerplanarity testing. IEEE Trans. on Parallel and Distributed systems, Vol. 11. (2000) 110 - 118
8. S.-C. Wang, D. S. L. Wei and S.-Y. Kuo: A topology control algorithm for constructing power efficient wireless ad hoc networks. In: IEEE GLOBECOM(2003)1290–1295

Computational Biomechanics and Experimental Verification of Vascular Stent

Yuexuan Wang, Hong Yi, and Zhonghua Ni

Department of Mechanical Engineering,
Southeast University,
Nanjing 210096, China
wangyx100@hotmail.com

Abstract. Vascular stent is a small tubular device expanded into stenotic artery to restore natural blood flow. This paper introduces application background and research overview of medical vascular stent. A tubular mini stent was specially designed and manufactured for the small diameter coronary vessels. A computational and experimental method of research on biomechanics of vascular stent was presented. Computational simulation of stent deployment expanded by the balloon based on nonlinear finite element analysis was performed including large displacement and deformation, geometric and material nonlinearity. The experimental platform of in vitro stent expansion based on the machine vision technology was established and the image processing software was developed. The fabricated stent was tested on the assembled experimental equipments. Matching between the computational and experimental results was quite satisfactory. The experimental scheme provides powerful support for the computational analysis of stent biomechanics.

1 Introduction

Vascular stent is an endovascular prosthetic device supporting a blood vessel permanently open and compressing plaques protruding into the vessel lumen. The implantation of coronary stent is utilizing catheter balloon to position and deploy the stent in the vascular lumen where the plaques exit to prevent recoil of the atherosclerotic vessel wall. Nowadays stent interventional therapy prevails as a typical treatment method for the coronary heart diseases.

The ideal stent possesses a low profile, good flexibility to navigate tortuous vessel, adequate radiopacity, low recoil, sufficient radial strength, a low metal surface area, high scaffolding ability, and thrombo-resistivity. The performance of stent expansion is vital to succeed in operation of implantation and prove if the stent can meet the clinical requirements. Therefore, much attention was paid to the research on the characteristics of stent expansion. The stent expansion is a complicated process in which the stent behaviors are full of complication and difficult to predict.

The main objectives of this paper are to investigate biomechanics of stent by computational analysis based on finite element method and verify the results of numerical simulation by the in vitro experimental scheme of a real stent expansion.

2 Research Overview

The performance of stent can be observed via animal experiment (for example reference [1]), but the period of observation is too long. So researchers studied the characteristics of stent by the in vitro experiment. Reference [2] compared the mechanical properties such as profile, elastic recoil and stent foreshortening based on the measurements obtained from the five different stents. Reference [3] presented a stereo-photogrammetric system for the three-dimensional reconstruction of stent geometry during expansion to aid future balloon and stent design.

Computational simulation of stent expansion began to gain the recognition from engineering prospect when the nonlinear finite element analysis (FEA) software became available. Many researchers deem FEA as an alternative to experimental test as it possesses a few advantages such as cost and difficulty of experimental test. For example, references [4,5] performed finite element simulation of stent expansion to analyze the interactions between the stent and the plaque or artery. References [6,7] applied the finite element method to investigate the effects of different geometrical parameters of a typical coronary stent on the stent mechanical performance and investigated the stent flexural ability to bend in order to accommodate curvature and angle of vessel during delivery. Reference [8] studied and compared the mechanical characteristics of two different type of tubular and coil stents. Reference [9] evaluated the mechanisms of the Palmaz stent expansion and its long-term behaviors. Reference [10] focused on biomechanical interaction between a balloon-expandable stent and a stenotic artery.

Although using computational simulation it allows researchers to look closely into the specific and interested problems they never think could be possible to get answers by the clinical or in vitro experiment, the results of computation should be validated by the experiment. However, all the simulated achievements above did not design the experimental scheme to verify the computational results. Their results could be better convinced if they had considered experimental verification problem. As a result, further studies combined with experiments are still necessary. Therefore, in this paper we aim to investigate the biomechanics of stent by computational simulation combined with the experiment deploying a real stent by the balloon expansion.

3 Computational Simulation

3.1 Modeling of Stent

Most stents are designed for use in coronary arteries of > 3.0 mm. However, clinical studies have demonstrated that about 40% of coronary lesions eligible for stenting are located in vessels with a diameter of < 3.0 mm. Small coronary stenting requires improvement of several stent characteristics. Flexibility, radial strength, percentage open area, wall thickness, range of available length and ease of deliverability are the principal features. In our study the stent to be simulated by balloon expansion is illustrated in the Fig.1. This stent CAD model was specially designed for treating small diameter coronary vessels with de novo and restenotic lesions by the PRO/E

modeling tool. Compared to previous stents its structural geometry features repeating “S” type of bridging links pattern, which offers significant longitudinal flexibility.

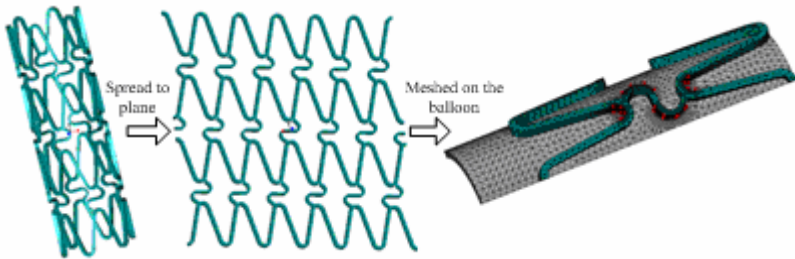


Fig. 1. Stent model

A large deformation analysis was performed using ANSYS commercial code based on the finite element method. By taking the advantage of symmetrical conditions, only one-eighth of the stent CAD model was exported from PRO/E as an IGES file. Furthermore, one-eighth of the balloon geometry was developed by ANSYS. The balloon was placed inside the stent with its outside diameter being equal to the inside diameter of the stent. In reality the stent is expanded by the balloon as the stent is crimped on the delivery system.

The assembled model of balloon and stent was automatically meshed using 10-node tetrahedral solid elements by ANSYS as shown in Fig.1. The finer meshes were specified at the bridging links and rings near the bridging links to accurately investigate the stress generated. The material property of 316L stainless steel was utilized for the stent and described through a Von Mises-Hill plasticity model with isotropic hardening. The Mooney-Rivlin model was used to represent the balloon.

3.2 Loading and Solution

The process of stent expansion is described as follows: Inflation of balloon allows deployment of the stent crimped on the balloon. After deflation of the balloon (unloading) the stent undergoes intrinsic elastic recoil that reduces its diameter from the inflated maximum. Subsequently, the force exerted by the arterial wall further compresses the stent. The total recoil corresponds to the sum of the intrinsic elastic recoil plus compression exerted by the diseased arterial wall. The compression of arterial wall was ignored in our computation. The simulation of the stent expansion was carried out with pressure load varying in three stages: uniform linearly increasing radial pressure, constant pressure and uniform linearly decreasing radial pressure. These three stages were used to simulate the inflation and deflation of the balloon.

Symmetric boundary conditions were imposed on the nodes of the stent and balloon in the planes of symmetry where all the nodes belonging to a symmetric plane are required to have zero displacement in the direction normal to the symmetric plane and just allowed to move within the plane of symmetry. All the central nodes were constrained in axial direction and the distal nodes were free from any constraint so that the behaviors of expanding and shortening would be investigated. The balloon

was fully tethered at both ends and hence only the expansion in radial direction was permitted. An automatic surface to surface algorithm was selected in order to cope with the nonlinear contact problem between the contact surfaces. The outer surface of the balloon was allowed to contact with the inner surface of the stent. The nonlinear problem was solved using a Newton-Raphson method due to material plasticity and contact constraint.

3.3 Results and Analysis

Stress Analysis within Stent. Fig.2 shows the distribution of residual stress in the stent after the balloon deflation. It can be seen from the figure that the highly stressed areas were localized in the corners of the rings. The maximum residual stress that remained in the stent after deflation was approximately 475 MPa approaching to the tensile strength 515 MPa of 316L steel. We can see that the minor stressed regions were located at the middle zones of the rings and the middle bridging links.

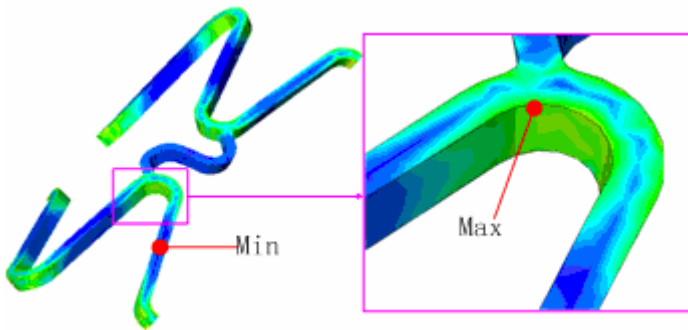


Fig. 2. Residual stress in the deformed stent

Characteristic of Diameter Expansion. Fig.3 shows the radial displacement of the central and distal vertices of stent at different levels of deployment pressure when

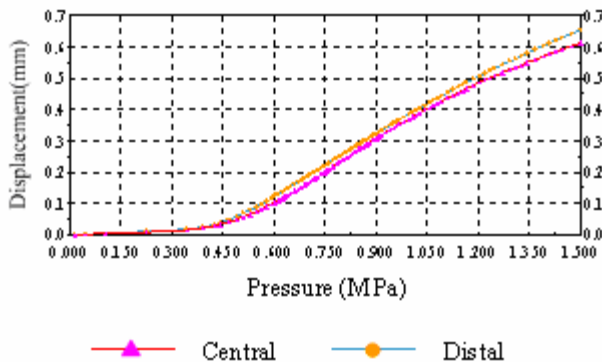


Fig. 3. Radial displacement of the central and distal vertices with loading pressure

loading. From the Fig.3 we can see a stronger expansion of the distal ends with respect to the central zones. We quantified this undesired phenomenon called dogboning characteristic of stent by calculating a ratio as $\frac{D_{distal}^{loading} - D_{central}^{loading}}{D_{central}^{loading}} \cdot D_{distal}^{loading}$

and $D_{central}^{loading}$ are respectively the distal and central diameter at the maximum pressure.

Foreshortening. Fig.4 illustrates the axial displacement of the distal vertex with loading pressure. Here the displacement of the distal vertex is equal to the half of magnitude of foreshortening with deployment pressure during the inflation process. It can be seen from the figure that the foreshortening increased quickly after the beginning of the stent deployment. Any alteration in the stent diameter reflected the changes in the length of stent because the foreshortening is related to the stent diameter and any reduction in length of the stent is accompanied by a corresponding increase in stent diameter. We quantified the maximum foreshortening characteristic

of stent by calculating a ratio as $\frac{L - L^{loading}}{L}$. Here L is initial length of stent and $L^{loading}$ is the stent length at the maximum loading pressure.

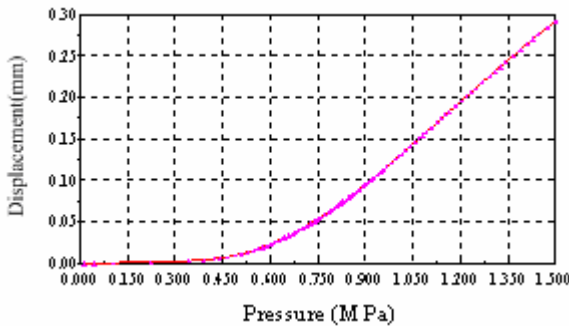


Fig. 4. Axial displacement of the distal vertex with loading pressure

Elastic Recoil. Fig.5 shows the radial displacement of the distal and central vertices during the balloon deflation when unloading. The curves reflect that the diameter of the stent decreased with the decreasing pressure load. After the balloon was removed eventually the stent diameter stopped decreasing. This is due to the separation of the balloon from the stent and the fact that the stent had undergone the plastic deformation. The radial strength of the stent kept the tube in deformed shape and prevented it from recoil.

In order to quantify the elastic recoil characteristic of stent the following quantities were defined and calculated in the Table 1.

- The ratio of distal and central radial recoil was respectively defined as $\frac{D_{distal}^{loading} - D_{distal}^{unloading}}{D_{distal}^{loading}}$ and $\frac{D_{central}^{loading} - D_{central}^{unloading}}{D_{central}^{loading}}$. Here $D_{distal}^{unloading}$ and $D_{central}^{unloading}$ are respectively the distal and central diameter after the balloon was deflated.

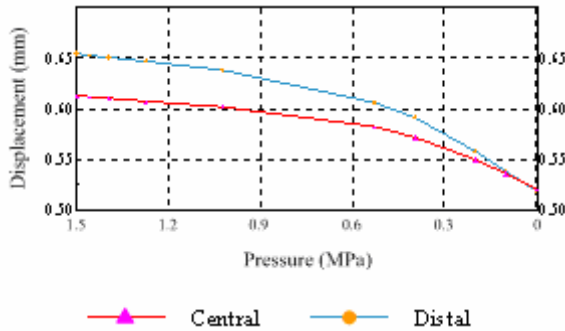


Fig. 5. Radial displacement of the central and distal vertices with unloading pressure

- The ratio of axial recoil was defined as $\frac{L^{unloading} - L^{loading}}{L^{loading}}$. Here $L^{unloading}$ is the stent length after the balloon was deflated.
- The ratio of the ultimate foreshortening was defined as $\frac{L - L^{unloading}}{L}$.

Table 1. Calculated ratio

Dogboning ratio	2.97%
Maximum foreshortening ratio	5.69%
Ultimate foreshortening ratio	4.38%
Distal radial recoil ratio	9.43%
Central radial recoil ratio	7.18%
Axial recoil ratio	1.58%

4 Experimental Verification

For the verification of simulated results mentioned above we designed the experimental system based on the machine vision technology for the in vitro stent expansion to investigate the expansion behavior of a balloon-expandable stent with pressure applied as a surface load on the inner surface of the balloon. Fig.6 shows the schematic diagram of the experimental equipments. The JAI CV-A1 CCD camera with the TEC-M55 telecentric lens was positioned perpendicular to the stent crimped on a balloon catheter to record the process of stent expansion deformation. The NI 1409 image acquisition device captured the images of stent expansion from output of camera. The image processing software was programmed using Vision Development Module in LabVIEW. The pressure inside the balloon was acquired from the pressure transducer by the NI 6036E data acquisition device.

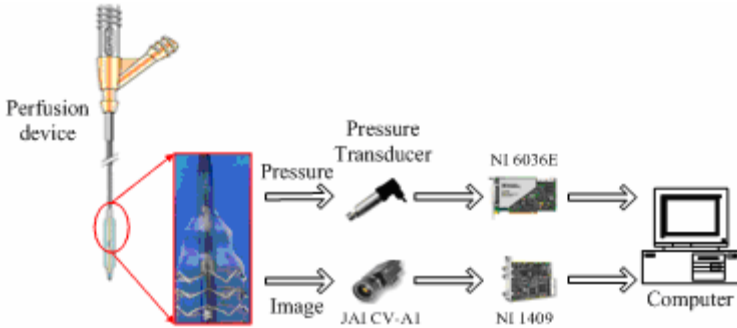


Fig. 6. The schematic diagram of the experimental equipments

We fabricated a stent whose geometrical structure was shown in the Fig.1 by laser cutting, polishing, and vacuum heat treatment from a smaller and thinner wall tube of 316L stainless steel. This stent was expanded on the presented experimental platform shown in Fig.6. In the beginning of expansion both of the distal ends had a quicker expansion than the central zones. But in the end the distal and central zones had approximately the same diameter.

The analysis of the experimental results shows that the process of the stent diameter increase does not proceed proportionally to the given deployment pressure in the balloon. A sudden increase of the diameter was observed after exceeding the given critical deployment pressure (0.4 MPa). The further increase of the deployment pressure (1.5 MPa) did not cause the increase of the diameter. Although the errors shown in the Table 2 between the results obtained from the experiment and computational simulation is a little high. We still regard these results as being within the normal range. Because the stent was subject to chemical polishing after it was cut by laser cutting machine. As a result, the stent eventual geometrical dimensions were reduced and so its “stiffness” also decreased. So the stent needed less pressure to be expanded to the same diameter as used in computational simulation.

Table 2. Comparison between computational analysis and experimental expansion (unit: mm)

	Computational analysis	Experimental expansion	Discrepancy	Percentage of error
Ultimate distal diameter	2.5648	2.975	0.4102	13.79%
Ultimate central diameter	2.5526	2.880	0.3274	11.37%
Ultimate length	9.8554	9.712	0.1434	1.48%

5 Conclusion

This paper presented the methodology for biomechanics of the vascular stent based on computational analysis and experimental test. A high correlation between results obtained from the experiment and with the use of the computational simulation proves that computational analysis could help to design competing stent alternatives with excellent biomechanics prior to costly prototype fabrication. Moreover, the presented

experimental scheme provides powerful experimental support for the optimization of stent design and computer simulation of stent expansion and also extends the application of machine vision technology in the field of medical and micro devices. The main limitation of this study is that the atherosclerotic arteries in which the stents are implanted are not modeled and the interactions between the stents and arteries are not considered. These are further research works we will be engaged on.

References

1. Eigler N., Whiting J., Li A., Frimerman, A.: Effects of a Positron-Emitting Vanadium-48 Nitinol Stent on Experimental Restenosis in Porcine Coronary Arteries: An Injury - Response Study. *Cardiovascular Radiation Medicine*. **1** (1999) 239-251
2. Schmidt W., Behrens P., Behrend D., Schmitz, K.-P.: Experimental Study of Peripheral, Balloon-Expandable Stent Systems. *Progress in Biomedical Research*. (2001) 246-254
3. Narracott A.J., Hose D.R., Lawford P.V., Gunn, J.: Measurement of the Symmetry of in Vitro Stent Expansion: A Stereo-Photogrammetric Approach. *Journal of Medical Engineering & Technology*. **27** (2003) 59-70
4. Chua, S.N.D., MacDonald, B.J., Hashmi, M.S.J.: Finite-element simulation of slotted tube (stent) with the presence of plaque and artery by balloon expansion. *Materials Processing Technology*. **155-156** (2004) 1772-1779
5. Chua, S.N.D., MacDonald, B.J., Hashmi, M.S.J.: Effects of varying slotted tube (stent) geometry on its expansion behaviour using finite element method. *Materials Processing Technology*. **155-156** (2004) 1764-1771
6. Migliavacca, F., Petrini, L., Montanari, V., Quagliana, I.: A predictive study of the mechanical behaviour of coronary stents by computer modelling. *Medical Engineering & Physics*. **27** (2005) 13-18
7. Petrini, L., Migliavacca, F., Auricchio, F., Dubini, G.: Numerical investigation of the intravascular coronary stent flexibility. *Journal of Biomechanics*. **4** (2004) 495-501
8. Etave, F., Finet, G., Boivin, M.: Mechanical properties of coronary stents determined by using finite element analysis. *Journal of Biomechanics*. **34** (2001) 1065-1075
9. Dumoulin, C., Cochelin, B.: Mechanical behaviour modelling of balloon-expandable stents. *Journal of Biomechanics*. **33** (2000) 1461-1470
10. Auricchio, F., Loreto, M.D., Sacco, E.: Finite-element analysis of a stenotic artery revascularization through a stent insertion. *Computer Methods in Biomechanics and Biomedical Engineering*. **00**(2000) 1-15

Numerical Computing of Brain Electric Field in Electroencephalogram

Dexin Zhao^{1,2}, Zhiyong Feng¹, Wenjie Li², and Shugang Tang²

¹ Department of Computer Science and Technology,
Tianjin University, Tianjin 300072, China
zhaodexin@eyou.com, zyfeng@tju.edu.cn

² Department of Computer Science and Engineering,
Tianjin University of Technology, Tianjin 300191, China
lwj13579@sina.com, tsg@tjut.edu.cn

Abstract. This paper expatiated on the process of numerical computing of brain electric field in electroencephalogram (EEG). Based on boundary element method (BEM), a 3D reconstruction of computing model is presented first, which is the premise of BEM computing in EEG. A simple but efficient triangular mesh generation method with constrained points is developed, furthermore, a mesh subdivision method is also put forward. Forward computation of EEG is investigated and acceptable results are obtained with the simulated experiments by using these methods.

1 Introduction

Electroencephalogram (EEG) is a non-invasive technique that measures the electric field of the brain, by placing electrodes on the scalp, which is regarded as a satisfactory way to explore brain functions. Recently, the research on this field has become more and more important in clinical applications, such as pre-surgical planning, pathology diagnosis and computer integrated surgery.

Generally, bioelectromagnetic processes can be considered as a quasi-static current field, and the equivalent current dipoles can be adopted to simulate the active neural sources [1]. The computation of the location of the electric activity in the brain is called source location or solving the inverse problem of biomagnetism. The brain electric field problem (also called EEG forward problem) is to obtain the potential distribution based on known dipoles' positions and the moment in the brain. Some geometric and electromagnetic constraints must be added to obtain the most reasonable solution. Based on the distributed dipole source model, that is, the positions of dipoles being restricted to a discrete solution space, the linear relationship of the current density \mathbf{J} at M nodes of the solution space and the electric potential \mathbf{u} at the N scalp electrodes can be expressed by

$$\mathbf{u}=\mathbf{G}\mathbf{J}$$

If the strength of current density is considered in three directions, \mathbf{J} is a $3M \times 1$ vector; \mathbf{G} is a $N \times 3M$ lead field matrix, which can be calculated by solving the forward problem [2].

Up to now, there is no efficient approach to describe the conductivity distribution of every tissue in the head thoroughly and correctly. The boundary element method (BEM) seems to be a more practical one to solve the EEG forward problem, due to its consideration of the average conductivity of a concrete form. Moreover, BEM can reduce the computing complexity for that it only requires discreteness on the boundary of different regions. Many related problems in BEM solution process are studied in the paper. In addition, a computing head model that has a realistic geometry shape of head tissues must be given first. In this paper, an interactive segmentation method of 3D medical images based on gray-level information is presented. Then a constrained triangular mesh generation method through characteristic points is developed.

2 Construction of BEM Computing Models from MRI

2.1 Brain Segmentation

Segmenting and extracting tissues or organs from medical images are the premises of accurate reconstruction 3D models. In this paper, using magnetic resonance images (MRI), realistic head and brain models are reconstructed by an interactive segmentation method based on gray-level information.



Fig. 1. Segmented results

The segmentation process consists of following steps: filtering the images, creating the threshold values after analyzing the gray-level histograms of the whole 3D image and some slice images, binarizing the images, processing the images using properly mathematical morphology operations (such as erode and dilate operations) according to the feature of tissues or regions to extract. An example result is shown in Fig.1. The data obtained from head and brain is saved in matrixes according to our designed data structure which is convenient for the generation of triangular mesh.

2.2 Triangular Mesh Generation Through Constrained Points

For electromagnetic calculation in EEG, it is necessary to establish a head mesh model that the triangular vertexes can track through the electrode positions. Besides, there are some regions that have pronounced curvature, which contained characteristic information of the model, these points in the regions should be chosen as mesh points, in order to ensure the reconstruction 3D model is realistic. A novel and efficient triangular mesh generation method through given positions is developed as below.

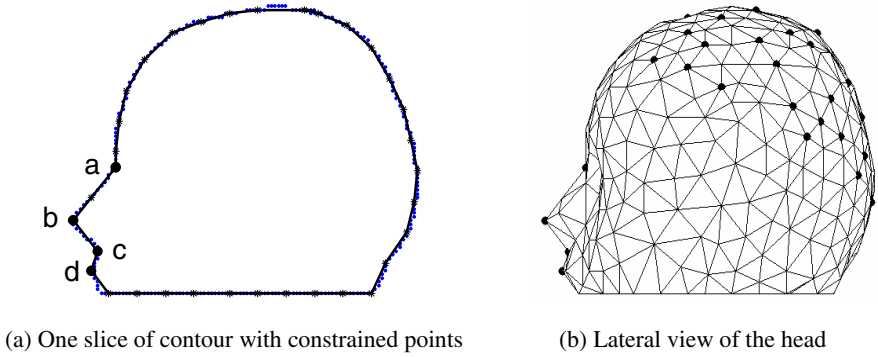


Fig. 2. Generation of the triangular mesh through constrained points

Firstly, to stack uniformed contours together, set random points as constrained points in order to test the program. As shown in Fig.2 (a), points a, b, c, d are constrained points.

Secondly, to estimate contour sampling step according to the distance of adjacent constrained points, the sampling step may be several different values under the condition of more than two constrained points. In Fig.2 (a), the star-like points represent the mesh points on the basis of this algorithm.

Thirdly, repeat the above-mentioned process, a mesh points matrix is achieved.

Finally, generate the BEM triangular model by shortest distance method, for details, please refer to [3]. Head mesh generation results with constrained points (the black points on the head) are shown in Fig.2 (b), and the result of brain triangulation mesh generation is shown in Fig.3.

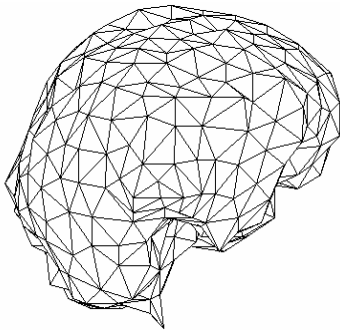


Fig. 3. Result of brain triangular mesh generation

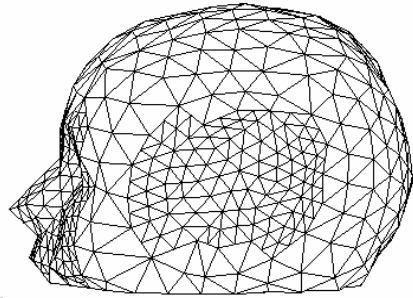


Fig. 4. An example of mesh subdivision

For the numerical computing, mesh subdivision may be necessary for the need of computing precision or the interests of some regions. A simple method is presented which can subdivide the interests area. The process is described in below:

Firstly, to find the midpoints of each side of the triangles in the region that we want to subdivide, and connect the midpoints of each triangle with lines in the area, then a

triangle will turn to four small ones with fine regular triangular shape. Secondly, save the data of the midpoints and the newly generated triangles in related matrix. Finally, reuse the mesh generation algorithm that is presented before, a good subdivided result is obtained. An example is shown in Fig.4 with the subdivisions of the face and the slide head.

3 Numerical Computing in EEG Based on BEM

3.1 BEM Solutions of EEG Forward Problem

Using the quasi-static electromagnetic field Maxwell’s equation, the solution of Poisson’s equation for a piecewise homogeneous volume conductor with closed surface S_k of the conductivity boundaries is given by [4]:

$$u_{S_k}(\mathbf{r}) = \frac{2\sigma_s}{\sigma_k^- + \sigma_k^+} u^\infty(\mathbf{r}) + \frac{1}{2\pi} \sum_{l=1}^m \frac{\sigma_l^- - \sigma_l^+}{\sigma_k^- + \sigma_k^+} \sum_{j=1}^{n_l} \iint u_{S_l}(\mathbf{r}') d\Omega(\mathbf{r}') \quad (1)$$

where $d\Omega(\mathbf{r}') = ((\mathbf{r}' - \mathbf{r}) / |\mathbf{r}' - \mathbf{r}|^3) \cdot \mathbf{n} dS_l$ is solid angle, \mathbf{n} is the surface normal on S_k , \mathbf{r}' is the vector pointing to the source and \mathbf{r} is the field point, σ_l^- and σ_l^+ are the conductivities on the inner and outer side of the surface S_k , and σ_s is the conductivity in the volume containing the sources. When the sources only exist at discrete points, for example, there is an electric current dipole \mathbf{P} at \mathbf{r}_0 , the $u^\infty(\mathbf{r})$ can be calculated by

$$u^\infty(\mathbf{r}) = (\mathbf{D} / 4\pi\sigma) ((\mathbf{r} - \mathbf{r}_0) / |\mathbf{r} - \mathbf{r}_0|^3)$$

when all m boundaries are meshed into small enough triangular elements, (1) can be written in matrix notation as

$$\mathbf{U} = \mathbf{G} + \mathbf{B}\mathbf{U} \quad \text{or} \quad \mathbf{u}^k = \mathbf{g}^k + \sum_{l=1}^m \mathbf{B}^{kl} \mathbf{u}^l \quad k = 1, \dots, m$$

where u^k and g^k are $n'_k \times 1$ vectors (n'_k is the number of nodes on surface S_k), respectively, made up of each potential u_i^k and $g_i^k = 2\sigma_s u_{k,i}^\infty / (\sigma_k^- + \sigma_k^+)$. \mathbf{B}^{kl} is the $n'_k \times n'_l$ matrix related to the solid angles. Based on different discrete basic functions, there are more or less differences in constructing \mathbf{B}^{kl} . In order to get unique \mathbf{U} , deflation technique is employed, i.e., corresponding to matrix \mathbf{B} , introduce deflated matrix \mathbf{C} : $C_{ij}^{kl} = B_{ij}^{kl} - 1/N$, where N is the sum of the triangles on all m boundary surfaces. Then, we obtain

$$\mathbf{U} = (\mathbf{I} - \mathbf{C})^{-1} \mathbf{G}$$

If the dipole sources \mathbf{P} have unit strengths, \mathbf{U} is equal to the lead field matrix \mathbf{G} .

To overcome the numerical error that has been caused by the relatively low conductivity of the skull compared with other tissues, the isolated problem with an adjustable parameter is introduced [5].

3.2 Results of Simulation

To verify the software that has been developed according to the above idea, a four-layer-sphere head model is computed and the results are compared with the analytical solutions [6]. The four homogeneous compartments represent the brain, CSF, skull, and scalp with the radii of 8.4cm, 8.67cm, 9.47cm and 10cm respectively. And their conductivity values are (0.33, 1, 0.0042, 0.33) s/m. To quantify the numerical errors, we define

$$RDM^* = \sqrt{\sum_{i=1}^M (u_i^A / \sum_{i=1}^M u_i^A - u_i^N / \sum_{i=1}^M u_i^N)^2} * 100\%$$

to show the electric field differences between the analytically calculated potential u_i^A and the BEM potentials u_i^N at the electrode i ($i=1,2,\dots,N$) on the outer surface of the head model. Each of the four boundary surfaces is covered with the same number of the triangles and nodes. Suppose a single current dipole $\mathbf{P}=0\mathbf{i}+0\mathbf{j}+1\mathbf{k}$ located at (0,0,1) cm, table 1. gives the computational results under different amount of mesh triangles per compartment. With 808 triangles resulting in 406 nodes on each boundary surface, the influence of the tangential orientation sources with different depth but the same dipole moment is shown in table 2.

Table 1. Affect of the mesh density on numerical precision

Num- bers	196	248	296	354	420	488	556	636	716	808
RDM*	1.07	0.81	0.73	0.59	0.50	0.39	0.39	0.35	0.30	0.30

Table 2. Affect of the source depth on numerical precision

Location	0,0,0.2	0,0,0.3	0,0,0.4	0,0,0.5	0,0,0.6	0,0,0.7	0,0,0.8
RDM*	0.656	1.087	1.616	2.275	3.122	4.283	6.267

Based on the data shown in the tables, two conclusions can be deduced below:

One is that numerical errors can be reduced in a range with the increase of triangles on the surface, and the other is, the deeper the source is, the larger numerical error is made.

4 Conclusion

Several aspects in the numerical computing of the brain electric field are discussed. It includes realistic calculation head & brain models establishment which involved an

image segmentation, the mesh generation method, and an EEG forward problem computing with BEM. The simulation results have proved the methods' effectiveness.

Acknowledgement

This work was part supported by the Natural Science Foundation of Tianjin under Grant No.043800411, and the Science Foundation of Tianjin University of Technology under Grant No.LG04040.

References

1. Phillips, J.W., Leahy, R.M., Mosher, J.C., Timsari, B.: Imaging Neural Activity Using MEG and EEG. *IEEE Trans. on BME*, Vol. 16, No.3, (1997) 34 - 42
2. Q. Wu, W. L. Yan, X. Q. Shen, etc.: EEG Source Reconstruction Based on the Boundary-Element Method and Weighted Minimum Norm Approaches. *IEEE Trans. on Magn.*, Vol.39, No.3, (2003) 1547-1550
3. S. J. He, X. Q. Shen, Y. M. Yang, etc.: Research on MRI Brain Segmentation Algorithm With the Application in Model-based EEG/MEG. *IEEE Trans. on Magn.*, Vol.37, No.5, (2001) 3741-3744,.
4. M. Fuchs, R. Drenkhahn, H.A. Wischmann, and M. Wagner: An Improved Boundary Element Method for Realistic Volumeconductor Modeling. *IEEE Trans. BME*, Vol.45, No.8, (1998) 980-997
5. J. W. H. Meijs, O. W. Weier, etc.: On the Numerical Accuracy of the Boundary Element Method. *IEEE Trans on BME*. Vol.36, No.10, (1989) 1038~1049
6. M. Sun: An Efficient Algorithm for Computing Multishell Spherical Volume Conductor Models in EEG Dipole Source Localization. *IEEE Trans. on BME*, Vol.44, No.12, (1997) 1243-1252

A Novel Multi-stage 3D Medical Image Segmentation: Methodology and Validation

Jianfeng Xu¹, Lixu Gu^{1,2}, Xiahai Zhuang¹, and Terry Peters²

¹ Computer Science, Shanghai Jiaotong University,
1954 Huashan Road, Shanghai 200030, China
xujf@sjtu.edu.cn

² Robarts Research Institute, London, Ontario, Canada

Abstract. In this paper, we present a novel multi-stage algorithm for 3D medical image segmentation that is inspired by an improved Fast Marching method and a morphological reconstruction algorithm. The segmentation procedure consists of three steps: Connectivity Reduction, Hybrid segmentation, and Region recovery. The approach is tested on CT cardiac and MRI brain images, to demonstrate the effectiveness and accuracy of the technique. In order to validate this segmentation algorithm, a novel Radial Distance Based Validation (RDBV) method is proposed that provides a global accuracy (GA) measure. GA is calculated based on Local Radial Distance Errors (LRDE), where measured errors are along radii emitted from points along the skeleton of the object rather than the centroid, in order to accommodate more complicated organ structures. Using this GA measure, our results demonstrate that this multi-stage segmentation is fast and accurate, achieving approximately the same segmentation result as the watershed method, but with a processing speed of 3-5 times faster.

1 Introduction

Medical Segmentation plays a very important role in the field of computer-aided diagnosis and therapy planning, and has been the topic of much research focused on two main approaches: model-based and region-based methods, although these classifications are not mutually exclusive.

The Snake [1], introduced by Kass et al, provides a general model-based solution to the segmentation problem, while at the cost of higher computational complexity. Level Set [2] methods solve region breaking and merging problems. Much effort has focused on reducing the computational time for these algorithms, through the use of approaches such as the Fast Marching method [3]. However, model-based methods are sometimes limited in accuracy. In contrast to model-based methods, region-based methods can achieve more accurate results. Watershed [4] and Morphological Reconstruction [5] are two such methods, both derived from mathematical morphology. Since the methods are pixel-based, they are usually more computationally expensive.

In this paper, we present a multi-stage 3D medical image segmentation algorithm that integrates both the advantages of the Fast Marching method and a Morphological Reconstruction algorithm. To evaluate its performance, we propose a novel Radial Distance Based Validation (RDBV) method, which can quantify both local errors and

the global accuracy between a segmented result and a gold standard. Its advantages over standard techniques such as Similarity Index [6] are its universality and ability to reflect significant local errors in global accuracy function.

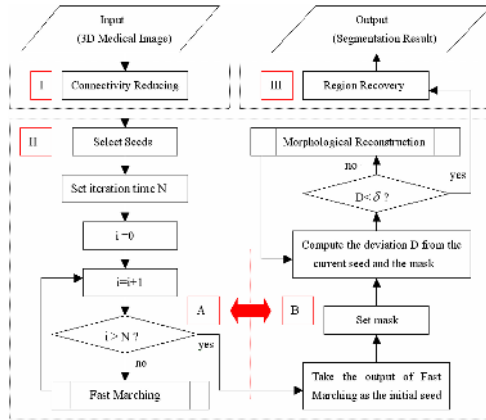


Fig. 1. Flowchart of the multi-stage segmentation method

The rest of the paper is organized as follows: in section 2, an improvement of the traditional Fast Marching method and 3D Morphological Reconstruction algorithm are presented. We describe the multi-stage segmentation strategy in section 3, and focus on the novel validation method RDBV in section 4.

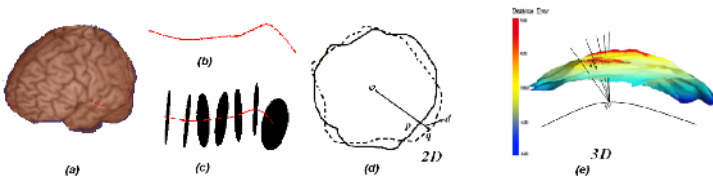


Fig. 2. (a) Origin 3D data. (b) The skeleton computed by distance mapping method. (c) Radius construction. (d) and (e) show the calculation of Local radial distance error (LRDE). Certain radial ϕq intersects with S and G resulting the intersection points q and p, respectively. d denotes the LRDE along ϕq . Black curve in (e) represents the skeleton of the object.

2 Improved Fast Marching and 3D Morphological Reconstruction

The Fast Marching (FM) algorithm is a special case of the level-set segmentation method. While the sign of the speed function in general in level-set approaches can be both positive and negative, that in the FM does not change. This restriction makes the FM much more rapid. However, sometimes the traditional FM method leads to overflow of the evolving front into neighboring regions, which is connected with the ROI. To prevent the front propagation from overflowing, we employ the improved FM described in [7] which introduced global information.

Mathematical morphology provides us with many powerful transformations for image analysis. It can efficiently classify particular shapes in an image and reconstruct defined regions via appropriate structuring elements (SE).

Morphological reconstruction (MR) is a typical approach to extract seeded regions, and is defined as:

$$\begin{cases} S_{i+1} = (S_i \oplus k) \cap |m| & (i = 0,1,2 \dots) \quad S_i - S_{i-1} > \delta \\ Stop & S_i - S_{i-1} \leq \delta \end{cases} \quad (1)$$

Where \oplus represents a dilation operation, and $|m|$ represents the mask reflecting the edge information in ROI, which is defined via a threshold operation using a histogram analysis. The mask is used to restrict the propagation within the ROI. S denotes the seed when k is the kernel size of dilation operation. δ is the deviation threshold value between $|m|$ and S_i . δ and k are both user defined. Since MR operates on individual pixels, it is sufficiently accurate to recover the ROI.

3 Multi-stage Segmentation Strategy

It is not practical to manually define an initial seed sufficiently close to the final boundary, and thus more computing effort is required in the MR. We introduce here an improved FM method to perform fast propagation from a user-defined seed to the position close to the final boundary, and this is employed as the seed during the MR phase. As shown in Fig.1, the procedure of the proposed multi-stage segmentation method includes three stages:

- I. Reduce the connectivity between the ROI and the neighboring tissues.
- II. Hybrid segmentation, where in Part A, after we perform the improved FM method to prepare a good initial seed, the MR is employed in Part B.
- III. Recover the lost data elements of ROI from stage I.

4 Validation Method

When a new segmentation method is proposed, it is crucial to provide an objective and quantitative analysis of its performance. We propose a novel validation method based on radial distances where the radii are extended from the points along the skeleton of a region (see Fig.2).

4.1 Skeletons and Radial Construction

Several skeleton algorithms, such as manual extraction, topological thinning, distance mapping [8] and so on, can be found in literature. Here, we employ the distance mapping method to construct the object's skeleton, as shown in Fig.2 (b).

From a point ϕ in the skeleton, we construct sufficient equi-spaced radii in the plane perpendicular to the skeleton as shown in Fig.2 (c). Each radial line intersects the surfaces of the segmented region S and the corresponding ground truth image G . The intersection points are q and p , respectively (see Fig.2 (d)-(e)). We define

$d = \varphi_q - \varphi_p$ as the local radial distance error (LRDE). If we apply the radial construction method to every point along the skeleton, we obtain LRDE everywhere.

4.2 Framework

Points along the skeleton are numbered in order as $\varphi_0, \varphi_1, \varphi_2 \dots \varphi_{N-1}$, where N is the number of points. Furthermore, radial lines emitted from the point φ_i are denoted $R_{i1}, R_{i2}, R_{i3} \dots R_{iM}$, where M is the number of radii emitted from φ_i , and the intersection points on the surfaces S and G are presented as $q_{i0}, q_{i1}, q_{i2} \dots q_{iM}$ and $p_{i0}, p_{i1}, p_{i2} \dots p_{iN}$, respectively. Thus, the LRDE can be formally defined as:

$$d_{ij} = \varphi_i q_{ij} - \varphi_i p_{ij} \quad (0 \leq i \leq N - 1, 0 \leq j \leq M - 1) \tag{2}$$

Furthermore, the intermediate variable c_{ij} is defined as below:

$$c_{ij} = \left| \frac{d_{ij}}{\varphi_i p_{ij}} \right| \quad (0 \leq i \leq N - 1, 0 \leq j \leq M - 1) \tag{3}$$

c_{ij} represents the fractional under- or over-segmentation along the direction of R_{ij} . We place c_{ij} in one of three categories:

- $0 < c_{ij} \leq 1/k$: Local segmentation is considered accurate;
- $1/k < c_{ij} < 2/k$: Local segmentation is acceptable;
- $c_{ij} \geq 2/k$: Local segmentation is unacceptable.

An experienced radiologist considered local segmentation could be considered acceptable for $c_{ij} \leq 0.05$, leading to the selection of a value of 20 for k as a constant defining an acceptable segmentation. We introduce the concept of Global Accuracy (GA) to reflect the accuracy of the global segmentation:

$$GA = N.M \left[\sum_{i,j=0}^{N-1, M-1} L_{i,j} \right]^{-1} \quad (0 \leq i \leq N - 1, 0 \leq j \leq M - 1) \tag{4}$$

Where, the target function $L_{i,j}$ is defined as:

$$L_{ij} = \lambda^{c_{ij}^k} \quad (0 \leq i \leq N - 1, 0 \leq j \leq M - 1) \tag{5}$$

And, $\lambda = 1.54$ [9].

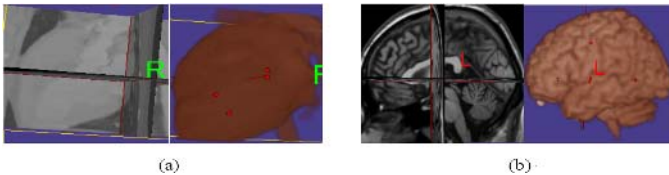


Fig. 3. Examples of segmentation results of heart (a) and brain (b) by the proposed method

Since $c_{ij} \leq 1/k$ is considered as an accurate segmentation, we can define GA_t , the threshold value of GA, as:

$$GA_i = N.M \left[\sum_{i,j=0}^{N-1,M-1} \lambda^{k \times \frac{1}{k}} \right]^{-1} = 0.65 \quad (0 \leq i \leq N - 1, 0 \leq j \leq M - 1) \quad (6)$$

So, $GA \geq GA_0 = 0.65$ is considered to be a good segmentation using the RDBV method.

5 Experimental Results

We performed our validation within the ‘‘TkSegmentation’’ software environment, which is based on the Visualization (VTK) and Insight (ITK) toolkits as well as the Python language. To perform the segmentation and validation, the software was implemented on a 2GHz Pentium-4 PC, running on Windows-XP.

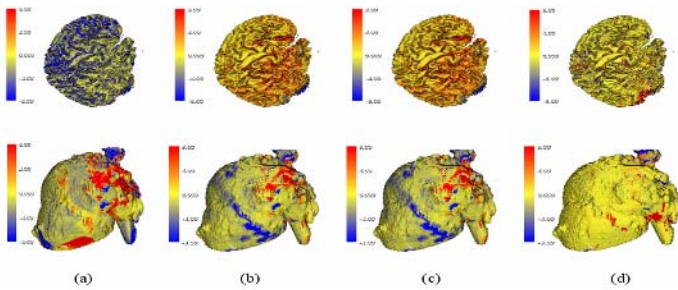


Fig. 4. LRDE distributions in the segmentation results of the FM (a), MR (b), PM (c) and WA (d) on brains and hearts

Table 1. Global Accuracy (GA) measure of the FM, MR, PM and WA on testing Datasets

Datasets	FM	MR	PM	WA
CJH27	0.70	0.82	0.82	0.88
Brain 1	0.68	0.83	0.83	0.88
Brain 2	0.71	0.84	0.84	0.89
Heart1	0.66	0.78	0.78	0.92
Heart2	0.69	0.79	0.79	0.97
Heart3	0.67	0.78	0.79	0.94
Average	0.69	0.81	0.81	0.91

Table 2. The computing time in seconds of FM, MR, PM and WA on testing Datasets

Datasets	FM	MR	PM	WA
CJH27	16.3	188.2	45.8	242.5
Brain 1	16.1	80.6	34.5	131.6
Brain 2	16.4	78.7	31.8	125.2
Heart1	66.9	442.1	120.8	562.6
Heart2	68.7	450.6	116.6	549.3
Heart3	67.4	466.2	137.0	558.7

The simulated datasets employed in our experiments were the standard CJH27 image volume for which “ground truth” is known. CJH27 is a $181 \times 217 \times 181$ voxel volume, with isotropic 1 mm^3 voxels. 2 MRI clinical brain volumes ($256 \times 256 \times 124$) and 3 3D CT canine heart images ($512 \times 512 \times 87$) were also tested. The “ground truth” of each real clinical dataset was manually defined by an expert radiologist. Examples of segmented results using the proposed approach are shown in Fig.3. The FM, MR, Watershed (WA) and our hybrid method (PM) are compared and evaluated with respect to both accuracy and efficiency using the RDBV metric. The results shown in Fig.4 and Table 1-2 reveal that this new multi-stage algorithm achieves superior performance with respect to the other approaches.

6 Conclusion

In this paper we presented a new multi-stage segmentation algorithm integrating the improved FM and MR approaches. Although our approach is still slower than the FM, it offers significantly improved accuracy. In addition, when compared to the MR and watershed methods, it maintained equivalent accuracy at a fraction of the computing cost. Meanwhile, the RDBV method is proposed to offer an improved measure of the segmentation results using both local and global information.

Acknowledgement

This work is partially supported by the Shanghai municipal research fund and the Ontario Consortium for Image-guided Therapy and Surgery, Canada. The authors want to thank Dr. Shixin Chang, director of the radiology department of Shanghai Oriental Hospital, for his assistance in providing testing data and clinical advices.

References

1. Kass, M., Witkin, A., and Terzopolous, D.: Snake: Active Contour Models. *Int. J. Compt. Vision*, Vol.1 (1988) 321-331
2. Osher, S., and Sethian, J.A.: Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulation. *J. of Comput. Phys.*, vol. 79 (1988) 12-49
3. Sethian, J.A.: Fast marching method. *SIAM Review*, 41(2) (1999) 199-235
4. Beucher, S., and Meyer, F.: The Morphological Approach to Segmentation: The Watershed Transform. *Mathematical Morphology in Image Processing*, E.R. Dougherty, Ed. New York: Marcel Dekker, vol.12 (1993) 433-481
5. Gu, L., and Kaneko, T.: Extraction of Organs Using 3D Mathematical Morphology. *Systems and Computers in Japan*, Vol.31-7 (2000) 29-37
6. Zijdenbos, A. P., Dawant, B. M., Margolin, R. A., Palmer, A. C.: Morphometric Analysis of White Matter Lesions in MR Images: Method and Validation. *IEEE Trans. Med. Imag.*, vol.13, (1994) 716-724
7. Yan, J. and Zhuang, T.: An improved fast marching method for detection of endocardial boundary in echocardiographic images. *Processing of SPIE.*, vol. 5032 (2003) 1292-1299
8. Wan, M., Liang, Z., Ke, Q., Hong, L., Bitter, I., Kaufman, A.: Automatic Centerline Extraction for Virtual Colonoscopy, *IEEE Trans. Med. Imag.*, vol.21(2002) 1450-1460
9. Xu, J., Gu, L., and Qi, W.: Presentation and Validation of an Accurate and Effective Segmentation for Dynamic Heart Modeling. *Processing of the 27th IEEE-EMBC (2005)*

Medical Image Alignment by Normal Vector Information

Xiahai Zhuang, Lixu Gu, and Jianfeng Xu

Computer Science, Shanghai Jiaotong University,
800 Dongchuan Road, Shanghai 200240, China
arhye@sjtu.edu.cn

Abstract. In this paper, a new approach on image registration is presented. We introduce a novel conception- normal vector information (NVI) - to evaluate the similarity between two images. NVI method takes advantage of the relationship between voxels in the image to extract the normal vector (NV) information of each voxel. Firstly, NVI criterion is presented. Then, based on the criterion, we find that NVI related metric has a quite perfect global optimal value on transformation parameter ranges. Finally, registration examples which are based on NVI criterion are provided. The result implies that the feature of smooth value distribution and one global optimal value that NVI metric has makes the optimization procedure much easier to be implemented in image registration.

1 Introduction

In image registration there is a need to find and evaluate the alignment of two images for other applications [1], [2]. Image registration is to find a geometric transformation that maps a given moving image into a fixed image [3]. Current popular metrics are mostly based on voxel intensity value of the images, such as mean squares metric, normalized correlation metric and mutual information metric. Those metrics employ the intensity value pair of the fixed image voxel and corresponding moving image voxel to evaluate their similarity in a special space transformation. Both mean square metric and normalized correlation metric work well in image registration but they are restricted to mono-modality applications¹. Mutual information metric can be applied in multi-modality medical image registration [4]; and much improvement has been achieved on it [5], [6]. However, the feature that the metric has many local maximum values in multi-modality registration makes the optimization difficult (fig.1 (a)). Optimization used in registration based on mutual information should be specially designed to achieve a good performance [7].

In this paper, we propose a new registration method based on NVI. Fig.1 (b) shows that the distribution of mean squares metric in mono-modality is very smooth and has excellent advantages in optimization. Here, we will present a novel criterion that employs the NVI of image to evaluate the similarity between two images. NVI is extracted from normal vector of each image voxel due to their isosurfaces or in image. This metric is not restricted in mono-modality. We will present both theory explanation and relative datasets to prove the feasibility of this method.

¹ The conclusion is drawn from experiments using source of Insight Toolkit (www.itk.org).

The rest of this paper is organized as follows: in section 2, we introduce NVI criterion use data analysis to show the advantages of NVI metric. In section 3, we offer medical image registration examples.

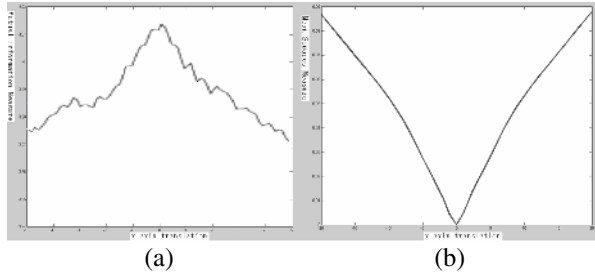


Fig. 1. Value of metrics to describe how similar two images are aligned plotted against x-axes translation. (a) mutual information metric. (b) mean squares metric.

2 NVI Based Similarity

2.1 Normal Vector Images

NV can be computed from the direction value of gradient of image [9]. Here, we give each NV component to each pigment of a RGB color to display a NV image.

The left two images of Fig.2 display two T1, T2 MRI images but they are similar in structure. In fact, they are from the some position of one brain but reflect different information. The NV value of each voxel of these two images is calculated and the result is that the NV images are almost the same, as Fig.2 (b) shows. The reason lies in that whatever the image want to reflect of the body function, the imaging result will catch the structure of imaged objects. Next, we will provide the definition of NVI metric and some algorithm used in registration process.

2.2 A Formal Definition of NVI Metric

To evaluate the alignment between two images, a direct idea is to calculate how much the NVI of the images are similar. Here we use cosine of the included angle of two NV value to obsess the similarity of two NV. Therefore, the sum of all the cosine value should be the function to represent the similarity of the two NVI images. A NVI metric evaluates alignment of images is presented as below:

$$S(p|F, M, T) = \sum_{i=1}^n |\cos(\theta(N_{F(x_i)}, N_{M(x'_i)}))|^e \tag{1}$$

Where, S means the similarity of the two images. Interpolator method is p, fixed image F, moving image M, transformation T. n is the number of voxels that have NV value and are counted. $N_{F(x_i)}$ is the normal vector of voxel xi in fix image. $N_{M(x'_i)}$ is that of move image. $X'_i = T(X_i)$. $\theta(N_{F(x_i)}, N_{M(x'_i)})$ is the included angle of the two vectors. $|\cos(\theta)|$ represents the absolute value of the included angle's cosine value. Parameter e stands of an exponent operator. As gradient descent optimizer is easiest

way to look for a maximum of a metric, and NV value could be proximately set as the normalized vector value of gradient, then formula used to compute the derivate of NVI metric S with respect to transformation T is offered here:

$$\partial S / \partial(T) = (\partial S / \partial(x_m, y_m, z_m)) / (\partial(x_m(T), y_m(T), z_m(T)) / \partial(T)) \tag{2}$$

$X'_i = T(X_i) = (x_m, y_m, z_m)$ After some manipulation $\partial S / \partial(x_m, y_m, z_m)$ might be written as follows,

$$\partial S / \partial(x_m, y_m, z_m) = \frac{2 * M_{fm}}{M_f^2 * M_m^4} * \left[\begin{array}{l} ((G_{fx} * \frac{\partial V_m^2}{\partial x \partial x} + G_{fy} * \frac{\partial V_m^2}{\partial y \partial x} + G_{fz} * \frac{\partial V_m^2}{\partial z \partial x}) * M_m^2 - M_{fm} * (G_{mx} * \frac{\partial V_m^2}{\partial x \partial x} + G_{my} * \frac{\partial V_m^2}{\partial y \partial x} + G_{mz} * \frac{\partial V_m^2}{\partial z \partial x})) \\ ((G_{fx} * \frac{\partial V_m^2}{\partial x \partial y} + G_{fy} * \frac{\partial V_m^2}{\partial y \partial y} + G_{fz} * \frac{\partial V_m^2}{\partial z \partial y}) * M_m^2 - M_{fm} * (G_{mx} * \frac{\partial V_m^2}{\partial x \partial y} + G_{my} * \frac{\partial V_m^2}{\partial y \partial y} + G_{mz} * \frac{\partial V_m^2}{\partial z \partial y})) \\ ((G_{fx} * \frac{\partial V_m^2}{\partial x \partial z} + G_{fy} * \frac{\partial V_m^2}{\partial y \partial z} + G_{fz} * \frac{\partial V_m^2}{\partial z \partial z}) * M_m^2 - M_{fm} * (G_{mx} * \frac{\partial V_m^2}{\partial x \partial z} + G_{my} * \frac{\partial V_m^2}{\partial y \partial z} + G_{mz} * \frac{\partial V_m^2}{\partial z \partial z})) \end{array} \right] \tag{3}$$

Where, G=(Gx,Gy,Gz) is the gradient, V is an image voxel. F means Fixed image, M means moving image. Mfm stands of the point multiply of gradient of fixed image and moving image. Mf=|Gf|, Mm=|Gm|

2.3 Advantages of NVI Metric

Distribution plots of NVI metric value against rigid transformation are provided to describe the advantages of NVI metric in biomedical registration.

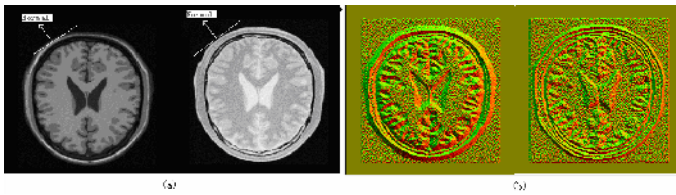


Fig. 2. T1, T2 MRI images² and their corresponding NVI images are presented. (a) Every location in the image has an isosurface and its normal vector. (b) NVI images of (a) images but does not include the “normal” and arrow.

We use CT-MRI images to show the distribution of the similarity value by rotating from -20 degree to 20 degree and x, y-axis each translating from -20 to 20 pixels. The metric formula used to calculate the similarity value is the formula (1) provided in section 2.2, and parameter e is set to 2. Fig.3 shows that the NVI metrics have only one maximum value; and all the curves or surfaces plotted against rigid transform parameters are quite smooth. We can expect that optimizers used in looking for an optimal value in NVI registration might be much easier and simpler to implement

² These two images are from Insight Toolkit example data. (www.itk.org)

because the perfect feature of the metric. Next section we will demonstrate the advantages using two registration experiments.

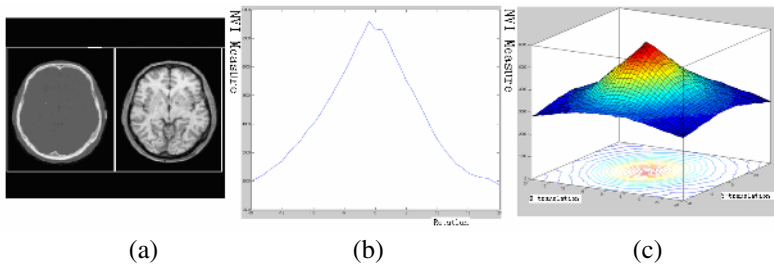


Fig. 3. The value distribution of CT-MRI images similarity by NVI metric against rigid transformation. (a) CT-MIR images both in size of 125*125 (b) value plotted against rotation (c) value plotted against x, y axes translation.

3 Registration Experiments

In this section we demonstrate medical image registration by NVI metric. Our experiments are based on MS Visual C++ environment and running on a P-IV 2.60 GHz PC, 1.0GB main memory, MS Windows XP.

In our experiments, we adopt bi-linear interpolation when needed and only consider rigid transformation. We use regular gradient descent as the optimizer. We adopt $\partial S/\partial(T) = (S(T_{+1}) - S(T_{-1})) / (T_{+1} - T_{-1})$ to approximately compute derivative of S against transformation parameters. This method makes our experiments much easier to implement and the learning rate of each parameter could be set almost the same. The NVI metric formula we provided in section 2.2 has the parameter value 2 which is our best experimental value. Here we offer T1 MRI to T2 MRI and CT to MRI registration experiments. The two MRI images employed in this experiment are shown in Fig.2 (a), which are both in size of 221*257. The CT-MRI images employed in this experiment are shown in Fig.3 (a), which are both in size of 125*125. The maximal iterations in both cases are set to 200 steps when the learning rate set to 0.01 for translation parameter, and 0.01 for rotation parameter in T1-T2 MRI example. In CT-MRI example, learning rate is 0.001 for both translation and rotation. The first step length of both examples is set to 5.

In each example, we performed a number of randomized experiments to determine the convergence, accuracy and efficiency. Mono-modality test data is reported in Table.1, multi-modality test data is reported in Table.2. In each example, the initial images are perfectly aligned. We use initial transform (X, Y, θ) to establish a misaligned pose. Each 4 experiments (E1, E2, E3, E4) in the two examples have 50 times registration.

- In E1, X, Y translation scope randomly varies from 0 pixel to 10 pixels, angle of rotation randomly varies from 0 to 10 degree;
- In E2, X, Y translation scope randomly varies from 0 pixel to 30 pixels, angle of rotation randomly varies from 0 to 10 degree;

- In E3, X, Y translation scope randomly varies from 0 pixel to 20 pixels, angle of rotation randomly varies from 0 to 20 degree;
- In E4, X, Y translation scope randomly varies from 10 pixel to 20 pixels, angle of rotation randomly varies from 20 to 40 degree.

Initial and Results columns in Table.1 and Table.2 are the average value of the 50 time registration initial value and result value. Column Time offers how many seconds the average test takes in these experiments. Column Steps presents average steps of optimization in these experiments. Column Suc in the two tables means how many times succeed in registration in the 50 registration.

Table 1. Registration results of mono-modality images (MRI-MRI)

E	Initial			Results			Time	Steps	Suc
	avgX	avgY	avg θ	avgX	avgY	avg θ			
E1	4.02	4.56	4.18	0.3372	0.3106	0.3140	10.0	6.7	50
E2	13.62	12.28	3.9	0.3036	0.2778	0.3222	9.8	6.7	48
E3	8.92	10.06	10.5	0.3682	0.2622	0.3602	10.04	6.9	50
E4	13.94	14.08	31.36	0.3308	0.3042	0.3200	8.52	5.82	50

Table 2. Registration results of multi-modality images (CT-MRI)

E	Initial			Results			Time	Steps	Suc
	avgX	avgY	avg θ	avgX	avgY	avg θ			
E1	4.64	4.44	5	0.5395	0.5873	0.3640	10.38	30.2	50
E2	16.76	14.38	4.92	0.5672	0.6289	0.3631	13.34	39.14	50
E3	9.18	9.36	8.8	0.5679	0.6160	0.3570	13.1	38.44	50
E4	14.44	14.68	30.48	0.5493	0.6609	0.3690	22.48	67.72	50

As shown in Table 1, the result of the experiments demonstrates that the alignment procedure is reliable, accurate and efficient in MRI-MRI medical images registration. In Table 2, the result of multi-modality registration is almost as good as that of mono-modality. The accuracy and robustness are nearly equal to the first example. Average time consumption and steps in each experiment are much more than that in mono-modality. The reason is the NVI that effectively used in similarity computing is less in multi-modality registration and the gradient of metric distribution is less steep than that of mono-modality.

4 Conclusion

In this paper, we presented a new conception to evaluate the similarity between images. Compared to other existing metrics, NVI metric has an excellent smooth value distribution plotted against transformation, and its optimal value is almost unique so that the optimization work can be easily implemented. Furthermore, NVI criterion is not restricted to mono-modality registration and has an outstanding performance in multi-modality domain as well.

Acknowledgements

The authors would like to thank all the members in the digital surgery laboratory of Shanghai Jiaotong University for their helps. We are also grateful to ITK members for their great work and enthusiastic help.

References

1. Gu, L.: Dynamic Heart Modeling Based on a Hybrid 3D segmentation Approach. In: Yang, G., Jiang, T. (eds.): *Medical Imaging and Augmented Reality. Lecture Note and Computer Science*, Vol. 3150. Springer (2004) 237-244
2. Xu, J., Gu, L.: Evaluation of Morphological Reconstruction, Fast Marching and a New Hybrid Segmentation. In: Zhang, J., He, J. (eds.): *Computational and Information Science. Lecture Notes in Computer Science*, Vol. 3314 (2004) 678-684
3. Brown, L.G.: A Survey of Image Registration Techniques. *ACM Comput. Surv.*, Vol. 24 (1992) 325-376
4. Viola, P., Wells III, W.: Alignment by Maximization of Mutual Information. *International Journal of Computer Vision*, Vol. 24(2) (1997) 137-154
5. Mattes, D., Haynor, D.R., Vesselle, H., Lewellen, T., Eubank, W.: Nonrigid multimodality image registration. In: Sonka, M., Hanson, K.M. (eds): *Medical Imaging 2001: Image Processing*, Vol. 4322. SPIE Press, Bellingham, WA (2001) 1609-1620
6. Mattes, D., Haynor, D.R., Vesselle, H., Lewellen, T., Eubank, W.: PET-CT Image Registration in the Chest Using Free-form Deformations. *IEEE Transactions in Medical Imaging*, Vol. 22, No.1 (2003) 120-128
7. Thevenaz, P., Unser, M.: Optimization of Mutual Information for MultiResolution Image Registration. *IEEE Transactions in Image Processing*, Vol. 9, No. 12 (2000) 2083-2099
8. Ng, L.: Overview: ITK Registration Methods. *SPIE2004: Medical Image Segmentation and Registration With ITK* (2004)
9. Hoehne, K.H., Bernstein, R.: Shading 3D-Images from CT Using Gray-Level Gradients. *IEEE Trans. Medical Imaging*, Vol. MI-5, No. 1 (1986) 45-47

Global Exponential Stability of Non-autonomous Delayed Neural Networks*

Qiang Zhang^{1,2}, Dongsheng Zhou², and Xiaopeng Wei¹

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

² School of Mechanical Engineering,
Dalian University of Technology, Dalian 116024, China
zhangq30@hotmail.com

Abstract. Global exponential stability of non-autonomous delayed neural networks is discussed. A new sufficient condition ensuring the exponential stability for this type of neural networks by constructing a new suitable Lyapunov functionals. Since the condition does not impose boundedness on activation functions, it is less restrictive than some established in the earlier literature.

1 Introduction

The dynamics of autonomous delayed neural networks has attracted the attention of many researchers in the past decades and many results on the global asymptotic stability and global exponential stability have been presented, see, for instance, [1]-[12] and references cited therein. However, up to now, few studies have considered stability for non-autonomous delayed neural networks [13]. In this paper, by making use of Lyapunov functional approach, we discuss the global exponential stability of non-autonomous delayed neural networks and obtain a new sufficient condition which guarantees global exponential stability for the non-autonomous delayed neural networks. We will see that the obtained result improves some results on the global exponential stability for the autonomous and non-autonomous delayed neural networks given in the previous literature.

2 Preliminaries

The dynamic behavior of a continuous time non-autonomous delayed neural networks can be described by the following state equations:

$$\begin{aligned}x'_i(t) = & -c_i(t)x_i(t) + \sum_{j=1}^n a_{ij}(t)f_j(x_j(t)) \\ & + \sum_{j=1}^n b_{ij}(t)f_j(x_j(t - \tau_j(t))) + I_i(t).\end{aligned}\tag{1}$$

* The project supported by the National Natural Science Foundation of China (grant no. 60403001), LNET and Dalian Youth Foundation.

where n corresponds to the number of units in a neural networks; $x_i(t)$ corresponds to the state vector at time t ; $f(x(t)) = [f_1(x_1(t)), \dots, f_n(x_n(t))]^T \in R^n$ denotes the activation function of the neurons; $A(t) = [a_{ij}(t)]_{n \times n}$ is referred to as the feedback matrix, $B(t) = [b_{ij}(t)]_{n \times n}$ represents the delayed feedback matrix, while $I_i(t)$ is a external bias vector at time t , $\tau_j(t)$ is the transmission delay along the axon of the j th unit and is non-negative, bounded and continuously differentiable satisfying $0 \leq \tau_j(t) \leq \tau$ and $\tau'_j(t) \leq \tau_j^* < 1$.

Throughout this paper, we will assume that the real valued functions $c_i(t) > 0$, $a_{ij}(t), b_{ij}(t), I_i(t)$ are continuous functions. The activation functions $f_i, i = 1, 2, \dots, n$ are assumed to satisfy the following hypothesis

$$|f_i(\xi_1) - f_i(\xi_2)| \leq L_i |\xi_1 - \xi_2|, \forall \xi_1, \xi_2. \tag{2}$$

This type of activation functions is clearly more general than both the usual sigmoid activation functions and the piecewise linear function (PWL): $f_i(x) = \frac{1}{2}(|x + 1| - |x - 1|)$ which is used in [14].

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

$$x_i(s) = \phi_i(s), \quad s \in [-\tau, 0].$$

in which $\phi_i(s)$ are continuous for $s \in [-\tau, 0]$.

We say that system (1) is globally exponentially stable if there are constants $\eta > 0$ and $M \geq 1$ such that for any two solutions $x(t)$ and $z(t)$ with the initial functions $x(s) = \phi(s)$ and $z(s) = \varphi(s)$ for $s \in [-\tau, 0]$, respectively, one has

$$\|x(t) - z(t)\| \leq M \|\phi - \varphi\| \exp(-\eta t), \text{ for all } t \geq 0.$$

where $\|\phi - \varphi\| = \sup_{-\tau \leq s \leq 0} [\sum_{i=1}^n |\phi_i(s) - \varphi_i(s)|^3]^{1/3}$.

3 Global Exponential Stability Analysis

In this section, we will establish the global exponential stability of system (1). Consider two solutions $x(t)$ and $z(t)$ of system (1) for $t > 0$ corresponding to arbitrary initial values $x(s) = \phi(s)$ and $z(s) = \varphi(s)$ for $s \in [-\tau, 0]$. Let $y_i(t) = x_i(t) - z_i(t)$, then we have

$$\begin{aligned} y'_i(t) = & -c_i(t)y_i(t) + \sum_{j=1}^n a_{ij}(t) (f_j(x_j(t)) - f_j(z_j(t))) \\ & + \sum_{j=1}^n b_{ij}(t) (f_j(x_j(t - \tau_j(t))) - f_j(z_j(t - \tau_j(t)))) \end{aligned} \tag{3}$$

Set $g_j(y_j(t)) = f_j(y_j(t) + z_j(t)) - f_j(z_j(t))$, one can rewrite Eq.(3) as

$$y'_i(t) = -c_i(t)y_i(t) + \sum_{j=1}^n a_{ij}(t)g_j(y_j(t)) + \sum_{j=1}^n b_{ij}(t)g_j(y_j(t - \tau_j(t))) \tag{4}$$

Note that the functions f_j satisfy the hypothesis (2), that is,

$$|g_i(\xi_1) - g_i(\xi_2)| \leq L_i |\xi_1 - \xi_2|, \forall \xi_1, \xi_2. \tag{5}$$

$$g_i(0) = 0$$

By Eq.(4), we have

$$D^+ |y_i(t)| \leq -c_i(t) |y_i(t)| + \sum_{j=1}^n |a_{ij}(t)| L_j |y_j(t)| + \sum_{j=1}^n |b_{ij}(t)| L_j |y_j(t - \tau_j(t))| \tag{6}$$

Theorem 1. *Eq.(1) is globally exponentially stable if there exist real constants $\alpha_{ij}, \alpha_{ij}^*, \beta_{ij}, \beta_{ij}^*, \zeta_{ij}, \zeta_{ij}^*, \eta_{ij}, \eta_{ij}^*$ and positive constants $\gamma_i > 0$ such that*

$$\begin{aligned} & \sum_{j=1}^n \left\{ |a_{ij}(t)|^{3\alpha_{ij}} L_j^{3\beta_{ij}} + |a_{ij}(t)|^{3\alpha_{ij}^*} L_j^{3\beta_{ij}^*} + |b_{ij}(t)|^{3\zeta_{ij}} L_j^{3\eta_{ij}} + |b_{ij}(t)|^{3\zeta_{ij}^*} L_j^{3\eta_{ij}^*} \right\} \\ & + \sum_{j=1}^n \frac{\gamma_j}{\gamma_i} \left\{ |a_{ji}(t)|^{3-3(\alpha_{ji} + \alpha_{ji}^*)} L_i^{3-3(\beta_{ji} + \beta_{ji}^*)} + \frac{|b_{ji}(t)|^{3-3(\zeta_{ji} + \zeta_{ji}^*)} L_i^{3-3(\eta_{ji} + \eta_{ji}^*)}}{1 - \tau_i^*} \right\} \\ & < 3c_i(t) \end{aligned} \tag{7}$$

Proof. Since the inequality (7) holds, we can choose a small $\varepsilon > 0$ such that

$$\begin{aligned} & \varepsilon - 3c_i(t) + \sum_{j=1}^n \left\{ |a_{ij}(t)|^{3\alpha_{ij}} L_j^{3\beta_{ij}} + |a_{ij}(t)|^{3\alpha_{ij}^*} L_j^{3\beta_{ij}^*} \right. \\ & \left. + |b_{ij}(t)|^{3\zeta_{ij}} L_j^{3\eta_{ij}} + |b_{ij}(t)|^{3\zeta_{ij}^*} L_j^{3\eta_{ij}^*} \right\} \\ & + \sum_{j=1}^n \frac{\gamma_j}{\gamma_i} \left[|a_{ji}(t)|^{3-3(\alpha_{ji} + \alpha_{ji}^*)} L_i^{3-3(\beta_{ji} + \beta_{ji}^*)} \right. \\ & \left. + \frac{|b_{ji}(t)|^{3-3(\zeta_{ji} + \zeta_{ji}^*)} L_i^{3-3(\eta_{ji} + \eta_{ji}^*)} e^{\varepsilon\tau}}{1 - \tau_i^*} \right] \\ & < 0 \end{aligned}$$

To obtain the result, the following positive definite Lyapunov functional will be used:

$$\begin{aligned} z(t) &= \frac{1}{3} \sum_{i=1}^n \gamma_i |y_i(t)|^3 e^{\varepsilon t} \\ &+ \frac{1}{3} \sum_{i=1}^n \sum_{j=1}^n \gamma_i \frac{|b_{ij}(t)|^{3-3(\zeta_{ij} + \zeta_{ij}^*)} L_j^{3-3(\eta_{ij} + \eta_{ij}^*)}}{1 - \tau_j^*} \int_{t-\tau_j(t)}^t |y_j(s)|^3 e^{\varepsilon(s+\tau)} ds \end{aligned} \tag{8}$$

Calculating the upper right derivative of $z(t)$ along the solutions of Eq.(4), we have

$$\begin{aligned}
 D^+z(t) &\leq \sum_{i=1}^n \gamma_i |y_i(t)|^2 D^+|y_i(t)|e^{\epsilon t} + \frac{1}{3} \sum_{i=1}^n \epsilon \gamma_i |y_i(t)|^3 e^{\epsilon t} \\
 &\quad + \frac{1}{3} \sum_{i=1}^n \sum_{j=1}^n \gamma_i \frac{|b_{ij}(t)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)} L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}}{1-\tau_j^*} |y_j(t)|^3 e^{\epsilon(t+\tau)} \\
 &\quad - \frac{1}{3} \sum_{i=1}^n \sum_{j=1}^n \gamma_i \frac{e^{\epsilon t} |b_{ij}(t)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)} L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}}{1-\tau_j^*} \\
 &\quad \times |y_j(t-\tau_j(t))|^3 (1-\tau_j'(t)) \\
 &\leq \sum_{i=1}^n \gamma_i e^{\epsilon t} |y_i(t)|^2 \left\{ -c_i(t) |y_i(t)| + \sum_{j=1}^n |a_{ij}(t)| L_j |y_j(t)| \right. \\
 &\quad \left. + \sum_{j=1}^n |b_{ij}(t)| L_j |y_j(t-\tau_j(t))| \right\} + \frac{e^{\epsilon t}}{3} \sum_{i=1}^n \epsilon \gamma_i |y_i(t)|^3 \\
 &\quad + \frac{e^{\epsilon(t+\tau)}}{3} \sum_{i=1}^n \sum_{j=1}^n \gamma_i \frac{|b_{ij}(t)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)} L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}}{1-\tau_j^*} |y_j(t)|^3 \\
 &\quad - \frac{1}{3} \sum_{i=1}^n \sum_{j=1}^n \gamma_i e^{\epsilon t} |b_{ij}(t)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)} L_j^{3-3(\eta_{ij}+\eta_{ij}^*)} |y_j(t-\tau_j(t))|^3 \\
 &= - \sum_{i=1}^n \gamma_i c_i(t) |y_i(t)|^3 e^{\epsilon t} + \sum_{i=1}^n \sum_{j=1}^n \gamma_i |a_{ij}(t)| L_j |y_i(t)|^2 |y_j(t)| e^{\epsilon t} \\
 &\quad + \sum_{i=1}^n \sum_{j=1}^n \gamma_i |b_{ij}(t)| L_j |y_i(t)|^2 |y_j(t-\tau_j(t))| e^{\epsilon t} + \frac{e^{\epsilon t}}{3} \sum_{i=1}^n \epsilon \gamma_i |y_i(t)|^3 \\
 &\quad + \frac{e^{\epsilon(t+\tau)}}{3} \sum_{i=1}^n \sum_{j=1}^n \gamma_i \frac{|b_{ij}(t)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)} L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}}{1-\tau_j^*} |y_j(t)|^3 \\
 &\quad - \frac{1}{3} \sum_{i=1}^n \sum_{j=1}^n \gamma_i e^{\epsilon t} |b_{ij}(t)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)} L_j^{3-3(\eta_{ij}+\eta_{ij}^*)} |y_j(t-\tau_j(t))|^3 \quad (9)
 \end{aligned}$$

Since

$$\begin{aligned}
 |a_{ij}(t)| L_j |y_i(t)|^2 |y_j(t)| &= \left(|a_{ij}(t)|^{\alpha_{ij}} L_j^{\beta_{ij}} |y_i(t)| \right) \left(|a_{ij}(t)|^{\alpha_{ij}^*} L_j^{\beta_{ij}^*} |y_i(t)| \right) \\
 &\quad \times \left(|a_{ij}(t)|^{1-(\alpha_{ij}+\alpha_{ij}^*)} L_j^{1-(\beta_{ij}+\beta_{ij}^*)} |y_j(t)| \right) \quad (10)
 \end{aligned}$$

Employing the element inequality $3abc \leq a^3 + b^3 + c^3$ ($a, b, c \geq 0$) in (10), we get

$$|a_{ij}(t)| L_j |y_i(t)|^2 |y_j(t)| \leq \frac{1}{3} \left(|a_{ij}(t)|^{3\alpha_{ij}} L_j^{3\beta_{ij}} + |a_{ij}(t)|^{3\alpha_{ij}^*} L_j^{3\beta_{ij}^*} \right) |y_i(t)|^3$$

$$+\frac{1}{3}|a_{ij}(t)|^{3-3(\alpha_{ij}+\alpha_{ij}^*)}L_j^{3-3(\beta_{ij}+\beta_{ij}^*)}|y_j(t)|^3 \tag{11}$$

Similarly, we have

$$\begin{aligned} |b_{ij}(t)|L_j|y_i(t)|^2|y_j(t-\tau_j(t))| &\leq \frac{1}{3}\left(|b_{ij}(t)|^{3\zeta_{ij}}L_j^{3\eta_{ij}} + |b_{ij}(t)|^{3\zeta_{ij}^*}L_j^{3\eta_{ij}^*}\right)|y_i(t)|^3 \\ &\quad + \frac{1}{3}|b_{ij}(t)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)}L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}|y_j(t-\tau_j(t))|^3 \end{aligned} \tag{12}$$

Substituting (11) and (12) into (9), we get

$$\begin{aligned} D^+z(t) &\leq \frac{1}{3}\sum_{i=1}^n\gamma_i e^{\epsilon t}\left\{\epsilon - 3c_i(t) + \sum_{j=1}^n\left\{|a_{ij}(t)|^{3\alpha_{ij}}L_j^{3\beta_{ij}} + |a_{ij}(t)|^{3\alpha_{ij}^*}L_j^{3\beta_{ij}^*}\right. \right. \\ &\quad \left. \left. + |b_{ij}(t)|^{3\zeta_{ij}}L_j^{3\eta_{ij}} + |b_{ij}(t)|^{3\zeta_{ij}^*}L_j^{3\eta_{ij}^*}\right\} \right. \\ &\quad \left. + \sum_{j=1}^n\frac{\gamma_j}{\gamma_i}\left[|a_{ji}(t)|^{3-3(\alpha_{ji}+\alpha_{ji}^*)}L_i^{3-3(\beta_{ji}+\beta_{ji}^*)} \right. \right. \\ &\quad \left. \left. + \frac{|b_{ji}(t)|^{3-3(\zeta_{ji}+\zeta_{ji}^*)}L_i^{3-3(\eta_{ji}+\eta_{ji}^*)}e^{\epsilon\tau}}{1-\tau_i^*}\right]\right\}|y_i(t)|^3 \\ &< 0 \end{aligned}$$

for all $t \geq 0$. Therefore, we have

$$z(t) \leq z(0) \tag{13}$$

It follows from Eq.(8) that

$$\begin{aligned} z(t) &\geq \frac{1}{3}\sum_{i=1}^n\gamma_i|y_i(t)|^3e^{\epsilon t} \\ z(0) &= \frac{1}{3}\sum_{i=1}^n\gamma_i|y_i(0)|^3 \\ &\quad + \frac{1}{3}\sum_{i=1}^n\sum_{j=1}^n\gamma_i\frac{|b_{ij}(0)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)}L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}}{1-\tau_j^*}\int_{-\tau_j(0)}^0|y_j(s)|^3e^{\epsilon(s+\tau)}ds \\ &\leq \frac{1}{3}\sum_{i=1}^n\gamma_i\left\{\sup_{s\in[-\tau,0]}(\phi_i(s)-\varphi_i(s))^3 \right. \\ &\quad \left. + \sum_{j=1}^n\frac{|b_{ij}(0)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)}L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}\tau e^{\epsilon\tau}}{1-\tau_j^*}\sup_{s\in[-\tau,0]}(\phi_i(s)-\varphi_i(s))^3\right\} \\ &\leq \frac{1}{3}\sum_{i=1}^n\gamma_i\left\{1 + \sum_{j=1}^n\frac{|b_{ij}(0)|^{3-3(\zeta_{ij}+\zeta_{ij}^*)}L_j^{3-3(\eta_{ij}+\eta_{ij}^*)}\tau e^{\epsilon\tau}}{1-\tau_j^*}\right\} \end{aligned}$$

$$\times \sup_{s \in [-\tau, 0]} (\phi_i(s) - \varphi_i(s))^3 \tag{14}$$

Hence, from (13) and (14) we obtain

$$\sum_{i=1}^n |y_i(t)|^3 \leq M \|\phi_i(s) - \varphi_i(s)\|^3 e^{-\epsilon t}$$

where $M = \frac{\gamma_{\max}}{\gamma_{\min}} \max_i \left\{ 1 + \sum_{j=1}^n \frac{|b_{ij}(0)|^{3-3(\zeta_{ij} + \zeta_{ij}^*)} L_j^{3-3(\eta_{ij} + \eta_{ij}^*)} \tau e^{\epsilon \tau}}{1 - \tau_j^*} \right\} \geq 1$. This com-

pletes the proof.

Remark 1. Note that the result obtained here is independent of delay and the activation functions $f_i(t)$ may be unbounded. For this reason, our result improves and extends those earlier established in [1]-[6], [12].

References

1. Arik, S.: An Improved Global Stability Result for Delayed Cellular Neural Networks. *IEEE Trans.Circuits Syst.I* **49** (2002) 1211–1214
2. Cao, J., Wang, J.: Global Asymptotic Stability of a General Class of Recurrent Neural Networks with Time-Varying Delays. *IEEE Trans.Circuits Syst.I* **50** (2003) 34–44
3. Huang, H., Cao, J.: On Global Asymptotic Stability of Recurrent Neural Networks with Time-Varying Delays. *Appl.Math.Comput.* **142** (2003) 143–154
4. Liao, X., Chen, G., Sanchez, E.N.: LMI-Based Approach for Asymptotically Stability Analysis of Delayed Neural Networks. *IEEE Trans.Circuits Syst.I* **49** (2002) 1033–1039
5. Liao, X.X., Wang, J.: Algebraic Criteria for Global Exponential Stability of Cellular Neural Networks with Multiple Time Delays. *IEEE Trans.Circuits Syst.I* **50** (2003) 268–274
6. Mohamad, S., Gopalsamy, K.: Exponential Stability of Continuous-Time and Discrete-Time Cellular Neural Networks with Delays. *Appl.Math.Comput.* **135** (2003) 17–38
7. Zeng, Z., Wang, J., Liao, X.: Global Exponential Stability of a General Class of Recurrent Neural Networks with Time-Varying Delays. *IEEE Trans.Circuits Syst.I* **50** (2003) 1353–1358
8. Zhang, J.: Globally Exponential Stability of Neural Networks with Variable Delays. *IEEE Trans.Circuits Syst.I* **50** (2003) 288–290
9. Zhang, Q., Ma, R., Xu, J.: Stability of Cellular Neural Networks with Delay. *Electron. Lett.* **37** (2001) 575–576
10. Zhang, Q., Ma, R., Chao, W., Jin, X.: On the Global Stability of Delayed Neural Networks. *IEEE Trans.Automatic Control* **48** (2003) 794–797
11. Zhang, Q., Wei, X.P. Xu, J.: Global Exponential Convergence Analysis of Delayed Neural Networks with Time-Varying Delays. *Phys.Lett.A* **318** (2003) 537–544
12. Zhou, D., Cao, J.: Globally Exponential Stability Conditions for Cellular Neural Networks with Time-Varying Delays. *Appl.Math.Comput.* **131** (2002) 487–496
13. Jiang, H., Li, Z., Teng, Z.: Boundedness and Stability for Nonautonomous Cellular Neural Networks with Delay. *Phys.Lett.A* **306** 313–325
14. Chua, L.O., Yang, L.: Cellular Neural Networks: Theory and Applications. *IEEE Trans.Circuits Syst.I* **35** (1988) 1257–1290

A Prediction Method for Time Series Based on Wavelet Neural Networks

Xiaobing Gan¹, Ying Liu², and Francis R. Austin³

¹ College of Management, Shenzhen University, Shenzhen 518060, China
ganxb5280@163.com

² China Academy of Urban Planning & Design, Shenzhen 518034, China

³ College of Science, Shenzhen University, Shenzhen 518060, China

Abstract. This paper introduces a prediction method for time series that is based on the multi-resolution analysis of wavelets (MRA). The MRA is better able to decompose the non-stationary time series of nonlinear systems into different components, allowing a better separation of the general trend terms, the periodic terms and the random fluctuation terms. By applying the most suitable prediction methods (for example, the neural networks method, cosine approximation, or the ARMA model) to the components under different resolutions, this new prediction method produces more accurate prediction results. The new approach is then applied to a real example – the BRENT oil price time series – to demonstrate its usefulness and validity.

1 Introduction

Traditional prediction methods for time series include regression analysis, exponential smoothing, and ARMA. These methods generally produce reasonable prediction results for stationary random time series of linear systems. While the Holt-Winters method and others based on neural networks are able to predict nonlinear systems, they generally fail for systems with strong perturbation. Murtagh and others [1] demonstrated the superiority of multi-resolution analysis methods over their single-resolution counterparts by applying multi-resolution analysis models of wavelet transforms to predict the values of financial futures and stock market prices. Anesti et al [2] proposed a prediction method for time series that combined wavelet methods with Hilbert's auto-regression random process and made a comparison between their method and traditional prediction methods. Dongjie Wu et al [3] used a combination of neural networks and wavelet methods to predict underground water levels. Similarly, using the concepts of wavelet analysis, Qiang Liang and others [4] predicted the long-term general trend of oil prices by applying cosine approximation to those components of the series that possessed a regular fluctuating pattern. Based on the multi-level decomposition of wavelet theory, this paper introduces a prediction method that is better able to separate the general trend terms, the periodic terms and the random terms of non-stationary time series. The general trend terms usually correspond to the non-stationary nonlinear time series that is obtained after the original series is smoothed under a certain resolution. This paper uses a neural networks method to predict the general trend terms, and either the cosine approximation or the ARMA model to predict the terms under other resolutions.

2 A Prediction Model Based on Wavelet Neural Networks

2.1 Wavelet Decomposition and Reconstruction Model

Let $x(t) = \{x_1, x_2, \dots, x_N\}$ be a time series, denoted by c_0 . With respect to any orthonormal wavelet basis, apply the Mallat algorithm to the original series to obtain the wavelet decomposition

$$\begin{cases} c_{j+1} = Hc_j \\ d_{j+1} = Gc_j \end{cases} \quad j = 0, 1, 2, \dots, J . \quad (1)$$

Equation (1) provides a recurrence relation for obtaining the decomposition c_{j+1} and d_{j+1} of c_j , where H and G are known respectively in the literature as the low-frequency-pass filter and the $\bar{\pi}$ bandpass filter. The Mallat algorithm, in effect, decomposes the time series according to its different frequency channel components. After C_0 is decomposed into $\{d_1, d_2, \dots, d_J, c_J\}$, the length of the detail d_{j+1} (which corresponds to the data in the high frequency section) and the approximation c_{j+1} (which corresponds to the data in the low frequency section) are both only half that of c_j , the length before decomposition.

If the coefficients after decomposition are known, then the original coefficients can be determined by using the equation

$$c_j = H^* c_{j+1} + G^* d_{j+1} \quad j = J - 1, J - 2, \dots, 1, 0 \quad (2)$$

where H^* and G^* are respectively the complex conjugate of the transpose of the matrices H and G . In view of the above discussion, the length of each reconstructed series for C_j is two times greater than that before reconstruction and hence of the same length as the original time series. Let $D_1, D_2, \dots, D_J, C_J$ be these reconstructed series, then

$$x(t) = D_1 + D_2 + \dots + D_J + C_J . \quad (3)$$

The concept of dynamic series prediction based on wavelet neural networks is as follows. First, predict a value for the approximation C_j in equation (3) using the BP neural networks model. Then, by observing the particular properties of each D_1, D_2, \dots, D_J , predict their values using the cosine approximation method. Finally, combine the above results to obtain a dynamic series prediction.

2.2 A Simulated Prediction Method for Approximating C_j : the BP Neural Networks Model

In general, an approximated time series is obtained by smoothly extending an original time series under a certain power of resolution. The series so obtained is usually nonlinear and non-stationary. Neural networks are nonlinear dynamical systems that

are made up of numerous parallel processes each one of which consists of a single unit that performs a simple function. Neural networks have applications in fields such as function approximations, pattern classification, and optimization. This paper presents the results of a simulated prediction of the approximation C_j in equation (3) using a three-level BP neural networks model.

Choose the nonlinear smooth function $g(x) = \frac{1}{1 + e^{-x}}$ in the BP neural networks model. Then the input-output equation for the j -th neural cell in the middle level is

$$u_j = g\left(\sum_{i=1}^M w_{ij}^1 X_i - \theta_j\right), \tag{4}$$

and that for the k -th neural cell in the output level is

$$\hat{Y}_k = g\left(\sum_{j=1}^L w_{jk}^2 u_j - \delta_k\right), \tag{5}$$

where X_i are the input values, \hat{Y}_k are estimates for the corresponding output values Y_k , M, L are respectively the number of neural cells in the middle and output levels, w_{ij}^1 and w_{jk}^2 are respectively the weighted coefficient matrices of the input data at the middle and output levels, and θ_j and δ_k are respectively the neural cell threshold at the middle and output levels.

After evaluating \hat{Y}_k for each group of input-output data, adjust the weights and the thresholds by using the negative error gradient of Y_k so that the output error is monotone decreasing and stability is achieved. Apply the above procedure repeatedly to n groups of data until the weighted coefficient matrices and threshold matrices satisfy all the n input-output relations. These weighted coefficient and threshold matrices can then be used for prediction.

Divide the time series into Ni segments. Then since data points that are closer to the present time are more reliable than those that are far away, the last data point X_{N_i} in each segment is chosen as the expected output value and $\{X_{N_i-r} / r, r = 1, 2, \dots\}$ is used for evaluating a prediction of the initial weighted value.

2.3 A Prediction Method for Details D_1, D_2, \dots, D_j : Cosine Wave Approximation

In effect, a detail signal series represents the magnitudes of the fluctuations of the original dynamic series (i.e. the original signal) in some frequency channel. A cosine wave is used in this paper to approximate the periodic terms.

Let ξ_t be the time series to be cosine wave approximated, q be the total number of wave crests and wave troughs, and t_i ($1 \leq i \leq q$) be the position of each wave crest (or trough). Then the mean period and mean amplitude of the wave ξ_t are given respectively by

$$T = \frac{2 \sum_{i=1}^{q-1} (t_{i+1} - t_i)}{q - 1}, \tag{6}$$

and

$$A = \frac{\sum_{i=1}^{q-1} |\xi_{t_{i+1}} - \xi_{t_i}|}{2(p - 1)}. \tag{7}$$

In order to choose a suitable level of details for cosine wave prediction, it is necessary to measure the extent of the fluctuations of parts of the details. Here, the measure of the degree of fluctuations used is μ [4], where

$$\mu = \frac{1}{sT} \sum_{i=1}^s \left| t_i - t_1 - \frac{(i-1)T}{2} \right|, \quad 1 \leq i \leq s. \tag{8}$$

It is possible to choose a small constant μ_0 such that when

$$\mu < \mu_0, \tag{9}$$

the fluctuations of D_k may be roughly regarded as regular so that D_k can be well approximated by a suitable cosine wave. Let T_k and A_k be the period and mean amplitude of the predicted cosine wave of D_k respectively, and let the final extreme point of the time series be a wave trough with Cartesian coordinates (a_k, b_k) . Then the equation for determining the predicted value at time t can be written as

$$\hat{D}_k(t) = b_k + A_k - A_k \cos \left[\frac{2\pi}{T} (t - a_k) \right]. \tag{10}$$

Similarly, if the final extreme point of the time series is a wave crest with coordinates (a_k, b_k) , then the equation for prediction is

$$\hat{D}_k(t) = b_k - A_k + A_k \cos \left[\frac{2\pi}{T} (t - a_k) \right]. \tag{11}$$

In this way, the future values of D_k can be predicted using either equation (10) or (11).

Finally, for details that do not satisfy the condition given in (9), other prediction methods are available that may depend on the particular properties of these details. For instance, components with low frequency or details with large amplitudes can be viewed as stationary processes and so are amenable to prediction by AR, MA or

ARMA models. Furthermore, details in the lowest scale of the wavelet decomposition can be viewed as white noise and generally neglected in prediction calculations without creating major errors.

2.4 Combining the Predictions

The final predicted value of C_J is obtained by combining the predicted values of its approximation and the details. Indeed, a simulated prediction model for dynamic time series using wavelet decomposition is obtained by superposing all the simulated prediction models discussed above. This model much better reflects the general trend and fluctuation patterns of the dynamic series under various time scales and so is able to produce prediction results that more closely resemble both the long-term trend as well as the short-term movements of the actual series. More precisely, the prediction results are combined as follows:

Let $x(t) = \{x_1, x_2, \dots, x_N\}$ be a non-stationary time series; determine the values of the coefficients $D_1, D_2, \dots, D_J, C_J$ in equation (3) using the corresponding values after wavelet decomposition; obtain a simulated prediction for the approximation C_J using a 3-level BP neural network model (let $\hat{C}_J(t)$ be its predicted value at time t); predict the details D_1, D_2, \dots, D_J using either cosine wave approximation or ARMA (let $\hat{D}_1(t), \hat{D}_2(t), \dots, \hat{D}_J(t)$ be their predicted values at time t). Then the combined prediction value $\hat{x}(t)$ of the time series $x(t)$ is

$$\hat{x}(t) = \hat{C}_J(t) + \hat{D}_1(t) + \hat{D}_2(t) + \dots + \hat{D}_J(t).$$

3 Example

The original series used comprised the first 600 data points of 1-Crude oil tabulated in the BRENT oil price time series(data resource : <http://tonto.eia.doe.gov/oog/ftparea/wogirs/xls/>). The graph of these 600 data points is shown below:

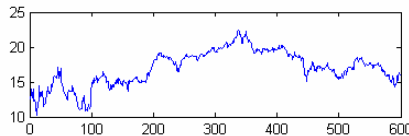


Fig. 1. The variation of oil prices with time (original data)

The last 88 data points were predicted using the first 512 data points. The db4 wavelet and a 3-level decomposition was used. The graph of the first 512 data points, as well as that of the reconstructed coefficients after wavelet decomposition are shown below:

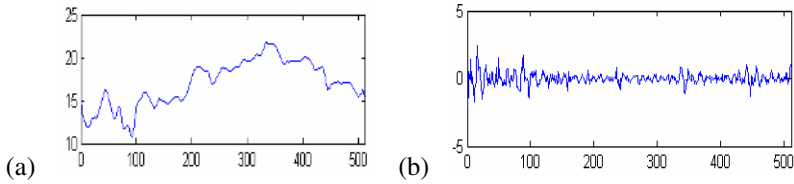


Fig. 2. Reconstructed values (a) low-frequency components (b) high-frequency components

Following the wavelet neural network prediction procedure, apply BP prediction to the low-frequency components (the approximated components) and cosine prediction to the high-frequency components (the details). Table 1 shows the fluctuation extent indices μ for the 3-level details. Choose $\mu_0 = 0.15$ in inequality (9). The level-3 details were used for calculating predictions. The mean period of level-3 is 6.3158 and the mean amplitude is 0.67655. The ARMA model was applied to level 1 and 2.

Table 1. 3-level details fluctuation extent indices

Level	1	2	3
fluctuation extent indices	0.3283	0.2133	0.1492

Finally, superpose the high-frequency components and low-frequency components prediction results to obtain a wavelet neural network prediction model. The actual and predicted values of the last 88 data points are shown in Fig. 4. The corresponding results using BP prediction are shown in Fig. 5. Table 2 compares the errors generated using these two methods.

Table 2. Comparison of prediction errors

	Wavelet neural network prediction	BP network prediction
mean square root error	0.3054	1.1263
mean absolute error	0.2000	0.9568
mean relative error	0.0117	0.0577

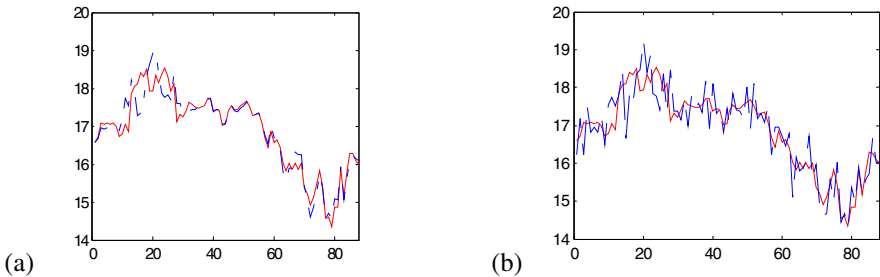


Fig. 3. (a) Comparison of wavelet neural network prediction values and actual values. (b) Comparison of BP network prediction values and actual values

Solid lines represent actual values; broken lines represent predicted values.

4 Conclusion

By using wavelet analysis of dynamic data, this paper introduced a prediction method for dynamic time series that is based on wavelet neural networks. The method predicts the data values after they have been wavelet transformed, fully utilizing the “mathematical microscope” property of these transforms. More precisely, the non-stationary time series of nonlinear systems are decomposed into different components, allowing a better separation of the general trend terms, the periodic terms and the random fluctuation terms. Consequently, by applying the most suitable prediction method to the components under different resolutions, the method produces more accurate prediction results.

References

1. Murtagh,F., Lstarch,J., Renaud,O.: Neuro-Wavelet Modeling. *Decision Support Systems*. 37 (2004) 475-484
2. Atoniadis,A., Saptinas,T.: Wavelet Methods for Continuous-Time Prediction Using Hilbert-Valued Auto-regressive Processes. *Journal of Multivariate Analysis*. 87(2003) 133-158
3. Wu,D., Wang,J., Teng,Y.: Prediction of Underground Water Levels Using Wavelet Decompositions and Transforms. *Journal of Hydro-Engineering*, 5(2004) 34-39
4. Liang,Q., Fan,Y., Wei,Y.: Prediction Methods for Oil Price Long-Term Trends Based on Wavelet Analysis and Their Verification. *Chinese Journal of Management Science*, 1(2005) 23-28.
5. Kim,C., Yu I., Song H.: Prediction of System Marginal Price of Electricity Using Wavelet Transform Analysis. *Energy Conversion and Management*. 13(2002) 1839-1851.
6. Feng, X., Gan, X.: *Numerical Functionals and Wavelet Theory*. Xidian University Press , Xi'an(2003)

Training Multi-layer Perceptrons Using MiniMin Approach

Liefeng Bo, Ling Wang, and Licheng Jiao

Institute of Intelligent Information Processing,
Xidian University, Xi'an 710071, China
{blf0218, wliiip}@163.com

Abstract. Multi-layer perceptrons (MLPs) have been widely used in classification and regression task. How to improve the training speed of MLPs has been an interesting field of research. Instead of the classical method, we try to train MLPs by a MiniMin model which can ensure that the weights of the last layer are optimal at each step. Significant improvement on training speed has been made using our method for several big benchmark data sets.

1 Introduction

In many important application areas such as pattern recognition, signal processing and control, it is needed to approximate an unknown nonlinear mapping through learning from examples. Multi-layer perceptrons (MLPs) [1] have been widely used to tackle this task, since they are shown to be universal function approximators [2].

Classical backpropagation (BP) algorithm [3] is usually quite slow due to nonlinear of MLPs and global property of sigmoid neuron. Many efforts have been made to improve training time. In 1990, Battiti and Masulli [4] used quasi-Newton algorithm (BFGS) to speedup MLPs. In 1994, Hagan and Menhaj [5] improved training time of MLPs by Levenberg-Marquardt algorithm where the Jacobian matrix is computed through a standard backpropagation technique that is much less complex than computing the Hessian matrix. In 1993, Moller [5] proposed a quite effective algorithm named scaled conjugate gradient (SCG) for MLPs. SCG is fully automated including no user dependent parameters and avoiding a time consuming line-search. Demuth's test report [7] also shows that SCG performs well over a wide variety of problems, particularly for networks with a large number of weights. For a good introduction of these algorithms, reader can refer to [8-9].

Our focus will be on two-Layer perceptrons with sigmoidal hidden units and a linear output unit. Our fundamental idea is that the weights of MLPs can be computed by a MiniMin model named MM-MLPs. This novel model allows the weights of the last Layer to be analytically calculated by linear equation systems. In other words, MM-MLPs can ensure that the weights of the last Layer are optimal at each step. An empirical study on four big data sets shows that MM-MLP yields a significant speedup relative to MLPs with the same training algorithm (in this paper, SCG is used). The speedup depends on the learning task. Experimental results seem to support that MM-MLPs usually obtain a bigger speedup for regression than for classification task.

2 MiniMin Multi-layer Perceptrons

We consider the training error to be the sum over output units of the squared difference between the desired output and actual output. Without loss of generality, we ignore the bias terms of network for convenience of formulation. In the classical MLPs, the weights is given by minimizing the following objective function

$$\min_{\mathbf{a}, \mathbf{W}} \left(E' = \left(\sum_{s=1}^c (\mathbf{Y}^s - \mathbf{H}\mathbf{a}^s)^T (\mathbf{Y}^s - \mathbf{H}\mathbf{a}^s) \right) \right). \tag{1}$$

where \mathbf{Y}^s and \mathbf{a}^s denote the s -th column of matrix \mathbf{Y} and \mathbf{a} respectively, and $\mathbf{H}_{ij} = \phi \left(\sum_{t=1}^n w_{ij} \mathbf{X}_{it} \right)$. Due to nonlinear and compact structure, many algorithms such as BP result in poor performance in this model. To ease this problem, we try to optimize the weights of MLPs by MiniMin model

$$\min_{\mathbf{W}} \left(E = \min_{\mathbf{a}} \left(f = \sum_{s=1}^c (\mathbf{Y}^s - \mathbf{H}\mathbf{a}^s)^T (\mathbf{Y}^s - \mathbf{H}\mathbf{a}^s) \right) \right). \tag{2}$$

By some mathematical tricks, we can get the analytical solution of the inner objective function. The inner objective function can be written as

$$E = \min_{\mathbf{a}} \left(\sum_{s=1}^c \left((\mathbf{a}^s)^T \mathbf{H}^T \mathbf{H} \mathbf{a}^s - 2(\mathbf{a}^s)^T \mathbf{H}^T \mathbf{Y}^s + (\mathbf{Y}^s)^T \mathbf{Y}^s \right) \right). \tag{3}$$

Let the derivative of f with respect to \mathbf{a}^s be zeros, we can compute $(\mathbf{a}^{opt})^s$ by

$$(\mathbf{a}^{opt})^s = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{Y}^s \tag{4}$$

where $(\mathbf{a}^{opt})^s$ is the s -th column of matrix \mathbf{a}^{opt} .

Substituting Eq. (4) into Eq. (3), we have

$$E = \sum_{s=1}^c \left(-(\mathbf{Y}^s)^T \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{Y}^s + (\mathbf{Y}^s)^T \mathbf{Y}^s \right). \tag{5}$$

Thus Eq. (2) is simplified into

$$\min_{\mathbf{W}} \left(E = \sum_{s=1}^c \left(-(\mathbf{Y}^s)^T \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{Y}^s + (\mathbf{Y}^s)^T \mathbf{Y}^s \right) \right). \tag{6}$$

The derivative of E with respect to the weights \mathbf{W}_{ij} can be computed by theorem 1.

Theorem 1

$$\frac{\partial E}{\partial \mathbf{W}_{ij}} = \sum_{i=1}^c 2 \left(\mathbf{Y}^s - \mathbf{H} (\mathbf{a}^{opt})^s \right)^T \left((\mathbf{a}^{opt})_{js} \Delta \mathbf{H}^j \odot \mathbf{X}^i \right) \tag{7}$$

where \odot denotes Hadamard product, $\Delta \mathbf{H}_{ij} = \Delta \phi \left(\sum_{t=1}^n w_{ij} \mathbf{X}_{it} \right)$ and $\Delta \mathbf{H}^j$ is the j -th column of matrix $\Delta \mathbf{H}$.

It is interesting to compare the derivative of E and the derivative of E' .

The derivative of E' with respect to \mathbf{W}_{ij} is

$$\frac{\partial E'}{\partial \mathbf{W}_{ij}} = \sum_{s=1}^c \left(2(\mathbf{Y}^s - \mathbf{H}\boldsymbol{\alpha}^s)^T \left(-\frac{\partial \mathbf{H}}{\partial \mathbf{W}_{ij}} \boldsymbol{\alpha}^s \right) \right). \quad (8)$$

Because \mathbf{W}_{ij} only appears in the j -th column of \mathbf{H} , Eq. (8) is further simplified as

$$\frac{\partial E'}{\partial \mathbf{W}_{ij}} = \sum_{s=1}^c \left(2(\mathbf{Y}^s - \mathbf{H}\boldsymbol{\alpha}^s)^T \left(-\boldsymbol{\alpha}_{js} \frac{\partial \mathbf{H}^j}{\partial \mathbf{W}_{ij}} \right) \right). \quad (9)$$

In terms of $\frac{\partial \mathbf{H}^j}{\partial \mathbf{W}_{ij}} = \Delta \mathbf{H}^j \odot \mathbf{X}^i$, Eq. (9) can be transformed into

$$\frac{\partial E'}{\partial \mathbf{W}_{ij}} = \sum_{s=1}^c \left(2(\mathbf{H}\boldsymbol{\alpha}^s - \mathbf{Y}^s)^T (\boldsymbol{\alpha}_{js} \Delta \mathbf{H}^j \odot \mathbf{X}^i) \right). \quad (10)$$

From Eq. (7) and Eq. (10), we can see that if replacing $\boldsymbol{\alpha}^s$ of Eq. (10) with $\boldsymbol{\alpha}^{opt}$, we can get Eq. (7). In MM-MLPs, $\boldsymbol{\alpha}^{opt}$ is optimal, which possible explains why our model usually has higher convergence rate.

According to Eq. (7), the computational complexity for the derivative of each weight is $O(cl)$, where l is the size of training samples. To obtain the gradient vector, we need to do it $(Ne \times n)$ times, which incurs a computational cost of $O(Ne \times ncl)$, where Ne is the size of hidden units. This cost seems too large. A better solution is based on the following theorem.

Theorem 2

$$\frac{\partial E}{\partial \mathbf{W}} = 2\mathbf{X}^T \left(\Delta \mathbf{H} \odot \left((\mathbf{H}\boldsymbol{\alpha} - \mathbf{Y}) (\boldsymbol{\alpha}^{opt})^T \right) \right) \quad (11)$$

A straightforward corollary of theorem 2 is that the gradient vector can be computed at $O(Ne \times nl + cnl)$ cost. Similar conclusion holds for MLPs model. Another time-consuming operation is computing $\boldsymbol{\alpha}^{opt}$ whose cost is $O(Ne^3 + Ne^2l)$. Thus the computational complexity of MM-MLPs at each step is $O(Ne \times nl + cnl + Ne^3 + Ne^2l)$. The computational complexity of MLPs at each step is $O(Ne \times nl + cnl)$.

As a result, we can derive that if $Ne < m$, then MM-MLPs and MLPs have the same computational complexity at each step. Since many practical problems satisfy this condition ($Ne < m$), our model should find wide applications.

3 Empirical Study

In order to know how well MM-MLPs work, we compare it with MLPs on two big data sets, each of which contains several thousand samples. Classification problems

are from Statlog [10] and regression problems are from Delve [11]. These data sets have been extensively used in testing the performance of diversified kinds of learning algorithms. Here, SCG algorithm is used to train networks. All the experiments are run on a personal computer with 2.4 GHz P4 processors, 2 GB memory and Windows XP operation system.

To avoid that the features with large magnitude dominate the output, all the training data are scaled in [-1, 1], then the test data are adjusted using the same linear transformation. The input weights are randomly initialized in the range $[-\frac{1}{\sqrt{m}}, \frac{1}{\sqrt{m}}]$

and hidden-to-output weights in the range $[-\frac{1}{\sqrt{Ne}}, \frac{1}{\sqrt{Ne}}]$. The number of hidden units depends on the task at hand; hence there is no foolproof method for setting the number of hidden units before training. In our experiments, the number of hidden units is determined by the 10-fold cross validation method.

The aim with this test is to compare the performance of MM-MLPs and MLPs on the classification problems. This task is to recognize the splice-junction gene sequences. This data set consists of 2000 training samples and 1186 test samples, 180 attributes of each sample. Three-layer network with five hidden units is used for this task. The convergence criterion is set to 0.030, 0.020, 0.010, 0.008, 0.006, 0.004 and 0.002. MM-MLPs and MLPs are tested on 10 random initial weights for each criterion.

From Table 1, we can see that MM-MLPs obtain the speedup range from 2 to 5 under the seven different convergence criterions. From Fig. 1, we can see that MM-MLPs reach the best test error with significantly fewer epochs.

Table 1. Epochs and training time of MM-MLPs and MLPs under the different criterion on Dna data set

Criterion	MM-MLPs		MLPs	
	Epoch	Time	Epoch	Time
0.030	11.200	1.482	22.600	2.833
0.020	17.200	2.217	30.700	3.803
0.010	28.900	3.659	53.900	6.609
0.008	32.500	4.249	60.200	7.486
0.006	38.000	4.861	102.000	12.655
0.004	55.000	6.847	229.400	29.788
0.002	130.200	15.541	472.400	60.971

The aim with this test is to compare the performance of MM-MLPs and MLPs on the regression problem. This task is to predict portion of time that CPUs run in user mode. This data set consists of 8192 samples, 21 attributes of each. Three-layer network with fifteen hidden units is used for this task. The convergence criterion is set to 0.000200, 0.00100, 0.00090, 0.00080, 0.00070, 0.00065, 0.00060 and 0.00055. MM-MLPs and MLPs are tested on 10 random initial weights for each criterion.

From Table 2, we can see that MM-MLPs obtain the speedup range from 6 to 12 under the seven different convergence criteria. From Fig. 2, we can see that MM-MLPs reach the best 10-fold cross validation error with significantly fewer epochs.

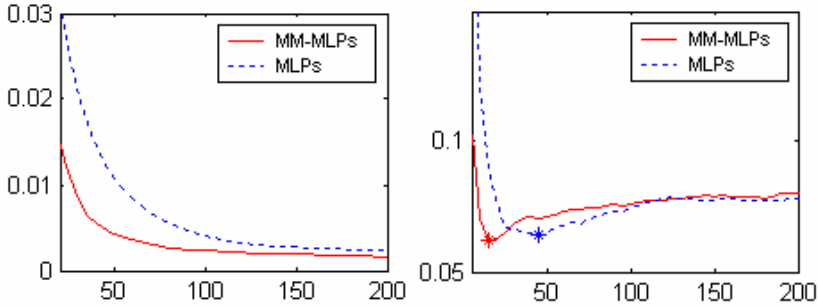


Fig. 1. Variation of the average training errors with epochs (left) and variation of average test errors with epochs (right) on Dna data set. “*” denotes the best test error.

Table 2. Epochs and training times of MM-MLPs and MLPs under the different criterion on Computer Activity data set

Criterion	MM-MLP		MLP	
	Epoch	Time	Epoch	Time
0.00200/19.6020	8.400	7.361	73.300	50.055
0.00100/9.8010	14.000	11.891	120.700	85.627
0.00090/8.8209	17.000	14.014	154.500	108.704
0.00080/7.8408	20.700	17.111	229.000	161.663
0.00070/6.8607	28.900	23.909	379.300	274.800
0.00065/6.3706	49.700	40.422	647.900	470.228
0.00060/5.8806	79.500	64.567	1156.800	835.763
0.00055/5.3906	306.200	246.749	2027.900	1483.817

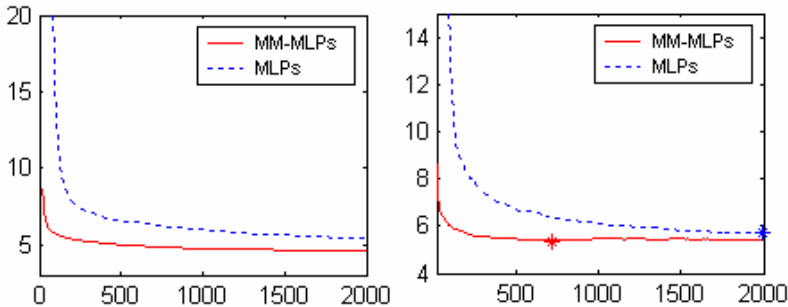


Fig. 2. Variation of average training errors with epochs (left) and variation of 10-fold cross validation errors with epochs(right) on Computer Activity data set. “*” denotes the best 10-fold cross validation error.

4 Conclusion

In this paper, MiniMin model is presented to train MLPs, which can ensure that the weights of the last layer are optimal at each step. The empirical comparisons on the four big benchmark data sets show that our method obtains a significant speedup relative to the classical formulation.

References

1. Rumelhart, D.E., Hinton, G.E. and Williams, R.J.: Learning representation of backpropagation errors. *Nature* 223 (1986) 533-536
2. Hornik, K., Stinchcombe, M. and White, H.: Multilayer feedforward networks are universal approximators. *Neural Networks* 2 (1989) 359-366
3. Rumelhart, D.E., Hinton, G.E. and Williams, R.J.: Learning Internal Representations by Error Propagation. In: *Parallel Distributed Processing: Exploration in the Microstructure of Cognition* (1986) 318-362
4. Battiti R. and Masulli, F.: BFGS Optimization for faster and automated supervised learning. *International Neural Network conference* (1990) 757-760
5. Hagan, M.T. and Menhaj, M.: Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks* 5 (1994) 989-993
6. Moller, M.F.: A Scaled conjugate gradient algorithm for fast supervised learning. *Neural Network* 6 (1993) 525-533
7. Demuth, H. and Beale, M.: *Neural network toolbox for use with MATLAB*. The MathWorks Inc. Natick, MA (1998)
8. Hagan, M.T., Demuth, H.B. and Beale, M.H.: *Neural Network Design*. Boston. MA: PWS Publishing (1996)
9. Boyd, S. and Vandenberghe, L.: *Convex Optimization*. Cambridge University Press (2004).
10. Michie, D., Spiegelhalter, D.J. and Taylor, C.C.: *Machine Learning, Neural and Statistical Classification*. Prentice Hall (1994)
11. Rasmussen, C.E., Neal, R.M., Hinton, C.E., Van Gamp, D., Revow, M., Ghahramani, Z., Kustra, R. and Tibshirani, R.: *The Deleve Manual* (1996)

Two Adaptive Matching Learning Algorithms for Independent Component Analysis*

Jinwen Ma, Fei Ge, and Dengpan Gao

Department of Information Science, School of Mathematical Sciences,
And LMAM, Peking University, Beijing 100871, China
jwma@math.pku.edu.cn

Abstract. Independent component analysis (ICA) has been applied in many fields of signal processing and many ICA learning algorithms have been proposed from different perspectives. However, there is still a lack of a deep mathematical theory to describe the ICA learning algorithm or problem, especially in the cases of both super- and sub-Gaussian sources. In this paper, from the point of view of the one-bit-matching principle, we propose two adaptive matching learning algorithms for the general ICA problem. It is shown by the simulation experiments that the adaptive matching learning algorithms can efficiently solve the ICA problem with both super- and sub-Gaussian sources and outperform the typical existing ICA algorithms in certain aspects.

1 Introduction

The independent component analysis (ICA) [1, 2] aims to blindly separate the independent sources \mathbf{s} from their linear mixture $\mathbf{x} = \mathbf{A}\mathbf{s}$ via

$$\mathbf{y} = \mathbf{W}\mathbf{x}, \quad \mathbf{x} \in \mathbb{R}^m, \quad \mathbf{y} \in \mathbb{R}^n, \quad \mathbf{W} \in \mathbb{R}^{m \times n}, \quad (1)$$

where \mathbf{A} is a mixing matrix, and \mathbf{W} is the de-mixing matrix to be estimated. In the general case, the number of mixed signals equals to the number of source signals, i.e., $m = n$, and \mathbf{A} is $n \times n$ nonsingular matrix. Although the ICA problem has been studied from different perspectives [3, 4, 5], it can be typically solved by minimizing the following objective function:

$$J = -H(\mathbf{y}) - \sum_{i=1}^n \int p_{\mathbf{W}}(y_i; \mathbf{W}) \log p_i(y_i) dy_i, \quad (2)$$

where $H(\mathbf{y}) = -\int p(\mathbf{y}) \log p(\mathbf{y}) d\mathbf{y}$ is the entropy of \mathbf{y} , $p_i(y_i)$ is the predetermined model probability density function (pdf), and $p_{\mathbf{W}}(y_i; \mathbf{W})$ is the probability distribution on $\mathbf{y} = \mathbf{W}\mathbf{x}$.

In the literature, how to choose the model pdfs $p_i(y_i)$ is still a key issue for the Eq.(2) based ICA algorithms [6, 7]. In fact, there has not existed any

* This work was supported by the Natural Science Foundation of China for Project 60471054.

efficient method for the cases that sources of both super-Gaussian and sub-Gaussian coexist in an unknown manner. In order to solve this difficult problem, Xu, Cheung and Amari [7] summarized the one-bit-matching conjecture which states that “all the sources can be separated as long as there is a one-to-one same-sign-correspondence between the kurtosis signs of all source pdf’s and the kurtosis signs of all model pdf’s”. Clearly, this conjecture is important since, if it is true, the complicated task of learning the underlining distribution of each source can be greatly simplified to the task of learning only its kurtosis sign.

Since there have been many studies supporting the one-bit-matching conjecture, e.g. [8, 9], it is widely believed in the ICA community. Recently, Liu et al. [10] proved that under certain assumptions, the global minimum of the objective function with the one-bit-matching condition leads to a feasible solution of the ICA problem. Ma et al. [11] further proved that under the same assumptions, all the local minimums of the objective function on the two-source ICA problem with the one-bit-matching condition lead to the feasible solutions of the ICA problem. Moreover, many simulation experiments also showed that the ICA problem can be solved successfully via minimizing the objective function under the one-bit-matching condition. So, we can believe that the minimization of the objective function with the one-bit-matching condition can lead to a feasible solution of the ICA problem. On the other hand, if we can parametrize the model pdfs such that they can become super-Gaussian or sub-Gaussian adaptively and make them match the source pdfs according to the kurtosis signs during the learning process, the minimization of the objective function can also lead to a feasible solution of the ICA problem. Xu et al. [6] have designed a model pdf with mixer of Gaussians and have shown its capability to estimate the source distribution. However, their model is complicated.

In this paper, we parametrize the model pdfs in two simple ways and propose two adaptive matching learning algorithms for the general ICA problem. That is, if we continuously change the parameter of model pdf, the kurtosis can continuously shift between positive and negative values. In the adaptation process, we learn the parameters of the model pdfs and the de-mixing matrix together via minimizing the objective function. It is shown by the simulation experiments that these adaptive matching learning algorithms can efficiently solve the ICA problem with both super- and sub-Gaussian sources and outperform the typical existing ICA algorithms in certain aspects.

2 The Adaptive Matching Learning Algorithms

We make two flexible pdf models, which can continuously shift between super-Gaussian and sub-Gaussian. One is a linear mixer of a super-Gaussian pdf and a sub-Gaussian pdf. The other is made of two translated super-Gaussian pdfs.

2.1 The Linearly Mixed Super- and Sub-Gaussian Model Pdf

We consider the following flexible parametric mixture pdf:

$$p_i(y_i) = \alpha_i p_{\text{super}}(y_i) + \beta_i p_{\text{sub}}(y_i), \quad (3)$$

where p_{super} is a super-Gaussian pdf, while p_{sub} is a sub-Gaussian pdf. α_i, β_i are parameters, with $\alpha_i, \beta_i \geq 0$, $\alpha_i + \beta_i = 1$. If α_i is greater than some constant value α_0 (determined by the two fixed pdfs), $p_i(y_i)$ is super-Gaussian. Otherwise, if $\alpha_i < \alpha_0$, $p_i(y_i)$ is sub-Gaussian.

We select the fixed pdf as

$$p_{\text{super}}(u) = \frac{1}{\pi} \text{sech}(u), \quad p_{\text{sub}}(u) = \frac{1}{2} [p_{N(1,1)}(u) + p_{N(-1,1)}(u)],$$

where $p_{N(\mu, \sigma^2)}$ denotes the Normal distribution.

In order to ensure that α_i, β_i satisfy the constraints, we use the following transformation:

$$\alpha_i = \frac{\exp(\gamma_{i1})}{\exp(\gamma_{i1}) + \exp(\gamma_{i2})}, \quad \beta_i = \frac{\exp(\gamma_{i2})}{\exp(\gamma_{i1}) + \exp(\gamma_{i2})},$$

so that α_i and β_j are equivalently expressed by free variables γ_{i1} and γ_{i2} . We can denote this flexible parametric mixture pdf by $p_i(y_i, \gamma_i)$ where $\gamma_i = (\gamma_{i1}, \gamma_{i2})$.

First, we must update \mathbf{W} to learn a de-mixing matrix. We compute the derivatives of the objective function $J = J(\mathbf{W}, \gamma)$ with respect to \mathbf{W} , and apply the natural gradient algorithm to modify \mathbf{W} in each step. The derivation is the same as in [4, 6, 8] and \mathbf{W} is modified by

$$\Delta \mathbf{W} = \eta [\mathbf{I} + \Phi(\mathbf{y})\mathbf{y}^T] \mathbf{W}. \quad (4)$$

where η is the learning rate, $\Phi(\mathbf{y}) = [\phi_1(y_1), \dots, \phi_n(y_n)]^T$, and

$$\phi_i(y_i) = \frac{p'_i(y_i, \gamma_i)}{p_i(y_i, \gamma_i)} = \frac{\alpha_i p'_{\text{super}}(y_i) + \beta_i p'_{\text{sub}}(y_i)}{\alpha_i p_{\text{super}}(y_i) + \beta_i p_{\text{sub}}(y_i)} \quad (5)$$

Meanwhile, we need to update the parameters of the model pdfs via the derivatives of $J(\mathbf{W}, \gamma)$ with respect to γ_{i1} and γ_{i2} . In fact, we have

$$\begin{aligned} \Delta \gamma_{i1} &= \eta \frac{\partial}{\partial \gamma_{i1}} \left(\sum_{l=1}^n \log p_l(y_l, \gamma_l) \right) \\ &= \eta \frac{p_{\text{super}}(y_i) - p_{\text{sub}}(y_i)}{\alpha_i p_{\text{super}}(y_i) + \beta_i p_{\text{sub}}(y_i)} \cdot \frac{\exp(\gamma_{i1}) \exp(\gamma_{i2})}{(\exp(\gamma_{i1}) + \exp(\gamma_{i2}))^2} \\ &= \eta \frac{(p_{\text{super}}(y_i) - p_{\text{sub}}(y_i)) \alpha_i \beta_i}{\alpha_i p_{\text{super}}(y_i) + \beta_i p_{\text{sub}}(y_i)} \end{aligned}$$

With the same derivation we can found out that $\Delta \gamma_{i2} = -\Delta \gamma_{i1}$.

Finally, we get to the following adaptive matching learning algorithm. At iteration k with an input \mathbf{x} , we can calculate \mathbf{y} via $\mathbf{y} = \mathbf{W}\mathbf{x}$. Then, \mathbf{W} and γ are modified by

$$\mathbf{W}^{(k+1)} = \mathbf{W}^{(k)} + \Delta \mathbf{W}, \quad \gamma_{ij}^{(k+1)} = \gamma_{ij}^{(k)} + \Delta \gamma_{ij}. \quad (6)$$

2.2 Mixed Translated Super-Gaussian Model Pdf

Another flexible model pdf is constructed by two symmetrically translated pdfs:

$$p_i(y_i) = \frac{1}{4}\text{sech}^2(y_i + \theta_i) + \frac{1}{4}\text{sech}^2(y_i - \theta_i), \tag{7}$$

where $\theta_i \geq 0$ is the model parameter. As θ increase, the kurtosis decrease and the model pdf change from super-Gaussian to sub-Gaussian.

The derivation of the learning algorithm is quite similar to that in the previous subsection. We replace θ_i with e^{γ_i} in order to keep it positive. The procedure to update \mathbf{W} is the same as Eq. (4) with

$$\phi_i(y_i) = \frac{-2\text{sech}^2(y_i + \theta_i) \tanh(y_i + \theta_i) + 2\text{sech}^2(-y_i + \theta_i) \tanh(-y_i + \theta_i)}{\text{sech}^2(y_i + \theta_i) + \text{sech}^2(y_i - \theta_i)}$$

Also, we need to update the parameters γ_i , and it turns out that

$$\begin{aligned} \Delta\gamma_i &= \eta \frac{\partial}{\partial \gamma_i} \left(\sum_{l=1}^n \log \left\{ \frac{1}{4}\text{sech}^2(y_i + e^{\gamma_i}) + \frac{1}{4}\text{sech}^2(y_i - e^{\gamma_i}) \right\} \right) \\ &= -2\eta\theta_i \frac{\text{sech}^2(y_i + \theta_i) \tanh(y_i + \theta_i) + \text{sech}^2(-y_i + \theta_i) \tanh(-y_i + \theta_i)}{\text{sech}^2(y_i + \theta_i) + \text{sech}^2(y_i - \theta_i)}. \end{aligned}$$

3 Experimental Results and Comparisons

First, we consider the ICA problem of seven independent sources including four super-Gaussian sources (generated from the exponential distribution $E(0.5)$, the Chisquare distribution $\chi^2(6)$, the gamma distribution $\gamma(1, 4)$ and the F distribution $F(10, 50)$, respectively) and three sub-Gaussian sources (generated from the beta distributions $\beta(2, 2)$, $\beta(0.5, 0.5)$, and the uniform distribution $U([0, 1])$, respectively). For each source, 100000 i.i.d. samples were generated and further normalized with zero mean and unit variance. The mixing matrix \mathbf{A} was randomly chosen.

We set the learning rate $\eta = 0.001$. \mathbf{W} was initially set as an identity matrix, and the initial model parameters were chosen such that the initial kurtosis of each mixture pdf $p_i(y_i, \gamma_i)$ is nearly zero.

The result of the adaptive matching learning algorithm using two model pdfs, respectively given by Eq. (3) and Eq. (7) are shown below, with \mathbf{W}_1 denoting the final \mathbf{W} got using the first model pdf and \mathbf{W}_2 using the second one.

$$\mathbf{W}_1 \mathbf{A} = \begin{bmatrix} -0.0125 & -0.0100 & -0.0283 & -0.0027 & -0.0041 & -0.0145 & -1.4867 \\ -0.0143 & 0.0021 & 0.0087 & -1.5540 & 0.0034 & -0.0366 & -0.0077 \\ 1.7166 & -0.0193 & 0.0159 & -0.0190 & -0.0074 & 0.0333 & 0.0066 \\ -0.0179 & 0.0006 & 1.6702 & -0.0203 & 0.0387 & 0.0075 & -0.0111 \\ 0.0149 & 1.5592 & -0.0202 & 0.0055 & 0.0024 & 0.0090 & 0.0168 \\ -0.0009 & -0.0204 & -0.0336 & 0.0018 & -1.4433 & -0.0060 & 0.0083 \\ -0.0188 & -0.0048 & -0.0157 & -0.0015 & -0.0143 & -1.4397 & -0.0162 \end{bmatrix}$$

$$\mathbf{W}_2\mathbf{A} = \begin{bmatrix} \boxed{-0.9769} & -0.0094 & 0.0078 & -0.0027 & 0.0303 & 0.0178 & 0.0193 \\ -0.0190 & -0.0021 & -0.0117 & -0.0161 & \boxed{-1.4425} & -0.0165 & -0.0045 \\ -0.0046 & -0.0027 & -0.0072 & 0.0052 & -0.0062 & -0.0150 & \boxed{-1.7547} \\ -0.0061 & -0.0002 & \boxed{0.9820} & -0.0131 & 0.0217 & 0.0241 & 0.0199 \\ 0.0159 & -0.0045 & -0.0258 & \boxed{0.9444} & 0.0186 & 0.0213 & 0.0443 \\ -0.0203 & -0.0220 & 0.0010 & -0.0054 & -0.0098 & \boxed{1.2511} & -0.0095 \\ 0.0271 & \boxed{0.9799} & -0.0113 & 0.0030 & -0.0008 & -0.0176 & -0.0510 \end{bmatrix}$$

For a feasible solution of the ICA problem, the obtained \mathbf{W} should make $\mathbf{W}\mathbf{A} = \mathbf{\Lambda}\mathbf{P}$ satisfied or approximately satisfied to a certain extent, where $\mathbf{\Lambda}\mathbf{P} = \text{diag}[\lambda_1, \lambda_2, \dots, \lambda_n]$ with each $\lambda_i \neq 0$, and \mathbf{P} is a permutation matrix. We can see, that our adaptive matching learning algorithm can solve this ICA problem of both super- and sub-Gaussian sources efficiently.

Next, we use audio data to perform the tests. Eight sound clips¹, each containing 100000 samples (at 22050Hz sample rate), were normalized and then mixed using an 8×8 random matrix. We process the mixed signals with our adaptive matching learning algorithms. We rearrange the output signals so that each output y_i matches the recovered source s_i . Figure 2 shows the wave forms of four of the eight sources and their corresponding recovered signals obtained by the algorithm given in Section 2.1.

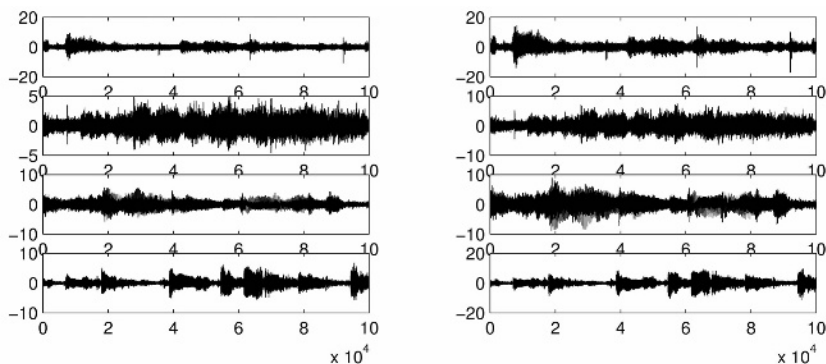


Fig. 1. Wave forms of 4 audio sources (left) and recovered signals (right)

For comparison, we performed experiments using the Extended Infomax algorithm[8] and the Fast-ICA algorithm[5]. Then we calculate signal-to-noise ratio (SNR) to evaluate each recovered signal. The results are summarized in Table 1. We can find that on the average, our two adaptive matching algorithms perform better than the Extended Infomax and the Fast-ICA algorithms in this test.

¹ wav files downloaded from <http://www-bcl.cs.may.ie/~bap/demos.html>

Table 1. The SNR of recovered sources by each method

Audio Source	Signal-to-Noise Ratio (dB)									
	1	2	3	4	5	6	7	8	Med.	Avg.
Adaptive Alg. 1	22.65	16.24	19.10	15.37	29.67	20.43	21.89	32.71	21.16	22.26
Adaptive Alg. 2	23.97	16.32	20.03	16.00	27.01	18.65	21.29	32.09	20.66	21.92
Extended Infomax	19.59	16.37	19.79	17.56	23.39	15.91	16.72	28.74	18.57	19.76
Fast-ICA	20.65	14.17	17.68	10.37	33.99	12.61	18.62	38.50	18.15	20.82

4 Conclusions

We have investigated the ICA problem from the point of view of the one-bit-matching principle and established two adaptive matching learning algorithms via two simple ways of parametrization of the model pdfs. It is demonstrated by the simulation experiments that the two adaptive learning algorithm solves the general ICA problem efficiently and even outperforms the typical existing algorithms in certain aspects.

References

1. Tong, L., Inouye, Y., Liu, R.: Waveform-preserving blind estimation of multiple independent sources. *IEEE Trans. on Signal Processing*, **41** (1993) 2461-2470
2. Comon, P.: Independent component analysis—a new concept? *Signal Processing*, **36** (1994) 287-314
3. Bell, A., Sejnowski, T.: An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, **7** (1995) 1129-1159
4. Amari, S. I., Cichocki, A., Yang, H.: A new learning algorithm for blind separation of sources. *Advances in Neural Information Processing*, **8** (1996) 757-763
5. Hyvärinen, A.: Fast and Robust Fixed-Point Algorithms for Independent Component Analysis. *IEEE Trans. on Neural Networks*, **10** (1999) 626-634
6. Xu, L., Cheung, C. C., Amari, S. I.: Learned parametric mixture based ica algorithm. *Neurocomputing*, **22** (1998) 69-80
7. Xu, L., Cheung, C. C., Amari, S. I.: Further results on nonlinearity and separation capability of a linear mixture ICA method and learned LPM. In C. Fyfe (Ed.), *Proceedings of the I&ANN'98*, (1998) 39-45
8. Lee, T. W., Girolami, M., Sejnowski, T. J.: Independent component analysis using an extended infomax algorithm for mixed subgaussian and supergaussian sources. *Neural Computation*, **11** (1999) 417-441
9. Welling, M., Weber, M.: A constrained EM algorithm for independent component analysis. *Neural Computation*, **13** (2001) 677-689
10. Liu, Z. Y., Chiu, K. C., Xu, L.: One-bit-matching conjecture for independent component analysis. *Neural Computation*, **16** (2004) 383-399
11. Ma, J., Liu, Z., Xu, L.: A further result on the ICA one-bit-matching conjecture. *Neural Computation*, **17** (2005) 331-334

Bioprocess Modeling Using Genetic Programming Based on a Double Penalty Strategy

Yanling Wu, Jiangan Lu, Youxian Sun, and Peifei Yu

National Laboratory of Industrial Control Technology,
Zhejiang University,
Hangzhou 310027, China
{ylwu, jglu, yxsun}@iipc.zju.edu.cn

Abstract. Using genetic programming (GP) integrated with nonlinear parameter estimation we can identify the model for avermectin process. In order to reduce the effect caused by bloating which appears when a GP run stagnates in the later period, a fitness function with a double penalty strategy is proposed. GP with this penalty strategy is less sensitive to the choice of penalty parameters and compromises the fitness and the complexity of an individual, so the method can save considerable amounts of computational effort and find models with better quality. In addition, we combine the mechanism knowledge of the fermentation in GP to increase the quality of population and the convergence speed. Experiments prove that this method outperforms standard GP in reducing computational effort and finding better models more quickly.

1 Introduction

In an industrial bioprocess, many biological variables are measured manually because of the lack of real-time measurements. The information about these variables is so little and lagging that it has a bad effect on the product quality and hampers the development of industrial bioprocesses. To overcome these problems, based on the relativity between biological and chemical variables (generally can be measured real-timely) [1], we model the avermectin process by GP, get the on-line estimation of biological variables and guide the production. GP was proposed by J.R.Koza in 1990's. It starts with an initial population created randomly, produces offspring by performing genetic operators on the parent population. The iterative process continues and then we get the best individual [2] when the termination criterion is satisfied. Generally, the size of individuals grows quickly and this bloating uses too much CPU time and memory. The worst is that it makes the model's ability to generalize compromised [3]. There are several strategies to solve this problem: modifying genetic operators, multi-objective GP and penalty function. Nevertheless, these strategies have certain disadvantages, such as needing the knowledge of how operators interact with the bloating, choosing an appropriate penalty factor, balancing the complexity of the individual model and the fitness in optimizing. We proposed a penalty function with two penalty parameters to limit the bloating and make the selection of penalty parameter much easier.

2 Modeling Avermectin Fermentation Process by GP

We modify the standard GP for the avermectin process: 1, some mechanism knowledge is combined into GP to improve the quality of the population. 2, a double penalty strategy is used to solve the problem of bloating. 3, principal component analysis is used to reduce the dimensions of the input variables. Here all the data used for modeling are obtained from industrial process.

2.1 Initial Population

In GP, the first step is the definition of a functional set and a terminal set. Here the functional set is: $F=\{+, -, *, /, \sin, \cos, \log, \wedge, \text{sqrt}\}$. Because there are too many variables in the bioprocess and there is a large amount of correlation among them, if all of these variables are used, we would get a too complex model and spend considerable amounts of computation time. The worst is that much measuring error would affect the accuracy of the model. So we use principal component analysis [4] to compress the dimension of the input variables. We select the first three principal components (x_1 , x_2 and x_3) as independent variables for GP based on cross check. Then the terminal set is defined as: $T=\{x_1, x_2, x_3, c\}$. c is a numerical parameter.

In GP, the creation of the initial population is a blind random search in the search space of the problem, so its individuals generally have poor fitness and the quality of population improves slowly. Because GP is an iterative process and is integrated with nonlinear regression, its computational effort is very huge. So it is important to improve the quality of the initial population to reduce the computational effort and increase the speed of convergence. The method is making the most of the mechanism knowledge of the bioprocess and some growth kinetic models (such as: Monod, Contris, etc.) to generate some possible individual models and replace the worst ones in the population. Of course, we don't know exactly the mechanism of the bioprocess, and have great difficulty in deciding which model is more appropriate, but these models may contain some useful genes, so we select a certain number of possible models which are created based on mechanism and some already known kinetic models. Considering the precise of the estimation and the complexity of the computation, we use second-order Runge-Kutta to discretize these models. For example, supposing kinetic model of Monod is selected and the total sugar in the substrate is the inhibiting substrate. Then Monod model ($(dx/dt=sx/(c+s)$, where s , x is the concentration of total sugar and biomass respectively) is discretized and simplified as:

$$x(k+1) = \left(c_4 s(k)x(k) + c_5 x(k) + c_6 s(k)^2 x(k) \right) / \left(c_1 + c_2 s(k) + s(k)^2 \right) \quad (1)$$

In our work, variables in the terminal set are principal components, so we also use the linear combination (which is defined by the inverse form of matrix in Equ.4) of these principal components to replace $s(k)$, $x(k)$. Then we translate equation 1 into a tree structure and use it as an individual in the initial population. We also can use several different tree forms to represent a same model to protect useful genes from being destroyed by crossover or mutation and use them to replace some worst individuals in the initial population and improve its quality.

2.2 Assignment of Fitness Values

In GP, the selection probability of an individual is decided by its fitness. So the fitness has crucial influence on GP’s performance. Typically, the fitness is designed as: $f_i = (\sum (\hat{y}(j) - y(j))^2) / N$, \hat{y} individual model’s output, y real output, $j=1 \dots N$ number of samples, $i=1 \dots M$ population size. But after some generations, a GP run usually stagnates and individual tree size grows rapidly [5,6]. This phenomenon of bloating excessively uses CPU time and memory, furthermore, the ability of generalization is compromised. So we combine a penalty function into the fitness:

$$f_i = (\sum (\hat{y}(j) - y(j))^2) / N + a(NOD(i) - b)^2 \times sign(NOD(i) - b) \tag{2}$$

$NOD(i)$, which represents the complexity of an individual, is the nodes of the i th individual. a, b control the severity of penalty for a tree’s size. $sign$ is SIGN function. a is a penalty parameter. If a is designed too large it would limit the growth of trees and prevent GP from finding a good model. On the other hand, if it is too small, individual tree would grow rapidly and the chance of GP to find a good model would reduce too. Here a double penalty strategy [7] is used to desensitize GP to the values of penalty parameter. The flow is:

1. The fitness of the current population is evaluated using two penalty parameters respectively, then forms two rank lists, each list corresponds one penalty parameter.
2. Create a list which mixes the influence of the two lists by the following method. Select the best individual of list 1, and then choose the best individual of list 2 that has not yet been selected. Repeat the process until all individuals are selected.
3. Perform mutation, crossover and reproduction operations on the individuals which are selected by linear ranking selection [8] and create offspring population.
4. Repeat the steps listed above until the termination criterion is satisfied. The best so far individual is the solution for the problem.

In order to make the penalty function appropriate for limiting the individual’s complexity, we modify the penalty function which is originally appropriate for constrains. The fitness should become better and increase the individual’s selection probability when the complexity of the individual decrease; become worse and decrease the individual’s selection probability when the complexity of the individual increases. b is the average complexity (nodes) of the current population. a , a penalty parameter, is a function of generations. In the beginning of a GP run, in order to ensure the diversity of the population, its value is low and do not limit the growth of individuals except some very complex individuals created by mutation. In the later period, increase its value to limit the growth of individuals. Moreover, from experience the good model for the bioprocess can not be too simple, so if the nodes of an individual is too small (for example 4), don’t improve its fitness. Thus the fitness is designed as:

$$f_i = \begin{cases} \frac{1}{N} \sum (\hat{y}(j) - y(j))^2 & NOD(i) < 4 \\ \frac{1}{N} \sum (\hat{y}(j) - y(j))^2 + a(t)(NOD(i) - b)^2 \times sign(NOD(i) - b) & \text{others} \end{cases} \tag{3}$$

2.3 The Model of the Avermectin Fermentation Process Obtained by GP

After PCA, the 8 dimensional variable X_0 (including previous total sugar, reducing sugar, nitrogen, biomass concentration, and current dissolved oxygen, gas flow rate, rotational speed, PH) is simplified as 3 dimensional variable X :

$$X = \begin{pmatrix} -0.35 & -0.47 & -0.17 & 0.36 & 0.44 & -0.23 & 0.23 & -0.46 \\ 0.37 & 0.12 & 0.49 & -0.23 & 0.10 & -0.51 & 0.51 & -0.16 \\ 0.28 & 0.08 & 0.16 & 0.84 & -0.32 & 0.16 & 0.23 & 0.04 \end{pmatrix} \times (X_0) \tag{4}$$

Then using GP, we get the model of the current biomass concentration (Xb):

$$Xb = 5.01(4.34\sqrt{x_1x_2} - 0.7382) \times (5.33x_2 - x_1 - 0.2035x_3) / (x_3 - 15.16x_1)^{15.16x_1} + 4.6 \tag{5}$$

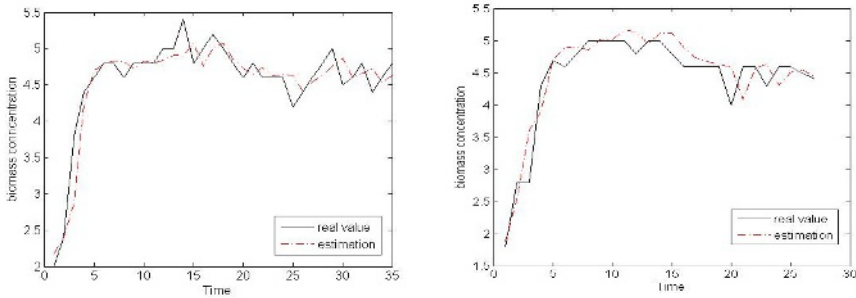


Fig. 1. Performance of the model on training data set (left) and testing data set (right)

Fig.1 shows plots of the model’s output and real output on the training and testing data set. The model has a Mean Square Error of 0.12 on the training data set and of 0.16 on the testing data set. So we can say that the model represents the process well, and the effect on the training data set is almost as good as on the testing data set.

3 Comparing with Other GP Methods

Here we compare GP having a double penalty strategy (DPS-GP) with GP having small penalty strategy (SP-GP) and large penalty strategy (LP-GP) on the following aspects. The results listed below are obtained by average values over 100 runs with the same initial population.

3.1 Best so Far Fitness against Computational Effort [9] for Generation g

This criterion implicitly means how fast a method finds a good model. Computational effort at generation g is defined as total nodes at generation g (all_nodes_g) and the total computational effort for generation g is $\sum_{j=1}^g all_nodes_j$. From Fig.2, we can see that

DPS-GP spends a much smaller computational effort than others to reach the same fitness and LP-GP has difficulty in finding a good model.

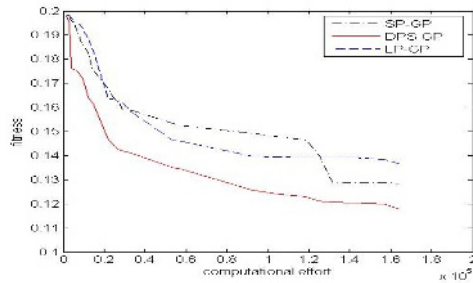


Fig. 2. The best so far fitness against the total computational effort for a generation

3.2 Success Rate for Different Penalty Strategies

A model is successful means that it performs well not only on the training data but also on the testing data. Success rate [10] is defined as the number of runs, in which a successful model has been found, divided by the total runs executed.

Table 1. Success rate of different penalty strategy

Name of Algorithm	Penalty strategy	Success rate
SP-GP	Small penalty	0.77
LP-GP	Large penalty	0.56
DPS-GP	Double penalty	0.88

Table 1 shows LP-GP performs worst. It limits bloating successfully but its success rate is the lowest because its large penalty prevents it from exploring all search space. SP-GP’s performance is also poor, its individual size bloats and usually the size of the best model found by it is larger than others, which has a bad effect on its success rate.

4 Conclusions

Penalty parameter has a crucial influence on the GP’s performance and its optimal value is difficult to find. Apart from tuning, there is no general solution to select penalties, so we propose GP with two penalty parameters to overcome the bloating and make the selection much easier. Furthermore, based on the mechanism knowledge of the bioprocess, we generate some possible models and replace the worst ones in the population to improve its quality and reduce the computational effort. From the effect of the model on the testing data sets, we conclude that the model is appropriate for the avermectin bioprocess and is promising in its control.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 20206027), the Key Technologies R&D Program in the 10th Five-year Plan of China (No. 2004BA210A01) and the National Basic Research Program (973) of China (No. 2002CB312200).

References

1. Wang Shuqing: Bioprocess modeling and computer control. ZheJiang Univ. publishing company (1998.)
2. Wolfgang, B., Francone, F. D., Robert, E. K., and Peter, N.: Genetic programming: An Introduction. Morgan Kaufmann, San Francisco, CA, (1998).
3. Liu Yong, Kang Lishan: Nonnumeric parallel algorithm-Genetic Algorithm. Science publishing company, BeiJing (2000).
4. Malinowski, E.R., Howery, D.G.: Factor Analysis in Chemistry. Wiley, New York(1980).
5. Terence, S., James, A.F.: Effects of code growth and parsimony pressure on populations in genetic programming. *Evolutionary Computation*, 6(4), (1999) 293-309.
6. John, R.K.: Genetic programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge, Massachusetts, (1992)
7. Riche, R.Le, Knopf-Lenoir, C, Hafika, R.T.: A Segregated Genetic Algorithm for Constrained structural Optimization. *Proceedings of the 6th International Conference on GA, San Francisco, USA (1995) 558—565.*
8. Baker, J E.: Adaptive Selection Methods for Genetic Algorithms. *Proceedings of the 1st International Conference on GA, Grefenstette, (1985) 110—111.*
9. Poli., R.: Evolution of graph-like programs with parallel distributed genetic programming. In T.Bäck, editor, *Proceedings of the 7th International Conference on Genetic Algorithms*, Morgan Kaufmann, (1997) 346-353.
10. Tomassini, M., Vanneschi, L., Cuendet, J., Fernandez, F.: A new technique for dynamic size populations in genetic programming. *Evolutionary Computation, CEC (2004) Vol.1 486 – 493.*

An Improved Gibbs Sampling Algorithm for Finding TFBS*

Caisheng He^{1,2} and Xianhua Dai¹

¹ Department of Electronic & Communications Engineering,
School of Information Science and Technology, Sun Yat-sen University,
Guangzhou 510275, China

² State Key Laboratory of Oncology in Southern China, Cancer Center,
Sun Yat-sen University, Guangzhou 510060, China
hecaisheng@yahoo.com.cn, issdxh@zsu.edu.cn

Abstract. Computational methods detecting the transcription factor binding sites (TFBS) remain one of the most intriguing and challenging subjects in bioinformatics. Gibbs sampling is essentially a heuristic method, and it is easy to trap into a nonoptimal “local maximum”. To overcome this problem and to improve the accuracy and sensitivity of the algorithm, we present an improved Gibbs sampling strategy MPWGMMS to search for TFBS. We have tested MPWGMMS and other existing Gibbs sampling algorithms on simulated data and real biological data sets with regulatory elements. The results indicate that MPWGMMS has better performance than other methods to a great extent in accuracy and sensitivity of finding true TFBS.

1 Introduction

Eukaryotic transcription factors (TF) have special structural features to recognize and bind to specific deoxyribonucleic acid (DNA) sequences. The transcription factor binding sites (TFBS) typically comprise a minority of the nucleotides within a promoter region. The common pattern of TFBS is also called a motif. Most motifs span 5-20 bp. Functional biological analyses reveal that coregulated genes with similar expression profiles share the some similar transcription regulatory mechanism.

Computational methods detecting TFBS remain one of the most intriguing and challenging subjects in bioinformatics. These methods can be divided into two major classes. The first class is string-based methods that mostly rely on counting and comparing oligonucleotide frequencies. The second class is based on probabilistic sequence models, such as expectation maximization (EM) and Gibbs sampling. For its sensitivity and fast convergence, Gibbs sampling has become one of the most popular tools for motif extraction in biological sequences.

In this paper, we briefly overview the several Gibbs motif sampling extensions and then describe our improved Gibbs sampling algorithm — multiple position weight matrices Gibbs motif sampling (MPWGMMS) for finding TFBS in DNA sequences.

* This work is supported by National Science Foundation grant 60474075 of China.

2 Existing Gibbs Sampling Extensions for Finding TFBS

Gibbs sampling (is also called “heat bath”) is a Markov chain Monte Carlo (MCMC) method for optimization by sampling. The idea is to describe a complex probability distribution in terms of a Markov chain built with the simpler marginals of the distribution [1].

In [2], Lawrence et al. presented the derivation of the basic algorithm for Gibbs motif sampling. In the recently years, several advanced methods based on Gibbs sampling algorithm for motif finding have been proposed, such as AlignACE [3], BioProspector [4], GLAM [5], Gibbs sampler [6], and SeSiMCMC [7].

AlignACE [3] returns a series of motifs found preferentially as weight matrices that are over-represented in the input set. The algorithm judges alignments sampled with a MAP (maximum *a priori* log likelihood) score. In BioProspector [4], zero to third-order Markov background models are used and the core sampling step is replaced by a threshold sampler. GLAM [5] automatically optimizes the alignment width and evaluates the statistical significance of its output. Motif Sampler [6] improves the performance of the Gibbs sampling by modeling the background model with a higher-order background model in the presence of noisy data. SeSiMCMC [7] optimizes the weight matrix for a given motif and spacer length, and then looks for best motif and spacer lengths for obtained motif positions.

Gibbs sampling is essentially a heuristic method, and it is not exhaustive. So these methods are heuristically search for a mathematically optimum alignment, and they are easy to trap into a nonoptimal “local maximum”. To cope with this problem, it is thus necessary to run it simply repeatedly starting with a set of random positions. In some methods, it reaches 100 repetitions so far.

To overcome the problem of initialization and getting stuck in local maxima and to improve the accuracy and sensitivity of the algorithm, we propose an improved Gibbs sampling strategy MPWGMMS to search for TFBS in coregulated genes.

3 Improved Gibbs Sampling Algorithm — MPWGMMS

3.1 Modeling TFBS

Before introduction to our algorithm MPWGMMS, we briefly describe the building probabilistic models for TFBS.

In the simplest model, we have a set S of N DNA sequences where each sequence contains a single copy of the motif of fixed length W . Except for the motif, a sequence is described as a sequence of independent nucleotides generated according to a single discrete distribution $\theta_0 = (q_0^A, q_0^C, q_0^G, q_0^T)^T$ called the background model. The motif is described as W independent positions generated according to different discrete distributions q_i^b . The motif model θ_w (equation 1) is also called position weight matrix (PWM).

To compute a PWM, a position frequency matrix (PFM) that contains the number of observed nucleotides at each position of the motif should be created firstly. Then the PFM is converted to a PWM using a formula (equation 2, f_i^b = counts of base b in position i ; N = number of sites; $s(b)$ = pseudocount function).

$$\theta_w = \begin{pmatrix} q_1^A & q_2^A & \dots & q_w^A \\ q_1^C & q_2^C & \dots & q_w^C \\ q_1^G & q_2^G & \dots & q_w^G \\ q_1^T & q_2^T & \dots & q_w^T \end{pmatrix}. \tag{1}$$

$$q_i^b = \frac{f_i^b + s(b)}{N + \sum_{b \in \{A,C,G,T\}} s(b)}. \tag{2}$$

For an observed segment x of length W , let P_x be the probability of segment x generated by model θ_0 and Q_x be the probability of segment x generated by model θ_w . The alignment vector of the segment in the sequence is sampled according to the distribution of normalized weight $A_x = Q_x / P_x$ (equation 3, $x_i =$ the base b in position i of segment x). We can find the alignment that maximizes the ratio of the corresponding segment probability to the background probability by updating this distribution.

$$A_x = Q_x / P_x = \prod_{i=1}^W q_i^{x_i} / \prod_{i=1}^W q_0^{x_i}. \tag{3}$$

3.2 Scoring Scheme

Since information content and related statistics have proven their usefulness for identifying and analyzing sequence alignments [8], information content of the motif matrix (equation 4) is chosen as the scoring scheme to measure the specificity of the alignment in MPWGMMS. So our aim is finding the TFBS with the highest IC value.

$$IC = \sum_{i=1}^W \sum_{b \in \{A,C,G,T\}} q_i^b \log_2 \frac{q_i^b}{q_0^b}. \tag{4}$$

3.3 MPWGMMS Algorithm

To improve the performance of finding motif, several modifications are made with respect to the original Gibbs sampling algorithm [2] in MPWGMMS.

First, simultaneous multiple motif searching is executed in every iteration. At the beginning of every run, the algorithm is initialized with alignments of multiple columns. The alignment of every column is initialized with random starting positions within the various sequences. Multiple columns are independently and simultaneously searching the motif based on basic Gibbs sampling strategy [2] in every iteration.

Second, at the end of every iteration, we introduce a PWMs mixture proposal. According to the PWMs of all sampling columns, we generate a mix-PWM and then update the PWM of each column by the mix-PWM. With the updated PWMs, we sample the alignment of each column again. Here we describe a simple PWMs mixture proposal. Let $\theta_{c,w}$ be the PWM of column c . Based on the average of the PFMs of all columns, a mix-PFM is created and then converted to the mix-PWM θ_{mix} . Then all PWMs are updated with θ_{mix} using equation 5 (Let α be the coefficient of the mixture, $\theta'_{c,w}$ be the updated PWM. $0 < \alpha < 1$). Finally, the alignment of each column is sampled again by the updated PWMs.

$$\theta'_{c,w} = (1 - \alpha)\theta_{c,w} + \alpha\theta_{mix} \tag{5}$$

Third, we improve the phase shift in motif searching of each column. After every M th iteration, we compare the current set of alignment with sets shifted left and right by up to maximum possible number of sites. Probability ratio of choosing phase shift is proportional to $\exp(IC)$ of the shifted motif alignment.

With the alterations of the simultaneous multiple motif searching, the PWMs mixture proposal and the improvement of the phase shift, MPWGMMS allows of escaping from local maxima and it fine-tunes the Gibbs sampling algorithm for motif finding in DNA sequences. The basic algorithm of MPWGMMS is described as Table 1.

Table 1. The basic algorithm of MPWGMMS for motif finding

INPUT: A set S of N sequences and the length W of the motif to search.

1. Initialize the alignment vector set $A = \{a_{c,i} | c = 1, \dots, C, i = 1, \dots, N\}$ uniformly at random, $C =$ number of columns.
2. Predictive update and sampling step:
 - (a) For each column c , proceed the GMS basic algorithm described in [2].
 - (b) Calculate $\theta_{c,w}$ of each column and create θ_{mix} from all $\theta_{c,w}$, then update $\theta_{c,w}$ with θ_{mix} , and then resample the alignment $a_{c,i}$ by the updated $\theta'_{c,w}$.
3. After every M th iteration, insert the phase shift.
4. Repeat step 2 and 3 until the Markov chain reaches convergence.

OUTPUT: An optimal motif matrix θ_w and an optimal alignment A .

4 Data Sets and Results

4.1 Data Sets for Testing

To test the performance of MPWGMMS in finding TFBS, we constructed several data sets listed in Table 2. The former 3 groups MCB, PDR3 and MCM1 shown in Table 3 in detail, which genes are known to be coregulated, are gotten from the promoter database of *saccharomyces cerevisiae* (<http://rulai.cshl.edu/SCPD/>). The latter 2 simulated groups, which data are also real promoter sequences sets downloaded from <http://bio.cs.washington.edu/assessment/download.html>, a well-conserved motif “CCGTACAAT” was inserted into these sequences at random positions.

Table 2. Overview of the contents of the data sets

Dataset name	Sequence length	Number of sequences	Consensus
MCB	800	6	ACGCGT
PDR3	800	7	TCCGYGGA
MCM1	1000	11	DWNMCHDDTDNGGHHM
Simulated data 1	509	8	CCGTACAAT
Simulated data 2	509	20	CCGTACAAT

Table 3. Summary of the biological TFBS data sets

Dataset name	Regulatory elements	Genes	ORFs
MCB	MCB	CDC2,CDC9,CDC6, CLN1,POL1,CDC21	YDL102W,YDL164C,YJL194W, YMR199W,YNL102W,OR074C
PDR3	PDR3	PDR3,SNQ2,HXT9, PDR15,YOR1, PDR5, HXT11	YBL005W,YDR011W,YJL219W, YDR406W,YGR281W, YOR153W,YOL156W
MCM1	MCM1	DIT1,SWI4,STE3, PMA1,PCK1,STE2, CCP1,PIS1,FAR1, MET2,CDC46	YDR403W,YER111C,YKL178C, YGL008C,YKR097W,YFL026W, YKR066C,YPR113W,YJL157C, YNL277W,YLR274W

4.2 Results

We tested our algorithm MPWGMMS and other existing Gibbs sampling algorithms on these data sets. Since Gibbs sampling is probabilistic, each test was repeated 20 times. All the results are listed in Table 4. The method GMS1 is referred to the algorithm in [2] and GMS2 is referred to the method on http://embnet.cifn.unam.mx/rsa-tools/gibbs_form.cgi.

We tested these existing Gibbs sampling methods with their default parameter settings on the data sets. GMS2 shows better performance than GMS1. Based on 1st-order background model, the Gibbs sampler [6] is still shown unsatisfied results on these data sets. Then we ran MPWGMMS on the data sets, all motifs except for MCM1 were found in all 20 runs, and MCM1 was detected in 90% of 20 runs.

We also tested AlignACE [3] on the data sets, but AlignACE could not discover them accurately at all.

The results in Table 4 indicate that the improved Gibbs sampling algorithm MPWGMMS has the best performance of a probabilistic implementation of motif finding in these Gibbs sampling algorithms on the data sets.

Table 4. Results of the motif finding by the different methods on the data sets. Each number corresponds to number and success rate of runs, out of 20, in which the corresponding consensus was found by these different algorithms.

Dataset name	GMS1 *	GMS2 *	Gibbs sampler	MPWGMMS
MCB	7 (35%)	14 (70%)	4 (20%)	20 (100%)
PDR3	7 (35%)	20 (100%)	5 (25%)	20 (100%)
MCM1	2 (10%)	8 (40%)	0 (0%)	18 (90%)
Simulated data1	2 (10%)	9 (45%)	9 (45%)	20 (100%)
Simulated data2	11 (55%)	20 (100%)	17 (85%)	20 (100%)

5 Discussion

We aimed at improving the accuracy and sensitivity of the Gibbs sampling algorithm for finding motifs in coregulated genes. To cope with the problem that Gibbs

sampling is easy to trap into a nonoptimal “local maximum”, we introduce the simultaneous multiple motif searching and PWMs mixture proposal and improve the phase shift to fine-tune the original Gibbs sampling algorithm. We tested MPWGMMS and other methods on both real promoter data and simulated data sets, and the results indicate that the improved algorithm MPWGMMS outperforms other existing Gibbs sampling algorithms by a significant scale.

Future work will concentrate on automatically optimization the alignment width and solution to the motifs with gaps.

References

1. Pierre Baldi, Søren Brunak : Bioinformatics: The Machine Learning Approach. 2nd edn. MIT Press, Cambridge MA (2001)
2. Charles E., Stephen F., Mark S., et al.: Detecting Subtle Sequence Signals: A Gibbs Sampling Strategy for Multiple Alignment. *Science*. 262 (1993) 208-214
3. Jason D., Preston W., Saeed Tavazoie, et al.: Computational Identification of Cis-regulatory Elements Associated with Groups of Functionally Related Genes in *Saccharomyces cerevisiae*. *J. Mol. Biol.* 296 (2000) 1205-1214
4. X.Liu, D.L.Brutlag, J.S.Liu : BioProspector: Discovering Conserved DNA Motifs in Upstream Regulatory Regions of Co-expressed Genes. *Proc. Pacific Symp. Biocomputing*. 6 (2001) 127-138
5. Martin C., Ulla Hansen, John L., et al.: Finding Functional Sequence Elements by Multiple Local Alignment. *Nucleic Acids Research*. 32(1) (2004) 189-200
6. Yves Moreau, Frank De Smet, Gert Thijs, et al.: Functional Bioinformatics of Microarray Data: From Expression to Regulation. *Proc. of the IEEE*. 90(11) (2002) 1722-1743
7. Favorov A.V, Gelfand M.S., Gerasimova A.V., et al.: Gibbs Sampler for Identification of Symmetrically Structured, Spaced Dna Motifs with Improved Estimation of the Signal Length and its Validation on the Arca Binding Sites. *Proc. of BGRS 2004*. (2004) 269-272
8. Gerald Z., Gary D.: Identifying DNA and Protein Patterns with Statistically Significant Alignments of Multiple. *Bioinformatics*. 15 (1999) 563-577

A Novel Fisher Criterion Based S_t -Subspace Linear Discriminant Method for Face Recognition

Wensheng Chen^{1,2}, Pong C. Yuen³, Jian Huang³, and Jianhuang Lai⁴

¹ Department of Mathematics, Shenzhen University, 518060 China

² Key Laboratory of Mathematics Mechanization, CAS, Beijing 100080, China
chenws@szu.edu.cn

³ Department of Computer Science,
Hong Kong Baptist University, Hong Kong, China
{pcyuen, jhuang}@comp.hkbu.edu.hk

⁴ Department of Mathematics, Sun Yat-Sen University, 510275, China
stsljh@zsu.edu.cn

Abstract. In this paper, a novel Fisher criterion is introduced and shown to be equivalent to the traditional Fisher criterion. Based on this new Fisher criterion and simultaneous diagonalization technique, a S_t -subspace Fisher discriminant (S_t -SFD) method is developed to deal with the small sample size (S3) problem in face recognition. The proposed method overcomes some drawbacks of existing LDA based algorithms. Also, our method has good computational complexity. Two public available databases, namely ORL and FERET databases, are exploited to evaluate the proposed algorithm. Comparing with existing LDA-based methods in solving the S3 problem, the proposed S_t -SFD method gives the best performance.

1 Introduction

Linear discriminant analysis (LDA) is a popular statistic method for both feature extraction and dimensionality reduction, which was developed by Belhumeur et al in 1997 [1]. After that, a large number of LDA-based algorithms/systems have been proposed. LDA is theoretically sound and a number of LDA-based face recognition algorithms/systems have been developed in the last decade [1]-[11]. However, the major drawback of LDA is the so-called Small Sample Size (S3) problem. This problem always occurs when the total number of training samples is smaller than the dimensionality of feature vector. Under this situation, the within-class scatter matrix S_w becomes singular and direct applying LDA approach is impossible. To overcome S3 problem, some algorithms, such as PCA+LDA method [1], subspace methods (Direct LDA [2], Huang et al's method [5]) and regularized discriminant analysis (RDA) method [6] [7] etc, are developed.

PCA+LDA method [1], also called Fisherface method, applies an intermediate dimension reduction stage using PCA on original feature space such that the within-class scatter matrix S_w is nonsingular and then the traditional LDA can

be performed in the PCA-transformed feature space. However, the null space of S_w contains many useful discriminant information for PR. Therefore, discarding all the null space of S_w to guarantee that S_w is a full rank matrix will lose many important discriminant information.

Yu et al [2] proposed a so-called direct LDA (DLDA) method to deal with S3 problem. They first discard the null space of between-class scatter matrix S_b and then keep the null space of S_w . Two scatter matrices S_w and S_b are diagonalized simultaneously by DLDA-projection-matrix. Since $\text{Rank}(S_b) \leq \text{Rank}(S_w)$, first discarding the null space of S_b will lead to discard partial or entire null space of S_w as well. It will decrease discriminant power.

Huang et al [5] proposed a improved null space method. It was shown in [5] that the null space of the total scatter matrix S_t is the intersection of the null spaces of within-class scatter matrix S_w and between-class scatter matrix S_b , i.e., $N(S_t) = N(S_w) \cap N(S_b)$. Moreover, this intersection set $N(S_t)$, the null space of S_t , contains no useful information for pattern classification. So, they first discard the null space $N(S_t)$ for dimensionality reduction without loss useful discriminant information. This dimensionality reduction technique will be exploited in this paper. But their proposed method have to solve three eigen-systems, the computational cost of Huang et al method is high.

For existing RDA method [6], it proposed 3-parameter method to solve S3 problem. This method is executed in the full original sample space, the dimensionality of which is very large. They [6] first have to use D4 wavelet decomposition to perform dimensionality reduction. In addition, it's complexity and difficult to determine three optimal parameters.

In view of the limitations in existing LDA-based methods, this paper introduces a new Fisher criterion which is shown to be equivalent to the commonly used traditional Fisher criterion. Based on the introduced Fisher criterion, a novel S_t -subspace Fisher discriminant method is proposed and developed. Moreover, the proposed method has good computational complexity.

Two public available databases, namely ORL and FERET face databases are used to evaluate the proposed method for face recognition. Comparing with the existing LDA-based methods, the results are encouraging.

The rest of this paper is organized as follows. Our proposed S_t -SFD method is given in Section 2. The experimental results and computational complexity are reported in Section 3. Finally, Section 4 draws the conclusions.

2 Proposed S_t -SFD Method

In this section, we propose and develop a novel S_t -subspace Fisher discriminant method for face recognition based on a new Fisher criterion. Details are discussed as follows.

2.1 Some Notations and Assumptions

Let d be the dimensionality of original sample feature space and C be the number of sample classes, the total original sample $X = \{X_1, X_2, \dots, X_C\}$, the j th class

X_j contains N_j samples, i.e., $X_j = \{x_1^{(j)}, x_2^{(j)}, \dots, x_{N_j}^{(j)}\}, j = 1, 2, \dots, C$. Let N be the total number of original training samples, $\bar{x}_j = \frac{1}{N_j} \sum_{x \in X_j} x$ be the mean of the sample class X_j and $\bar{x} = \frac{1}{N} \sum_{j=1}^C \sum_{x \in X_j} x$ be the global mean of the total original sample X . In discriminant analysis, three scatter matrices, namely within-class, between-class and total scatter matrices, are defined respectively as follows:

$$S_w = \frac{1}{N} \sum_{j=1}^C \sum_{x \in X_j} (x - \bar{x}_j)(x - \bar{x}_j)^T = \Phi_w \Phi_w^T, \tag{1}$$

$$S_b = \frac{1}{N} \sum_{j=1}^C N_j (\bar{x}_j - \bar{x})(\bar{x}_j - \bar{x})^T = \Phi_b \Phi_b^T, \tag{2}$$

$$S_t = S_w + S_b = \frac{1}{N} \sum_{j=1}^C \sum_{x \in X_j} (x - \bar{x})(x - \bar{x})^T = \Phi_t \Phi_t^T, \tag{3}$$

where $\Phi_w, \Phi_t \in R^{d \times N}$ and $\Phi_b \in R^{d \times C}$.

2.2 Traditional LDA

In traditional LDA method, the Fisher criterion function is defined as $J_1(W) = \text{tr}(W^T S_b W) \text{tr}(W^T S_w W)^{-1}$, where notation $\text{tr}(A)$ denotes the trace of matrix A . The goal of Fisher linear discriminant analysis is to find an optimal projection $W^* : R^d \rightarrow R^m, d > m$, such that

$$W^* = \arg \max_W J_1(W) \tag{4}$$

The problem (4) is equivalent to solving the eigen-system: $S_w^{-1} S_b W = W \Lambda$, where Λ is a $d \times d$ diagonal eigenvalue matrix.

2.3 Equivalent Fisher Criterion

In this section, a new Fisher criterion function is introduced and defined by $J_2(W) = \text{tr}(W^T S_t W) \text{tr}(W^T S_w W)^{-1}$. For this function, we have the following theorem. (Proof is omitted)

Theorem 1. *Matrix W maximizes the Fisher criterion function $J_1(W)$, if and only if W maximizes $J_2(W)$.*

Theorem 1 indicates that the new Fisher criterion function $J_2(W)$ is equivalent to the traditional version $J_1(W)$. In this paper, we will substitute the new Fisher criterion function $J_2(W)$ for the traditional version $J_1(W)$.

2.4 Simultaneous Diagonalization Strategy

Based on the analysis in above sections, the task of LDA is equivalent to solving the following problem

$$W^* = \arg \max_W J_2(W). \tag{5}$$

Also, the problem (5) is equivalent to the eigen-system: $S_w^{-1}S_tW = WA$, where A is a $d \times d$ diagonal eigenvalue matrix. When S3 problem occurs, it is impossible to solve the equivalent eigen-system directly, since S_w is singular. To deal with S3 problem, simultaneous diagonalization strategy is exploited. The basic idea of simultaneous diagonalization strategy is to find a optimal projection matrix W , such that

$$W^T S_t W = D, \quad W^T S_w W = 0,$$

where D is a diagonal and non-singular matrix.

From the statistical point of view [12], the null space of S_w provides most discriminant power for pattern classification, while the null space of S_b contains no useful information for feature discriminant. It showed in paper [5] that $N(S_t) = N(S_w) \cap N(S_b)$. Therefore, in our method, first discarding the null space of S_t will not loss useful discriminant information.

2.5 S_t -SFD Algorithm

Based on above sections, our S_t -SFD algorithm is designed as follows.

Step 1: Perform eignvalue decomposition on $N \times N$ matrix $\Phi_t^T \Phi_t \stackrel{\text{evd}}{=} V_t \Lambda_t V_t^T$, where $V_t = [v_1, \dots, v_N] \in R^{N \times N}$ is an orthonormal matrix, the diagonal matrix $\Lambda_t = \text{diag}\{\lambda_1, \dots, \lambda_\tau, 0, \dots, 0\} \in R^{N \times N}$ with $\lambda_1 > \dots > \lambda_\tau > 0$. Denote the first τ columns of V_t by $\tilde{V}_t \in R^{N \times \tau}$ and $D = \text{diag}\{\lambda_1, \dots, \lambda_\tau\} \in R^{\tau \times \tau}$. Calculate matrix $U_\tau = \Phi_t \cdot \tilde{V}_t \cdot D^{-1/2} \in R^{d \times \tau}$, where $\tau \leq N - 1$.

Step 2: Calculate $S'_w = U_\tau^T S_w U_\tau \in R^{\tau \times \tau}$ and perform eignvalue decomposition: $S'_w \stackrel{\text{evd}}{=} U_w \Lambda_w U_w^T$, where $U_w = [u_1, \dots, u_\tau] \in R^{\tau \times \tau}$ is an orthonormal matrix, the diagonal matrix $\Lambda_w = \text{diag}\{\delta_1, \dots, \delta_m, 0, \dots, 0\} \in R^{\tau \times \tau}$ with $\delta_1 > \dots > \delta_m > 0$, $m \leq \min\{N - C, \tau\}$.

Step 3: Denote the last $(\tau - m)$ columns of U_w by \tilde{U}_w , i.e., $\tilde{U}_w = [u_{m+1}, \dots, u_\tau] \in R^{\tau \times (\tau - m)}$.

Step 4: The optimal projection matrix: $W = U_\tau \tilde{U}_w \in R^{d \times (\tau - m)}$, which satisfies: $W^T S_t W = D$ and $W^T S_w W = 0$.

Remark: In many situation for proposed algorithm, $\tau = N - 1$ and $m = N - C$, hence the number of discriminant vectors is $\tau - m = C - 1$, which coincides with the number of ideal discriminant vectors [1].

3 Experimental Results

The ORL database and FERET database are selected for the evaluation. The original images with resolution 112x92 all are reduced to wavelet feature face with resolution 30x25 after two-level D4 wavelet decomposition.

In ORL database, there are 40 persons and each person consists of 10 images with different facial expressions, small variations in scales and orientations. Image variations of one person in the database are shown in figure 1. In FERET



Fig. 1. Images of one person from ORL database

database, we select 250 people, 4 images for each individual. Face image variations in FERET database include pose, illumination, facial expression and aging. Images from two individuals are shown in figure 2.



Fig. 2. Images of two persons from FERET database

3.1 Results on ORL Database

The experimental setting on ORL database is as follows. We randomly selected n ($n=2$ to 9) images from each individual for training while the rest ($10 - n$) images are for testing. The experiments are repeated 10 times and the average accuracy is then calculated. The results are recorded and tabulated in Table 1 and plotted in Figure 3(a). It can be seen from Table 1 that the recognition accuracy of the proposed method increases from 85.75% with 2 training images to 97.75% with 9 training images. The results are encouraging.

In order to compare the performance of the proposed method with existing methods, the same experiments are conducted using Fisherface method [1], Direct LDA method [2] and Huang et al method [5]. It can be seen from table 1 that when 2 images are used for training, the accuracy for Fisherface, Direct LDA and Huang et al methods are 81.88%, 78.34% and 83.59% respectively. When the number of training images is equal to 9, the accuracy for Fisherface is increased to 95.75% while Direct LDA and Huang et al methods are 96.00% and 97.50% respectively. The results show that our proposed S_t -SFD method gives the best performance for all cases in ORL database.

Table 1. Performance comparison on ORL Database, TN=Number of Training samples

TN	Fisherface [1]	DLDA [2]	Huang [5]	Our Method
2	81.88%	78.34%	83.59%	85.75%
3	87.18%	83.96%	89.03%	90.89%
4	92.04%	89.75%	92.92%	94.42%
5	93.30%	91.55%	94.55%	96.45%
6	93.06%	91.69%	95.56%	95.87%
7	93.67%	93.75%	95.58%	96.50%
8	94.50%	93.69%	96.00%	97.63%
9	95.75%	96.00%	97.50%	97.75%

3.2 Results on FERET Database

The section reports the results of the proposed S_t -RFD method on FERET database. We randomly select 2 images from each person for training ($2 \times 250 = 500$ images for training), while the rest of images of each individual are selected for testing ($2 \times 250 = 500$ images for testing). The experiments are repeated 10 times and the average accuracies of rank 1 to rank 5 are recorded in table 2 and shown in the Figure 3(b). The recognition accuracy of our method increases from 78.50% with rank 1 to 85.56% with rank 5. Comparing with other LDA-based methods, namely, Fisherface [1], direct LDA (DLDA) [2] and Huang et al methods [5], the recognition accuracies of Fisherface, direct LDA and Huang et al methods increase from 72.46%, 68.14% and 72.80% with rank 1 to 81.24%, 78.16% and 81.62% with rank 5 respectively. It can be seen that the proposed method gives the best performance for all cases. Comparing with the best algorithm reported in the literature, the proposed method gives around 6% accuracy improvement with rank 2 and 4% with rank 5.

Table 2. Performance comparison on FERET Database

	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
DLDA [2]	68.14%	72.56%	75.02%	76.68 %	78.16%
Fisherface [1]	72.46%	76.46%	78.66%	80.18%	81.24%
Huang et al [5]	72.80%	77.16%	79.16%	80.48%	81.62%
Our Method	78.50%	81.78%	83.46%	84.64%	85.56%

3.3 Computational Complexity

The major difference on complexity of different methods is on the training process to solve eigensystems. The computational cost of Fisherface includes solving one N by N eigensystem, $O(N^3)$, and two $(N - C) \times (N - C)$ eigensystem, $O((N - C)^3)$. In Direct LDA, the computational cost involves solving one C by C eigensystem, $O(C^3)$ and one $m \times m$ ($m \leq C - 1$) eigensystem, $O(m^3)$. In Huang's method, the computational cost consists of three parts. The first part is to solve one N by N eigensystem, $O(N^3)$. The rest parts are to solve one $(N - 1)$

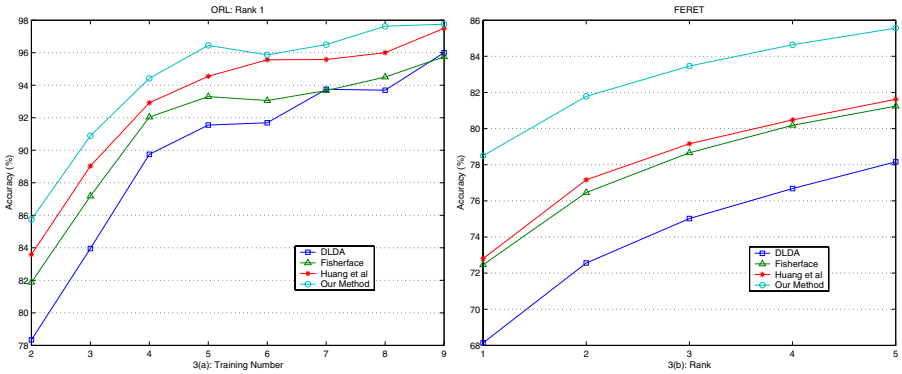


Fig. 3. Performance on (a) ORL database and (b) FERET database

by $(N - 1)$ eigensystem, $O((N - 1)^3)$ and one $(C - 1) \times (C - 1)$ eigensystem, $O((C - 1)^3)$. The computational cost of our proposed method consists of two parts, which involves solving one N by N eigensystem, $O(N^3)$ and one $\tau \times \tau$ ($\tau \leq N - 1$) eigensystem, $O(\tau^3)$.

4 Conclusions

A new Fisher criterion based S_t -subspace Fisher discriminant (S_t -SFD) algorithm has been developed and reported in this paper for solving the small sample size problem (S3) in face recognition. The proposed algorithm also has good computational complexity. Our S_t -SFD method has been evaluated with two public available databases, namely ORL and FERET databases. A comprehensive comparison with the existing LDA-based methods, namely Fisherface, Direct LDA and Huang et al methods, is also performed. Comparing with these three methods, the proposed method gives the best performance for both databases.

Acknowledgement

This project was supported by the Science Faculty Research grant of Hong Kong Baptist University RGC Earmarked Research Grant HKBU-2119/03E and NSF of China under Grant No.60373082. The authors would like to thank Olivetti Research Laboratory for contribution of the ORL database and US Army Research Laboratory for the FERET database.

References

1. Belhumeur, P. N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **19**, No. 7 (1997) 711–720
2. Yu, H. and Yang, J.: A direct LDA algorithm for high-dimensional data — with application to face recognition. *Pattern Recognition* **34** (2001) 2067–2070

3. Chen, L., Liao, H., Ko, M., Lin, J. and Yu, G.: A new LDA-based face recognition system, which can solve the small sample size problem *Pattern Recognition* **33**, No. 10, (2000) 1713–1726
4. Martinez, A. M. and Kak, A. C.: PCA versus LDA. *IEEE Trans. Pattern Anal. Mach. Intell.* **23** (2001) 228–233
5. Huang, R., Liu, Q., Lu, H. and Ma, S. D.: Solving small sample size problem in LDA. *Proceeding of International Conference in Pattern Recognition (ICPR 2002)* **3**(2002) 29–32
6. Dai, D. Q. and Yuen, P. C.: Regularized discriminant analysis and its application on face recognition. *Pattern Recognition*, **36** (2003) 845–847
7. Chen, W. S., Yuen, P. C., Huang, J.: A New Regularized Linear Discriminant Analysis Methods to Solve Small Sample Size Problems. *Int. J. Pattern Recognit. Artif. Intell.*, **19**, No.7, (2005). (To be published)
8. Lu, J., Plataniotis, K. N. and Venetsanopoulos, A. N.: Face Recognition Using LDA Based Algorithms. *IEEE Transactions on Neural Networks*, **14**, No.1 (2003) 195–200
9. Duin, R.P.W. and Loog, M.: Linear dimensionality reduction via a heteroscedastic extension of LDA: the Chernoff criterion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **26**, No. 6 (2004) 732–739
10. Chen, W. S., Yuen, P. C., Huang, J. and Dai, D. Q.: Kernel Machine-based One-parameter Regularized Fisher Discriminant Method for Face Recognition. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, **35**, No.4, (2005) 659–669
11. Ye J. P. and Li O.: A Two-Stage Linear Discriminant Analysis via QR-Decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**, No. 6 (2005) 929–941
12. Fukunaga, K.: *Introduction to Statistical Pattern Recognition*. 2nd ed. London, Academic Press (1999)

EmoEars: An Emotion Recognition System for Mandarin Speech

Bo Xie, Ling Chen, Gen-Cai Chen, and Chun Chen

College of Computer Science, Zhejiang University, Hangzhou 310027, P.R. China
lingchen@cs.zju.edu.cn

Abstract. In this paper, an emotion recognition system for mandarin speech is presented. Five basic human emotions including angry, fear, happy, neutral and sad are investigated. The recognizer is based on neural network with OCON and ACON architecture. Some novel feature selection methods are also added as optional tool to enhance the efficiency and classification accuracy. The system can train speaker dependent emotion speech model through online emotional utterance recording. Experiment results show that emotion can be recognized through neural network model, the best mean accuracy is 86.7%. In addition, the feature selection module is effective to reduce the compute load and increase the generalization ability of the recognizer.

1 Introduction

In human-machine interaction, the machine can be made to produce more appropriate responses if the state of emotion of the person can be accurately identified. In addition, the emotion recognition can also be used in tutoring, alerting, and entertainment [1]. Emotion recognition in speech has become one of the research fields for emotional human-computer interaction or affective computing [2].

Although there are a number of system proposed for emotion recognition based on facial expressions, only a few system based on speech input are reported in the literature, especially for mandarin language. ASSESS [3] is a system that use peaks and troughs of fundamental frequency, intensity and boundaries of pauses and fricative bursts to recognize Anger, Happy, Fear and Sadness based on discriminant analysis. Dellaert et al. [4] focused on the F0 information. Also the four emotions were considered. Using K-nearest neighbors as classifier and majority voting of specialists, it can achieve 79.5% recognition accuracy.

Performance of emotion recognition largely depends on acoustic features used in a classifier. Murray [5] summarized the relationship between emotion and acoustic features including pitch, intensity, rate and voice quality etc. The features adopted by most emotion classification research focus on statistics of fundamental frequency, energy contour, duration of silence and voice quality. Some feature selection methods are also added to reduce compute load and improve more recognition accuracy. Dellaert et al. [4] firstly used promising first selection (PFS) and sequential forward selection (SFS) for emotion recognition. They selected 5 features from 17, and error rate decreased by 3.5 %. Lee [6] used those two methods, and error rate decreased by 10

% on average. Kwon [7] selected features according to a two dimensional ranking figure result from SFS and sequential backward selection (SBS) methods.

This paper presents a mandarin speech emotion recognition system: EmoEars. The system can record emotional utterance online to create speaker dependent emotional speech samples. Feature selection can select more relevant features. Neural network based classifier will be offline trained using the sample and selected features. With the trained model, the system can online recognize the emotional state of the speaker.

The main contributions of this paper lie in: 1. implement a mandarin speech emotion recognition system; 2. feature selection module is integrated into the system to increase the efficiency; 3. evaluate the system in feature selection, the architecture of neural network and the recognition accuracy rate.

2 EmoEars

The block diagram of EmoEars is shown in Fig. 1. The speech signal is sampled at 16 kHz Mono and coded with 16 bits PCM. The emotional speech is firstly segmented into utterance level by using an endpoint detector based on Zero Crossing Rate (ZCR) and frame energy. The utterance speech samples are segmented into frames with 5ms frame shifts. For each frame of speech signal, we estimated F0, energy, and the first three formants.

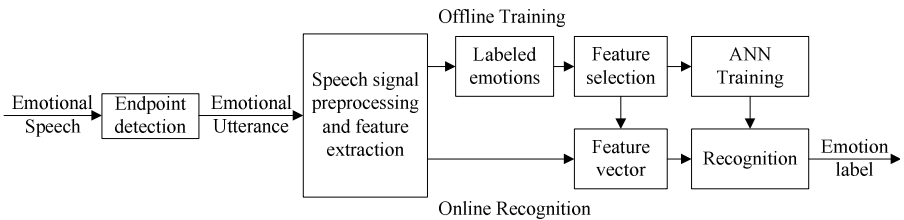


Fig. 1. Block diagram of the proposed system

For each emotional utterance, a feature vector which based on statistics computing is obtained. For the offline training, the utterances are labeled with corresponding emotion. A novel feature selection procedure is employed to find relevant features from original feature set. Neural network model is trained using the refined feature vectors of all labeled utterance. For online recognition, original feature vector from unlabeled utterance is firstly refined conduct by feature selection module. The refined feature vector for each utterance is then submitted to the neural network classifier.

2.1 Offline Model Training

The offline model training mainly deal with three tasks: collection of emotion labeled utterance which is used to train the model; acoustic feature statistics computing and selection; neural networks building and training.

2.1.1 Database

The emotional speech database includes short utterances covering four archetypal emotions and neutral emotion, namely Anger, Fear, Happy, Sad and Neutral. Non-professional speakers are selected to avoid exaggerated expression. A total of ten native Mandarin language speakers (five male and five female) are employed to speak emotional utterances. There were five text scripts are used in the recording and actors were asked to speak every script with all five emotions. The emotional mandarin speech database used in the research consisted of 697 utterances and the recording format is mono, 16-bit, 16 KHz.

2.1.2 Features and Selection

We selected the F0, energy and formant as the base features based on the previous study results [1] and our preliminary result. We also added velocity information for F0 and energy, respectively, to take the rate of speaking into account and model the dynamics of the corresponding temporal change of pitch and energy. Hence we have 12 feature streams (including velocity components). These streams were the base of feature selection and analysis.

Because emotion was expressed mainly at the utterance-level, it is crucial to normalize the feature. Although a back-end classifier working with a fixed-length feature vector is used for classification, it is also important to convert feature streams into a representative fixed-length feature vector. Therefore, some statistics (including mean, max, min, range and standard deviation) of above feature streams were calculated to get a fixed-length feature vector. We also add speed information for duration that concerned about how many words were pronounced in one second. The dimension of the feature vector, the input vector of the feature selection, was 31 for each utterance. Each feature has its own unit and samples must be normalized (mean is 0 and standard deviations is 1).

Because the accuracy rate will not increase along with the feature number and the generalization of the classifier will decrease while in high dimension space. So feature selection is necessary to achieve high recognition performance. Feature selection would tell us what features and properties of the speech are important in distinguishing emotion. We could then add more relevant features accordingly to improve classification accuracy. However, it is prohibitively time consuming to perform exhaustive search for the subset of features that give the best classification. Instead, the system uses some classical feature selection methods: PFS, SFS and SBS.

Those methods use KNN classification and leave-one-out cross validation error rate to evaluate the goodness of a feature subset. In PFS, original features were ordered by increasing error rate, thus adding a new feature from the original feature set to the selected feature set successively each time. SFS adds one feature at a time by choosing the next one that least decreases error rate, while SBS starts with all original features and sequentially deletes the next feature that least increases error rate.

2.1.3 Neural Network Architecture

Neural networks are popular for pattern recognition, and they are also adopted for the classification of emotion in speech [8]. The system provides two types of neural network architecture to train the recognition model. One type of network is actually composed of some sub-neural networks, with one network for each of five the

emotions that are examined, as shown in Fig. 2(a). This type of network is called One-Class-in-One neural Network (OCON) [9]. The feature vector (original set or selected set) of each utterance is input into each of the sub-neural networks. An output probability vector is composed of the output from each sub-network, each value in the vector representing the likelihood that the utterance corresponds to that sub-network’s emotion. Finally the decision logic determines the emotion type of the utterance based on these values.

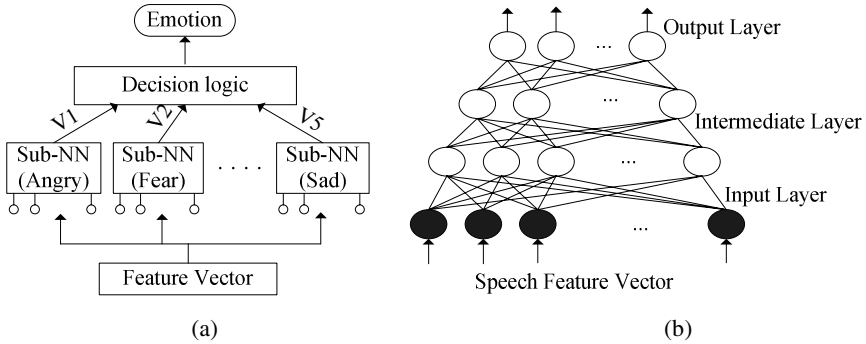


Fig. 2. (a) The OCON neural network architecture, (b) The ACON neural network architecture

The sub network has default 4 layers. The input layer takes the feature vector of utterance. There are two intermediate layers for default, and users can change the intermediate layer number and their neuron number. The output layer is a single node with an analogue value range from 0 to 1. The value denotes the probability that the emotion corresponds to the sub network. The training of each sub-neural network is done by a back-propagation algorithm, where the maximum allowable mean squared error is 0.001 for default and the training epoch is limited to 3.0×10^4 .

Another type of neural network architecture is called All-Class-in-One neural Network (ACON), as shown in Fig. 2(b). The ACON network used 4 layers for default. The input layer takes the feature vector into the network. ACON default have 2 intermediate layers, users can change the intermediate layer number and their neuron number. The output layer contains neurons which correspond to the target emotions respectively. Each neuron in output layer can output an analogue value range from 0 to 1, which denotes the probability that the emotion corresponds to the neuron. The training of the ACON is similar with OCON. The resultant output is given by selecting the emotion class where the output layer neuron has the highest value.

2.2 Online Emotion Recognition

Some existent classifiers that have been trained in advance are listed on the online speech emotion recognition stage. The information of the classifiers are displayed on the system interface, including the emotions the classifier could recognize, the number and the speakers of the training sample, the speaker’s gender etc. Users can select one proper classifier according to that information.

There is no doubt that speech energy has great effects on speech emotion recognition. It is more possible that different speaker express same emotion with different energy levels. Even one speaker may express the same emotion with different energy while his/her health condition is changed. So the energy in speech emotion expression depends on not only the emotion but also the speakers and their health condition. A recording volume and speech energy normalization procedure is employed to reduce the influence of the energy level change. Users are required to take head-phone to fix the distance of the microphone and mouth. Users must firstly modify their record volume to a proper value according to the denotation of a volume bar. With this volume value, the dB energy of no voice environment must below the threshold that the endpoint detection module assigned, and energy of neutral speech must above it. The users are required to speak some utterance with neutral emotion to measure the energy level. The system will calculate a coefficient, which is the ratio of the energy level and that of in neutral training samples. On the recognition, the speech energy will be scaled to the level of training sample using this coefficient.

After energy normalization, user can successively speak emotional utterances which are separated with a non voice pause to recognize the emotion. The endpoint detection module extracts each utterance and put it into a system queue. Feature extraction module takes the speech signal from the queue, and output the original feature vector. The original feature vector then is refined by the feature selection results which are defined in the classifier. The refined feature vector is submitted to the classifier, and it outputs the recognition result with emotion label and a cartoon symbol. The recognition procedure is always continued until users stop speaking and send stop command. One limitation in recognition stage is about words number counting. The system does not have such module to automatic compute the words number through the speech signal. Instead, the user must be required to type the text into system before speak them. Words number will be calculated through character string processing.

2.3 Implementation

EmoEars is implemented with Visual C++ 6.0 and matlab 6.5 environment. The VC interacts with matlab through matlab engine mechanism. The entire framework and user interface is written by VC. The feature extraction and selection module, and neural network module are implemented by matlab. The offline model training module generate feature selection result and trained classifier stored as *.fea and *.rg files respectively. The online recognition takes the files to run the recognition task. The system adopts wizard like form to construct the entire procedure.

3 Performance Evaluation

A comparison experiment was done between the original feature set and the refined ones to evaluate the performance of feature selection module. The database described in Section 2 was used in the experiment. An ACON neural network with two intermediate layers is employed. The maximum mean squared error was 0.005 and training epoch was limited to 3.0×10^4 . There were 209 utterances (about 40 utterances for one emotion class) were randomly selected from the database to form testing samples. The remained 488 utterances were reserved to from training samples. Some percent of the reserved

training samples was used to train the model according to a scale coefficient, which increases from 0.2 to 1 in 10 steps. In each step the training accuracy and the testing accuracy was recorded. The refined feature set selected by SFS method contains 22 features, and the original one has 31 features. The comparison result is shown in Fig. 3. It shows that the feature selection module reduces feature number without sacrificing accuracy. Even it can achieve higher generalization accuracy than the original feature set on some occasions. The training accuracy is little influenced by feature selection and the generalization accuracy is increasing along with the training sample number.

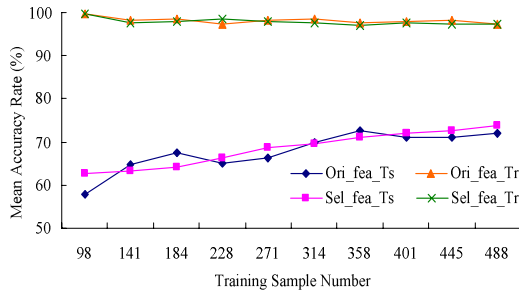


Fig. 3. The comparison of original and selected feature set. Ori = Original, Sel = Selected, Tr = Train, Ts = Test

Eight peoples (four male and four female) were recruit to evaluate the system. Five text scripts were used in the experiment and speakers were asked to speak every script with all five emotions: angry; fear; happy; sad and neutral. Utterance with every script and emotion was repeated 2 or 3 times. The classifier with ACON architecture was trained by two emotional speech corpuses. One is the existing emotional speech database for Speaker and Text Independent (SITI) testing, and the other is collect through recording every one utterance for each emotion and script from all of the 8 speakers for Speaker and Text Dependent (SDTD) testing.

The totally mean accuracy rate was 51.8% for SITI and 82.7% for SDTD in the classification of five emotions. To compare with the human recognition, the emotional utterances of the speaker were recorded. Another three peoples were recruited to recognize the emotion of the reserved emotional utterance. They should discuss the recognition result until get an agreement. The human recognition had the totally mean accuracy rate of 70%. The mean accuracy rate result from the classifier and human were grouped in gender and emotion category and shown in Fig. 4. In the SITI testing, the mean accuracy rate curve from classifier has the similar shape with the one generated by human, but they are in different performance level. The mean accuracy rate of human had 19% higher than the recognizer. In the SDTD testing, the mean accuracy rate of fear, happy and neutral emotion from classifier was close to the human recognition, whereas the other was higher than it. From the remarks of human, the poor recognition rate of angry and sad resulted from the extraordinary emotion expression. It can conclude that human recognition of speech emotion was more accuracy than the recognizer in the matter of SITI. But the accuracy rate would be low when some emotions of speech were expressed distinctively, while it can be well done by the recognizer in the matter of SDTD.

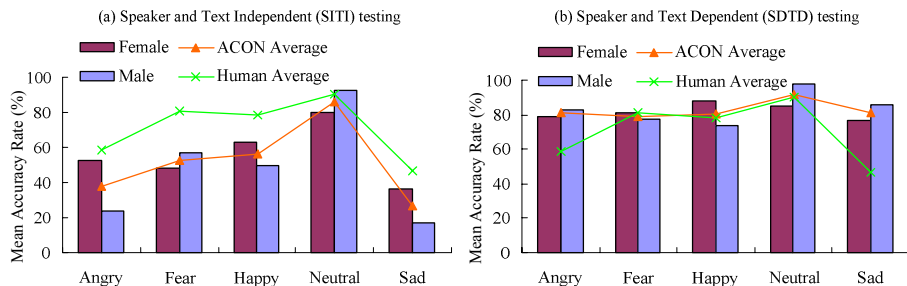


Fig. 4. Recognition result comparison by classifier and human

Because of the randomization of the network initialization, it could hardly find the optimum network configuration for emotion recognition. In addition, it is found that some emotions, such as anger or neutral, are easy to recognize, but fear and happy are difficult to recognize. The neural network with OCON architecture can optimize this limitation. It can select and change the configuration of corresponding sub-networks separately to improve the performance. Fig. 5 shows the mean accuracy rate over ACON and OCON after optimization.

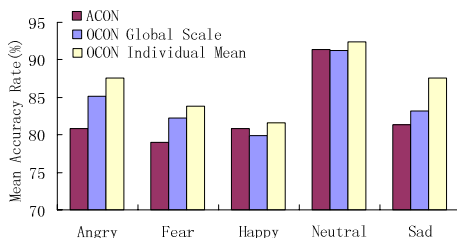


Fig. 5. The comparison of ACON and OCON

In the experiment, ACON and OCON model were all trained 10 times with different random initialization. The best ACON model in the 10 trained models could get the mean accuracy rate of 82.7%. The optimized OCON model was obtained through two methods. One is for all speaker global scale. It firstly get the mean accuracy rate of 8 speakers for each emotion using the 10 OCON models, then select the corresponding sub-networks which result in the highest mean accuracy rate of each emotion in the 10 sub-networks, and reconstruct an optimized OCON model for all speakers. The optimization total mean accuracy rate is 84.3% for global scale. The other is for individual speaker. It firstly select the corresponding sub-networks which result in the highest mean accuracy rate of each emotion in 10 sub-networks for each speaker, then reconstruct 8 individual optimized OCON networks, The mean of the 8 individual optimized OCON accuracy is 86.7%. From the Fig. 5 it can conclude that OCON architecture have better optimization performance than ACON, and it could effectively obtain optimum result for global scale and individual ones.

4 Conclusions

This paper introduces a mandarin speech emotion recognition system named EmoEars. The system uses neural network as classifier and it uses feature selection to enhance the efficiency and classification accuracy. Users can do online emotional utterance recording to create emotional speech database. Using the database, users can construct speaker-dependent emotion recognition model. The performance test indicates that feature selection module can reduce the feature number without sacrificing accuracy. In addition, the optimized network can get the best accuracy of 86.7% which beyond the 20% chances by a random hit for a sample set of 5 categories.

References

1. Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., et al.: Emotion recognition in human-computer interaction. *IEEE Signal Processing Magazine*, Vol. 18, No. 1 (2001) 32-80
2. Picard, R.W.: *Affective computing*. The MIT Press (1997)
3. McGilloway, S., Cowie, R., Douglas-Cowie, E., et al.: Approaching automatic recognition of emotion from voice: A rough benchmark. *ISCA Workshop on Speech and Emotion*, Belfast (2000)
4. Dellaert, F., Polzin, T., Waibel, A.: Recognizing emotion in speech. *Proceedings of International Conference on Spoken Language Processing* (1996) 1970-1973
5. Murray, I.R., Arnott, J.L.: Toward the simulation of emotion in synthetic speech: A review of the literature on human vocal emotion. *Journal of the Acoustical Society of America*, Vol. 93, No. 2, (1933) 1097-1108
6. Lee, C.M., Narayanan, S., Pieraccini, R.: Recognition of negative emotions from the speech signal. *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding* (2001) 240-243
7. Kwon, O.W., Chan, K., Hao, J., Lee, T.W.: Emotion recognition by speech signals. *Proceedings of EUROSPEECH* (2003) 125-128
8. Nicholson, J., Takahashi, K., Nakatsu, R.: Emotion recognition in speech using neural networks. *Neural Comput & Applic* (2000) 290-296
9. Marker, J.M., Gray, A.H.: *Linear prediction of speech*. Springer-Verlag (1976)

User Identification Using User's Walking Pattern over the ubiFloorII

Jaeseok Yun¹, Woontack Woo², and Jeha Ryu¹

¹ Dept. Mechatronics, Gwangju Institute of Science and Technology, S. Korea
{zapper, ryu}@gist.ac.kr

² Dept. Info. and Comm., Gwangju Institute of Science and Technology, S. Korea
woo@gist.ac.kr

Abstract. In this paper, we propose ubiFloorII, a novel floor-based user identification system to recognize humans based on their walking pattern such as stride length, dynamic range, foot angle, and stance and swing time. To obtain users walking pattern from their gait, we deployed photo interrupter sensors instead of switch sensors used in ubiFloorI. We developed a software module to extract walking pattern from users' gait. For user identification, we employed neural network trained with users' walking samples. We achieved about 96% recognition accuracy using this floor-based approach. The ubiFloorII system may be used to automatically and transparently identify users in home-like environments.

1 Introduction

With the availability of diverse sensors and computing power, it is becoming possible to provide the users with personal location-based services. For providing convenient services to users, reliable person identification through automatic, transparent, and often remote means is a must. Surrogate representations of identity such as password have been successful in conventional computers. However, in ubiquitous computing environments where computing resources exist everywhere, it is necessary to perform user identification through various means.

One approach in user identification is RFID (Radio Frequency IDentification) system [1, 2, 3]. RFID system has the potential to accurately recognize humans and is robust against environmental factors around an individual like light intensity. A disadvantage of RFID system is that individuals must always carry or attach sensors to their body, and may lose it.

Biometrics, which refers to automatic recognition of people based on their distinctive anatomical (e.g., face, fingerprint, iris, retina, hand geometry) and behavioral (e.g., signature, gait) characteristics, can provide automatic, secure, and user-friendly person identification solutions. This is because biometric identifiers cannot be shared or misplaced, and they intrinsically represent the individual's bodily identity [6]. However, the currently available biometrics systems have not yet carried out automatic and transparent human recognition because of several limitations [4, 5]. For example, the accuracy of camera-based systems is susceptible to the environmental factors like obstacles, shadow, and light intensity.

Gait recognition means to recognize an individual based on distinctive personal characteristics in gait which refers to the individual’s style of walking. Much evidence to support the utility of gait in recognizing people has been reported in other domains such as biomechanics, mathematics, and psychology which suggest that gait is unique [7]. Gait recognition systems can be divided into two categories: (i) vision-based [7, 8] and (ii) floor-based approach [9, 10]. Although vision-based approach has many advantages, extracted features are sensitive to environmental factors and individual’s privacy can be compromised because of camera surveillance. On the contrary, floor-based approach preserves privacy and is available without regard to light. Examples of this approach include active floor [9] and smart floor [10]. These floors consist of load cells and tiles. Features are extracted from the vertical GRF (Ground Reaction Force) signals derived from the individual’s footsteps. Experimental results showed respectively 91% and 93% correct footstep identification with 15 subjects. However, these systems need expensive load cells to get user’s stepping pattern (i.e., GRF signal). To track a resident, living room should be equipped with a large number of tiles. Thus, cost of the recognition system would be high.

To overcome the shortcomings of previous systems, we proposed the ubiFloorI, peculiar carpet where 144 switch sensors are uniformly arranged [11]. We extracted walking pattern including stride length, dynamic range, and foot angle from switch sensors, and used neural network to identify unknown walking samples. The experiments showed 90% recognition with 10 subjects. An advantage of the ubiFloorI is that it adopts inexpensive switch sensors instead of load cells, thus cost of system is reduced. This is because walking pattern is used instead of stepping pattern. Nevertheless, we could not use time information like stance and swing time, and stepping pattern like transitional footprint because of low resolution of the floor. Moreover, the centralized architecture of data acquisition was unfavorable for maintenance and extension of the system.

In this paper, we propose the ubiFloorII, a new floor-based user identification system to recognize humans based on their stepping pattern as well as walking pattern. To narrow the scope of the paper, we leave out the detailed organization

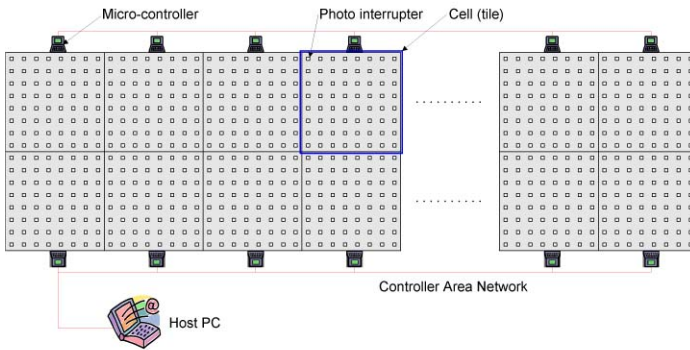


Fig. 1. Overall structure of the ubiFloorII system



Fig. 2. Implemented ubiFloorII

of the ubiFloorII system [12]. Fig. 1 shows overall structure of the ubiFloorII system. The ubiFloorII consists of a 12×2 array of wooden tiles. Each wooden tile measures 30cm square and 64 photo interrupter sensors are uniformly arranged in each tile. A micro-controller is responsible for data acquisition from a corresponding tile and transmits obtained information to the host PC through CAN (Controller Area Network) cable. Then, the host PC extracts user's walking features from received data and recognizes the user with well-trained neural network. Fig. 2 shows our implemented ubiFloorII system.

2 User Identification with the ubiFloorII

2.1 Walking Pattern Extraction

The software modules we have developed for extracting walking pattern from data sets fall into two categories: (i) step-feature extraction and (ii) walking-feature extraction.

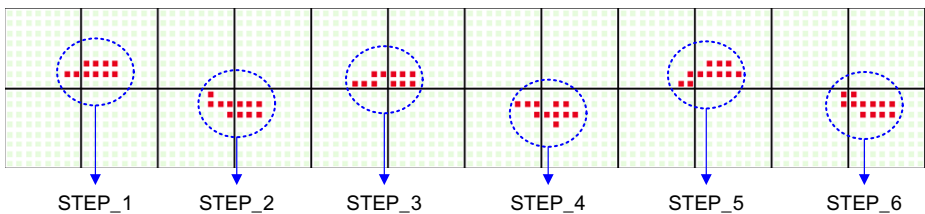


Fig. 3. An example of searching footprints

The step-feature extracting software is used to search all footprints in the data sets received during one walking over the ubiFloorII as shown in Fig. 3. We created the 8×4 footprint model to cover all probable footprints and chose three step-features as follows.

- The X index of the backmost sensor in a footprint
- The Y index of the backmost sensor in a footprint
- The footprint model of a footstep

Fig. 4 displays the footprint model extracted from the STEP_1 footprint in Fig. 3. As shown in Fig. 4, the backmost sensor in a footprint becomes the seed sensor and the other features can be extracted based on the seed sensor. We

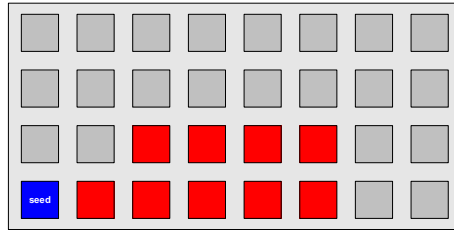


Fig. 4. An example of footprint model

extracted walking-features using the feature values obtained from step-feature extraction as inputs of walking-feature extracting software. We adopted seven walking-features as follows.

- FX : physical X coordinate of the backmost sensor in a footprint
- FY : physical Y coordinate of the backmost sensor in a footprint
- com_FX : compensated X coordinate based on footprint model
- com_FY : compensated Y coordinate based on footprint model
- $nSensor$: the number of pressed sensors in a footprint
- $fStart$: heel-strike time of a footstep
- $fEnd$: toe-off time of a footstep

FX and FY represent the physical X and Y coordinates of the seed sensor in a footprint with the bottom-left corner of the ubiFloorII as origin. Coordinates com_FX and com_FY represent the center of a footprint based on footprint model as shown in Fig. 5. Practically, com_FX and com_FY comprehend user’s stride length, dynamic range, and foot angle. $fStart$ and $fEnd$ imply user’s stance and swing time in walking. Finally, to create input vectors to our neural network, we need to make the sequences of each walking-feature in terms of footsteps such as $[com_FX1, com_FX2, com_FX3, \dots]$.

2.2 User Identification

We used multilayer perceptron networks for identifying individuals based on extracted walking features. Our neural network consists of three layers as shown in Fig. 6: (i) input layer N_1 with P_1 neurons, (ii) hidden layer N_2 with P_2 neurons, and (iii) output layer N_3 with P_3 neurons. The input layer N_1 represents the

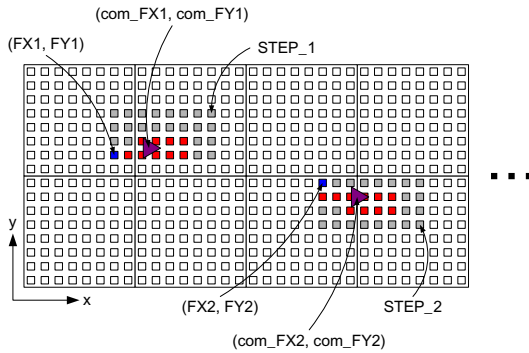


Fig. 5. Walking feature extraction

individuals’ walking features space. In the output layer N_3 , we can choose the index number of the output node with maximum output values as the user’s identification number.

$$User_Number = \max(O_1, O_2, \dots, O_M) \tag{1}$$

where O_i denotes the output values of i th node, and M is the number of users. For our neural networks, the transfer function for the hidden layer is tangent-sigmoid and the transfer function for the output layer is pure-linear.

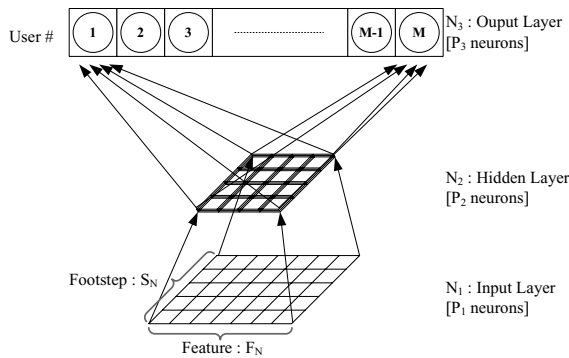


Fig. 6. Structure of neural network for user identification

3 Experiments

3.1 Experimental Population and Conditions

We had the following related hypotheses for walking samples we gathered.

- The subjects maintain walking pattern as regularly as possible.

- The target of our proposed system is home where the size of family is smaller than 10.
- While walking, the subjects listen to smooth music so that the variations of walking pattern are practically reduced.

We gathered walking samples for 10 subjects. They were aged between 27 and 35, and were 165 to 180cm tall. For each subject, we gathered 50 walking samples with the ubiFloorII. In total, we collected 500 walking samples. Depending on users' stride length, it took them five or six footsteps to cross the ubiFloorII. Therefore, we only considered first five footsteps (STEP_1 - STEP_5).

3.2 Walking Feature Sets

To verify the dominant walking features we used five feature sets as inputs to the network. Table 1 shows the combinations of features. In Case 1, coordinates FX and FY are the inputs to the network. This case is used as the standard for evaluating the results with the other feature sets.

Table 1. Classification of feature sets

Case	Features sets	# of inputs
1	FX , FY	10
2	com_FX , com_FY	10
3	com_FX , com_FY , and $nSensor$	15
4	com_FX , com_FY , $fStart$, and $fEnd$	20
5	com_FX , com_FY , $fStart$, $fEnd$, and $nSensor$	25

3.3 Results

We first demonstrate how the number of hidden nodes influences the performance of the neural network. In an effort to decide the optimal number of hidden nodes, we ran an experiment whereby we increased the number of hidden nodes while keeping other parameters fixed, and observed the resulting recognition accuracy. An experimental result with com_FX , com_FY , $fStart$, and $fEnd$ features are shown on the left side in Fig. 7. It shows that about 40 hidden nodes are enough for about 95% recognition rate. We also had experiments to decide $epoch$ and $goal$. The right side in Fig. 7 shows that after 1800 epoch, Mean Square Error is smaller than 10^{-4} and this value will be set to the goal.

We present the results of test with our feature sets and a comparative analysis in Table 2. In this experiment, the recognition accuracies were obtained by averaging 10 simulation results while changing the seed value. The seed value determines the initial values of weights and biases of the network. First, we can note that the compensation procedure for com_FX and com_FY results in about 10% improvement in recognition accuracy. Considering Case 2 and 3, Case 3 is worse than Case 2 (i.e., without $nSensor$) because $nSensor$ information already

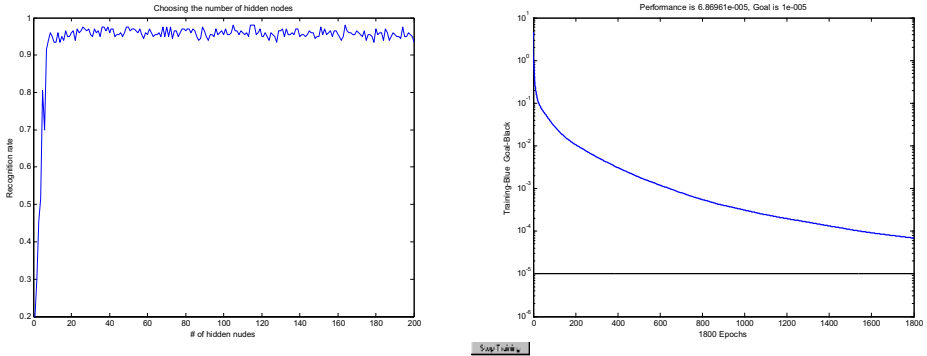


Fig. 7. Results for deciding the number of hidden nodes, epoch, and goal

influenced the compensation procedure for com_FX and com_FY . This result is very similar to that in ubiFloorI. We were able to achieve about 96% recognition accuracy when com_FX , com_FY , $fStart$, and $fEnd$ features were used. Therefore, we can note that stance and swing time is also dominant feature for user identification.

Table 2. Comparison of recognition accuracy

Case	Features sets	Recog. rate(%)
1	FX, FY	80.75
2	com_FX, com_FY	89.05
3	com_FX, com_FY , and $nSensor$	86.85
4	$com_FX, com_FY, fStart$, and $fEnd$	96.20
5	$com_FX, com_FY, fStart, fEnd$, and $nSensor$	95.20

We now compare our proposed method with previous floor-based approaches in terms of the recognition accuracy and system architecture. All the floor-based identification systems have about same recognition accuracy. Especially, if user identification would be performed using the stepping pattern as well as walking pattern, the performance of recognition could be definitely improved. Considering system architecture, modularized architecture of the ubiFloorII makes the floor favorable for extension and maintenance. Consequently, our proposed identification method has very wide applications in home-like environments.

4 Conclusion and Future works

In this paper, we have presented a floor-based approach to identify people by their walking pattern. We have designed the ubiFloorII system to provide a

way to automatically and transparently recognize users. In the ubiFloorII system, photo interrupter sensors are used to measure the users' walking pattern instead of switch sensors in ubiFloorI. Our achievable accuracy of about 96%, although not wholly successful, is nonetheless enough for common applications in home-like environments. We are further investigating appropriate ways to improve recognition accuracy by combining stepping pattern such as the array of transitional footprints from heel-strike to toe-off.

References

1. Harter, A., Hopper, A.: A distributed location system for the active office. *IEEE Network Mag.* 8 (1) (1994) 62–70
2. Harter, A., Hopper, A., Steggle, P., Ward, A., Webster, P.: The anatomy of a context-aware application. *Proc. ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99)*, Seattle, United States, Aug. (1999) 59–68
3. Bahl, P., Padmanabhan, V.N.: RADAR: An in-building RF-based user location and tracking system. *Proc. IEEE INFOCOM'00*, vol. 2, Tel Aviv, Israel, Mar. (2000) 775–784
4. Jain, A.K., Hong, L., Pankanti, S.: Biometric identification. *Commun. ACM*, 43 (2) (2000) 90–98
5. Jain, A.K., Ross, A., Prabhakar, S.: An introduction to biometric recognition. *IEEE Trans. Circuits Syst. Video Technol.* 14 (1) (2004) 4–20
6. Jain, A.K., Pankanti, S., Prabhakar, S., Hong, L., Ross, A.: Biometrics: A grand challenge. *Proc. International Conference on Pattern Recognition*, vol. 2, Cambridge, UK, Aug. (2004) 935–942
7. Nixon, M.S., Carter, J.N.: Advances in automatic gait recognition. *Proc. IEEE Conference on Automatic Face and Gesture Recognition*, Seoul, S. Korea, May (2004) 139–144
8. Kale, A., Sundaresan, A., Rajagopalan, A.N., Cuntoor, N.P., Roy-Chowdhury, A.K., Krugar, V., Chellappa, R.: Identification of humans using gait. *IEEE Trans. Image Process.* 13 (9) (2004) 1163–1173
9. Addelee, M.D., Jones, A., Livesez, F., Samaria, F.: The ORL active floor. *IEEE Personal Commun. Mag.* 4 (5) (1997) 35–41
10. Orr, R.J., Abowd, G.D.: The smart floor: A mechanism for natural user identification and tracking. *Extended abstracts Conference on Human Factors in Computing Systems*, Hague, Netherlands, Apr. (2000) 275–276
11. Yun, J., Lee, S., Woo, W., Ryu, J.: The user identification system using walking pattern over the ubiFloor. *Proc. International Conference on Control, Automation, and Systems*, Gyeongju, Korea, Oct. (2003) 1046–1050
12. Yun, J., Ryu, J., Woo, W.: ubiFloorII: User tracking and identification system using photo interrupters. *Proc. International Conference on Human-Computer Interaction*, Las Vegas, United States, July (2005)

Evolving RBF Neural Networks for Pattern Classification

Zheng Qin^{1,2}, Junying Chen¹, Yu Liu¹ and Jiang Lu¹

¹ Department of Computer Science, Xian JiaoTong University,
Xian 710049, China
vcjy@169cnc.com

² School of Software, Tsinghua University, Beijing 100084, China

Abstract. When a radial-basis function neural network (RBFNN) is used for pattern classification, the problem involves designing the topology of RBFNN and also its centers and widths. In this paper, we present a particle swarm optimization (PSO) learning algorithm to automate the design of RBF networks, to solve pattern classification problems. Simulation results for benchmark problems in the pattern classification area show that the PSO-RBF outperforms two other learning algorithms in terms of network size and generalization performance.

1 Introduction

RBF neural networks are widely used to perform pattern classification tasks due to their simple network structures and fast learning algorithms. When applying RBF networks to pattern classification, the principal difficulties lie in determining the numbers of basis functions and its center positions and widths. It is because that once the centers and the widths have been fixed, the weights between the hidden and the output layers can be calculated analytically using singular value decomposition [1] or any algorithm suitable to solve linear algebraic equations.

One class of traditional training algorithms for RBF networks starts with a predetermined network structure. Then the centers and widths of the RBFNN are trained. Several approaches have been proposed to determine the centers. In the work of Poggio and Girosi [2], the number of the centers equals to the number of training patterns and the centers are coincident with the training patterns. This produces a relatively large network, even for relatively simple problems. Improved methods for selecting centers to provide an adequate sampling of the input training patterns include the orthogonal least squares algorithm [3], K-means [4], and so on. In [5], input vectors are clustered based on K-means method, and the center of each cluster becomes a basis function center. The widths of the basis functions can be determined using various K-nearest neighbor heuristics. These methods require designers to fix the number of basis functions in advance according to prior knowledge. However the optimal number of neurons cannot be decided in advance. Predetermined architecture is either insufficient or over complicated. Genetic algorithms have been employed to automatically evolve the structure of neural network in [6]. The network configuration is formed as a subset selection problem from the input training data. The selected subset forms the centers of the basis functions. However, it has been proven that the best

center locations may not necessary be located inside the input training data [7]. In [8], Particle Swarm Optimization algorithms (PSOs) were used to evolve RBFNN structure and parameters for binmap classification. The network was initialized with a single basis function and one neuron was added if the overall performance of the swarm didn't improve. If the required number of hidden neurons is large, the method needs many trails in getting the appropriate number of basis functions.

In this paper, PSO algorithmic approach was used to evolve both RBFNN structure and parameters of the basis functions in a different way. The number of basis functions was dynamically adjusted according to the particles in the swarm and there was no need of initializing networks with one hidden neuron. Given the training patterns, the PSO searches to find a excellent configuration of RBFNN that has high classification rate as well as strong generalization ability for the given classification problems. The algorithm of PSO used to evolve RBF networks in this paper is referred to as PSO-RBF algorithm.

The rest of the paper is organized as follows. Section 2 presents the architecture of the RBF networks used. Section 3 describes the PSO learning algorithm. In Section 4, PSO-RBF algorithms are discussed. Parameter selection and experimental results are presented in Section 5. Finally, the conclusions are given in Section 6.

2 RBF Neural Network

Radial Basis Function neural networks were introduced into the neural network literature by Broomhead and Lowe [4], which are motivated by observation on the local response in biologic neurons.

RBF network is a three-layer network. The input layer consists of n units which represent the elements of the input vector x . The hidden layer is composed of some basis functions that execute non-linear mapping. The activation of a hidden neuron is determined by computing the distance (usually by using the Euclidean norm) between its center vector and the vectors which are yielded by the activation of the input layer. The activation of a neuron in the output layer is determined by computing the weighted sum of outputs of hidden layer. The RBFNN form with linear combination of Gaussian functions is shown in the following.

$$o_i(x) = \sum_{k=1}^N w_{ik} \exp\left\{-\frac{\|x - c_k\|^2}{2\sigma_k^2}\right\}, i = 1, 2, \dots, m. \quad (1)$$

Where $\|\dots\|$ represents Euclidean norm, c_k , σ_k and w_{ik} are the center, the width of the k -th neuron in the hidden layer and the weights in the output layer respectively, N is the number of neurons in the hidden layer.

RBFNN are universe function approximators if the centers and widths are set appropriately. The number of basis functions affects the performance of RBFNN, networks with too many hidden units overfitting training data and having poor predictive ability while ones with insufficient hidden units having poor approximation power. The width narrower than necessary will lead to overfitting, and width wider than necessary can result in poor approximation ability and give even worst results.

3 PSO Algorithm

PSO is a new population-based evolutionary computation technique [9]. Particle swarms explore the search space through a population of particles. The particle evolves by adjusting its position at diverse speed iteratively. The position of the i -th particle at t iteration is represented by $X_i^{(t)}=(x_{i1}, x_{i2}, \dots, x_{iD})$, and its velocity is represented by $V_i^{(t)}=(v_{i1}, v_{i2}, \dots, v_{iD})$. The movement of the particle is not only influenced by the particle's own memories but also the memories of its neighborhood. The velocity update equation (2) and position update equation (3) are described as follows:

$$V_i^{(t+1)} = w * V_i^t + c_1 * rand() * (P_i - X_i^{(t)}) + c_2 * rand() * (P_g - X_i^{(t)}) . \quad (2)$$

$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t)} . \quad (3)$$

Where P_i is the position with the best fitness value visited by the i -th particle, P_g is the position with the best fitness found by all particles, w is inertia weight which balances the global exploitation and local exploration abilities of the particles, c_1 and c_2 are acceleration constants, $rand()$ are random values between 0 and 1. The velocities of the particles are limited in $[Vmin, Vmax]^D$. If smaller than $Vmin$, an element of the velocity is set equal to $Vmin$, if greater than $Vmax$, and then set equal to $Vmax$. In general, each dimension of function optimization problems used to verify PSOs' abilities has range with the same limits in the problem space. However for our evolving RBFNN problem, each dimension of the patterns has range with different limits. First normalize features in patterns so that their values lie within the same range. In this way, each component of the velocity of the particle in the swarm has range with the same limits.

4 PSO Algorithms for RBFNN Configuration

When PSO used to solve pattern recognition problems, each particle is encoded as a combination of variables, which represent the relevant information of RBF networks in this study. A swarm of particles represents possible space solutions. Each particle is a potential solution to the problem and evaluated in every iteration based on its fitness. Through moves of the particles and new particles generated in next iteration, the better solutions are gradually achieved and after the predefined iterations the final best particle is considered to be the solution to the problem.

4.1 Denoting RBF Network Architecture

A swarm of particles is manipulated and each individual represents a possible solution to the problem. Each encoded particle denotes a kind of RBF architecture. As shown in Equation 4, each particle is encoded into a real-value vector which includes H (H is supposed as the maximal number of basis functions) parts, each part involving the information of one basis function.

$$[(f_1, c_1, \sigma_1), (f_2, c_2, \sigma_2), \dots, (f_H, c_H, \sigma_H)]. \tag{4}$$

$f_i, c_i, \sigma_i (i = 1, 2, \dots, H)$ represents whether or not the i -th basis function is involved in the network, the center of the i -th basis function, the width of the i -th basis function respectively. If f_i is positive, the i -th basis function is involved in the network, otherwise not involved.

4.2 Fitness Evaluation

Fitness designing technique is used to optimize network performance and network complexity. RBF networks with good configurations should have fewer hidden units and high prediction accuracy. To balance the network performance and network complexity, Akaike’s information criterion (AIC) [6] was adapted. The fitness of the RBF network was formulated as equation (5).

$$E = N_t \times \log\left(\frac{1}{N_t} \sum_{n=1}^{N_t} \|y_n - o_n\|^2\right) + \lambda \times N_h. \tag{5}$$

Where N_t is the number of training patterns, y_n, o_n are the desired output and network output for pattern n respectively, N_h is the number of hidden units involved in networks, λ is a parameter that balances network performance and network complexity.

4.3 PSO-RBF Algorithm

The particle swarm optimization to evolve RBF networks for pattern classification is presented in the following:

1. Initialize swarm of N particles. Each particle defines a network and the associated centers and widths. Set the number of iterations as *MaxIteration*. Set *count*=0.
2. Decode each particle into a network. Compute the connection weights between the hidden layer and the output layer. Compute the fitness of each particle.
3. Update P_i for each particle and P_g for whole swarm.
4. Update the velocity of each particle according to formula (2). Limit the velocity in $[Vmin, Vmax]^D$.
5. Update the position according to formula (3).
6. Set *count*=*count*+1; if *count* < *MaxIteration*, go to 2; otherwise, Terminate the algorithm.

5 Experiments

5.1 Benchmark Datasets

Four datasets were obtained from UCI repository of Machining Learning databases [10] for RBF executing pattern classification tasks.

IRIS Data. IRIS data have extensively been used for evaluating the performance of pattern classification algorithms. This dataset contains 150 patterns of 4 input variables that belong to three classes. Each class contains 50 patterns. One of the three classes are well separated from the other two, which are not easily separable due to the overlapping of their convex hulls. The 150 patterns were randomly split into two sets, 2/3 for training patterns and 1/3 for test patterns.

Thyroid Gland Data. Thyroid data was ever used to test the potential of pattern classification in medical decision-making area. This dataset contains 215 patterns that belong to three classes, 150 instances for class euthyroidism, 35 for class hyperthyroidism and 30 for class hypothyroidism. Each pattern includes 5 input variables based on a complete medical record. The patterns in every class were randomly split into two sets, 2/3 for training patterns and 1/3 for test patterns.

Ionosphere Data. Ionosphere are radar data collected by a system in Goose Bay, Labrador. This data set contains 351 patterns of 34 input variables that belong to two classes. The 351 patterns were randomly split into two sets, 2/3 for training patterns and 1/3 for test patterns.

Wine Data. Wine data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. This data set contains 178 patterns of 13 input variables that belong to three classes. The patterns were randomly split into two sets, 2/3 for training patterns and 1/3 for test patterns.

5.2 Experimental Setup

The parameters of the PSO algorithm were set as follows: weight w decreasing linearly between 0.9 and 0.4, learning rate $c_1 = c_2 = 2$ for all cases. The population size used by PSO-RBF was constant. The algorithm stopped when a predefined number of iterations has been reached. Normalize patterns by dividing the maximum value of each feature so that each feature may have the same influence on the cost function. The parameter λ was set 4 to balance the classification performance and generalization ability of the evolved networks.

For RBFNN, two initialization techniques were adapted to improve the performance of PSO-RBF algorithm. One is the method to initialize centers, and the other is the method to initialize widths. Once finished the long set of experiments, values selected for parameters were shown in tables 1.

Table 1. Execution parameters for PSO-RBF

Parameter	Value
Population Size	20
Iterations	5000
Vmax	1
Width Initialization	IW_AVG_DIS
Center Initialization	IC_RND

Furthermore, centers were initialized according to the IC_RND method, which uses randomly selected patterns from the training set as RBF centers. On the other hand, widths were initialized by using the IW_AVG_DIS method. This method computes the Euclidean distance from the center of the neuron being initialized to the rest of neurons in the net, and sets the width as average distance computed by previous operations.

5.3 Experimental Results

Evaluations of PSO-RBF on classification tasks were developed by using four well-known real databases, the Iris plants, Thyroid gland, radar Ionosphere and Wine data.

For comparison, the traditional approach based on K-means and K-nearest neighbor methods (named KM-RBF) was executed. KM-RBF used K-means algorithm and K-nearest neighbor heuristic to find centers and widths of the basis functions respectively and then used least mean square (LMS) to find the connection weights between hidden layer and output layer. The number of clusters in input training patterns was experientially set, two values considered for each dataset to relieve the risk of network being insufficient or over complicated.

In additional, the standard algorithm used in MATLAB TOOLBOX (named STD-RBF) was also compared with our PSO-RBF. STD-RBF started the process with an empty subset to which one center (taken from the set of input vectors) was added at a time until it met the specified mean squared error goal. Error goal was set 0.05 times the number of training patterns for Thyroid dataset and set 0.1 times the number of training patterns for Iris, Ionosphere and Wine. σ was set 1 for all four datasets.

Each algorithm has been run 20 times, the averaged values for every algorithm taking into account only the best individual created in every run. *Train Accuracy*, *Test Accuracy*, *Network Size* were used to evaluate the classification performance of the algorithm. Tables 2, 3, 4 and 5 listed the comparative results on Iris, Thyroid, Ionosphere, and Wine datasets by means of KM-RBF, STD-RBF, and PSO-RBF respectively.

Table 2. Comparative results on Iris dataset

Algorithm	Train Accuracy	Test Accuracy	Network Size
KM-RBF	0.9578	0.9646	5
	0.9627	0.9500	10
STD_RBF	0.9770	0.9280	4
PSO-RBF	0.9901	0.9812	4

Table 3. Comparative results on Thyroid dataset

Algorithm	Train Accuracy	Test Accuracy	Network Size
KM-RBF	0.9375	0.9437	5
	0.9667	0.9535	10
STD_RBF	0.9867	0.9208	27
PSO-RBF	0.9790	0.9472	5

Table 4. Comparative results on Ionosphere dataset

Algorithm	Train Accuracy	Test Accuracy	Network Size
KM-RBF	0.7508	0.7538	5
	0.7882	0.7863	10
STD_RBF	0.9338	0.7530	71
PSO-RBF	0.9675	0.9419	3

Table 5. Comparative results on Wine dataset

Algorithm	Train Accuracy	Test Accuracy	Network Size
KM-RBF	0.9608	0.9621	5
	0.9700	0.9724	10
STD_RBF	0.9891	0.9712	6
PSO-RBF	1	0.9790	5

By comparing the results, it is easy to see STD-RBF found the network with large size and KM-RBF was dependent on the predefined network size. However, the networks found by PSO-RBF have less hidden units and high classification accuracy than those found by the other two algorithms.

In our implementation, each iteration of the PSO-RBF required the computation of the output weights of 20 networks and the evaluation of their fitness values. Hence, the algorithm required excessive training time. However, the training process is often off-line work and the time cost doesn't influence practical use.

6 Conclusions

This paper shows PSO-RBF, an algorithm that evolves RBFNN, applied to the task of solving classification problems. PSO-RBF automatically determines the configuration of RBFNN. Thus, PSO-RBF finds the size of the network and the parameters that configure each neuron: center and width of its basis function. Results have been compared with others in literature showing that PSO-RBF can be used to efficiently configure RBFNN, since it obtains networks with high generalization power and size smaller or similar to other methods.

Acknowledgements

This work was supported by the national grand fundamental research 973 program of China (No.2004CB719401).

References

1. W.H.Press, S.A.Teukolsky, W.T.Vetterling, and B.P.Fkannery.: Numerical Recipes in C, 2nd.edition. Cambrige University Press (1992)
2. T. Poggio and F. Girosi.: Network for Approximation and Learning. Proc. IEEE, 78(9), (September 1990) 1481--1497

3. S. Chen, FN Cowan, and PM Grant.: Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks. *IEEE Trans. Neural Networks*, vol. 2, No. 2, (1991) 302-309
4. Broomhead, DS., Lowe, D.: Multivariable Functional Interpolation and Adaptive Networks. *Complex System*, Vol. 2, (1988) 321-355
5. Moody, J. Darken, CJ.: Fast Learning in Networks of Locally-tuned Processing Units. *Neural Computation*, Vol.1, (1989) 281-294.
6. SA Billings, GL Zheng.: Radial Basis Function Network Configuration using Genetic Algorithms. *Neural Networks* (1995) 877-890
7. Mak, M.W., Cho, K.W.: Genetic Evolution of Radial Basis Function Centers for Pattern Classification. *International Conference on Neural Networks, USA* (1998) 669-673
8. E Miguelanez, AMS Zalzal, P Tabor.: Evolving Neural Networks using Swarm Intelligence for Binmap Classification. *Proceedings of the 2004 IEEE Congress on Evolutionary Computation*, (2004) 978-985
9. J. Kennedy, R.C. Eberhart.: Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks, Piscataway, NJ* (1995) 1942-1948
10. Blake, C., Merz, C.J.: UCI Repository of machine learning databases. <http://www.ics.uci.edu/~mllearn/MLRepository.html>. (1998)

Discrimination of Patchouli of Different Geographical Origins with Two-Dimensional IR Correlation Spectroscopy and Wavelet Transform

Daqi Zhan¹, Suqin Sun¹, and Yiu-ming Cheung²

¹ Key Laboratory of Bioorganic Phosphorus Chemistry & Chemical Biology (Ministry of Education), Department of Chemistry, Tsinghua University, Beijing 100084, China
sunsq@chem.tsinghua.edu.cn

² Department of Computer Science, Hong Kong Baptist University, Hong Kong
ymc@comp.hkbu.edu.hk

Abstract. Patchouli is a common used traditional Chinese herbal medicine for flatulence or vomit with a long history in China and some other Asian countries. Patchouli of different geographical origins usually have different curative effects. In order to evaluate their qualities, it's very necessary to discriminate them. Our objective of this study is to develop a nondestructive and accurate identification method. The results in this paper showed that it's difficult to use conventional infrared spectra and second derivative spectra directly to distinguish them. But it's quite easy to use two-dimensional infrared (2D IR) correlation spectra, especially after applying wavelet transform process, to discriminate them. The resolution of the 2D IR spectrum is improved obviously after wavelet transform process, more peaks appear and the peaks become quite clear and separate. The differences of the 2D IR spectra become rather remarkable. In this way, it's not difficult to discriminate the patchouli of different geographical origins. This will be a nondestructive, economical and rapid way to distinguish complicated mixture like traditional Chinese medicines. Combined 2D IR and wavelet transform, Fourier transform infrared spectroscopy would become more powerful in analysis and discrimination.

1 Introduction

Infrared (IR) spectrum contains abundant structural information. And IR spectroscopy has been the classic analytical method for organic compound structure. Recently, Fourier transform infrared (FT IR) spectroscopy develops quickly due to its low noise, rapid speed, high repeatability, easy operation, low expense and so on.

Associated with other sciences just like mathematics or computer, or with other techniques such as two-dimensional correlation analysis, FT IR becomes more and more useful in the field of evaluating Chinese herbal qualities. For Chinese herbals, FT IR can be used to distinguish the herbals whether true or false, good or bad, or to distinguish the geographical origin or growth age [1]. One of the great advantages of FT IR to distinguish herbals is that the samples don't need to be preprocessed therefore this method is nondestructive. Another advantage is that the spectrum is actually the

reflection of the herbal as a whole [2]. This is consistent with traditional Chinese medicine theory.

This paper studies on how to discriminate patchoulis of different geographical origins. As we know, patchouli [*Pogostemon cablin* (Blanco) Benth.] is one of the famous Southern Medicines in China. In fact, patchouli is not only the main material in more than 30 traditional Chinese medicines, but also an important material in perfumery industry. In China, according to their geographical origins, patchouli is sorted into four kinds, Nanxiang, Paixiang, Zhanxiang, and Zhaoxiang. Based on clinic curative effects and traditional experiences, it is deemed that in quality Paixiang is the best, and Zhaoxiang is also good, while Zhanxiang and Nanxiang are relatively not so good. Therefore it's very important and necessary to discriminate the patchoulis of different geographical origins.

Although in many cases FT IR can distinguish the herbals quite efficiently, sometimes it become a little poor, for example, to distinguish patchoulis of different geographical origins. Therefore, it requires some techniques to improve the resolution of the spectrum so as to get more information to distinguish the samples. Wavelet transform is a powerful tool to analyze signals especially for signals that have that have non-stationary variations. It has been widely used to denoise and improve resolution in many scientific fields. So our study is concerned with the application of wavelet transform to 2D IR, wish to enhance the performances of 2D IR spectrum, and explore the possibility to distinguish the very similar complicated mixtures like patchoulis of different geographical origins.

1.1 Wavelet Transform and 2D IR

Wavelet analysis has been a powerful analytical tool in many scientific and practical fields, such as quantum mechanics, signal analysis, image processing and so on. The main reason is that wavelet analysis has many particular characteristics. For example, it can analyze signals at different scales or resolutions, which is called multiresolution. And it also can localize signals in both frequency and time domains. Compared with Fourier analysis, which uses infinite basic functions sine and cosine, wavelet analysis uses basic functions that are only nonzero in limited space or time. Therefore, wavelets can well express the signals that have non-stationary variations [3]. In FTIR spectra, sharp peaks always exist, so it would be better to process the data with wavelet transform than other methods.

When use wavelet transform, signals can be decomposed into several levels. At each level, two sets of coefficients are obtained. One is for approximation coefficients and the other is for detail coefficients. After decomposition, the approximation and detail coefficients can be processed according to special need, such as signal denoising, image compression and so on. At last, to obtain the processed signal, reconstruction with the coefficients is needed, which is also called inverse wavelet transform [4].

Two-dimensional (2D) correlation spectroscopy has proven to be powerful in unraveling complicated vibrational spectra of biological, polymeric, or industrial materials [5][6]. Two-dimensional infrared correlation spectroscopy was developed from generalized 2D correlation spectroscopy, whose concept was described in 1986 by Noda [7]. It can base on many kinds of perturbations, like temperature, concentration, pressure, magnetic field, electric field, etc. [7]. This technology has been use in various

fields, such as FTIR spectroscopy, fluorescence spectroscopy, and Raman spectroscopy and so on [8][9]. When used in FTIR, it is called 2D IR. 2D IR well improves the resolution of the IR spectrum, and can be used to discriminate rather similar complicated mixture systems like traditional Chinese medicine [10]. In common use, we should obtain a set of dynamic spectra first, then use correlation to analyze the spectra.

Many interesting information can be obtained from the correlation spectrum. In synchronous spectrum, the peaks indicate the relative similarity of variation behavior of spectral intensities measured at two separate wavenumbers. In asynchronous spectrum, the peaks denote the dissimilarity of the intensity variation behavior [8]. This 2D IR spectrum can be used to study the mechanism of chemical reactions, distinguish complicated mixture system, and so on.

2 Experiment

2.1 Apparatus and Accessories and Samples

Spectrum GX FTIR spectrometer (Perkin Elmer), equipped with a DTGS detector, in the 400-4000 cm^{-1} range with a resolution of 4 cm^{-1} . Spectra were calculated from a total of 16 scans.

Love Control Corporation's portable programmable temperature controller (Model 50-886). Temperature range: 50~120°C.

Patchouli [*Pogostemon cablin* (Blanco) Benth.] of different geographical origins called Nanxiang, Paixiang, Zhanxiang, and Zhaoxiang were identified and provided by Guangzhou University of Traditional Chinese Medicine.

2.2 Procedure

First, all the patchouli were purified, comminuted, and desiccated. Then each sample of patchouli powder was blended with KBr powder, ground again, and pressed into a tablet. Put the tablet in the sample pool of temperature controller. A pre-established program controlled the whole process of increasing temperature from 50 to 120 °C, the spectra were collected at intervals of 10 °C. The process of increasing temperature took a total time of 40 min.

In this paper, the wavelet transform was operated in Matlab which is a widely used tool in scientific computing, and 2D IR correlation spectra were obtained by analyzing the set of temperature-dependent dynamic spectra with our home written 2D IR correlation analysis software, TD2.1.

3 Results and Discussions

3.1 Conventional IR Spectra and Second Derivative IR Spectra

The conventional IR spectra of these four patchouli samples, taken at room temperature, are shown in Panel (I) of Fig. 1. It shows that these four spectra are rather similar. The differences are not noticeable. They all have peaks at $(1514\pm 2) \text{ cm}^{-1}$, (1377 ± 1)

cm^{-1} , $(1320\pm 1)\text{cm}^{-1}$, and $(781\pm 1)\text{cm}^{-1}$. The main difference between these four spectra is that in the range of $2000\sim 400\text{cm}^{-1}$, in the Spectrum A, the strongest peak is at 1652cm^{-1} , while in the other three spectra, the strongest peaks are at about 1036cm^{-1} . And in these three spectra, the peak belonging to carbonyl group at 1735cm^{-1} is more obvious than in Spectrum A. Based on this, Nanxiang can be somewhat easy to be distinguished from other three samples. Apparently, just with the comparison of the conventional IR spectra, it's rather difficult to discriminate the other three samples.

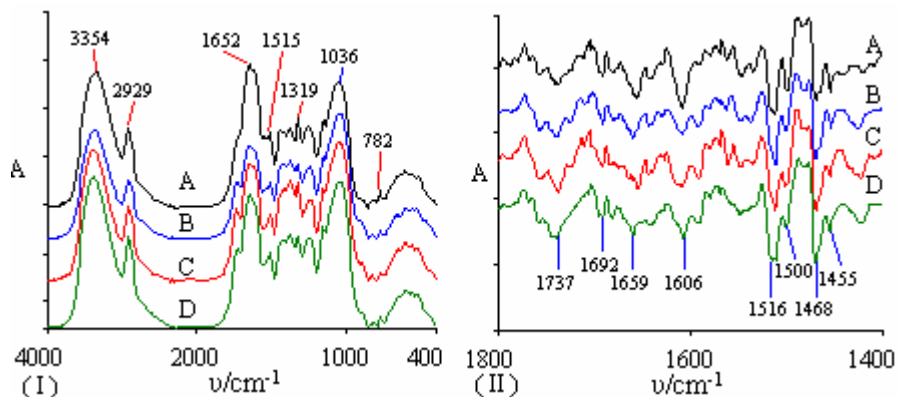


Fig. 1. Conventional IR spectra (Panel (I)) and second derivative IR spectra (Panel (II)) of four patchouli samples of different geographical origins at room temperature. (A) Nanxiang, (B) Paixiang, (C) Zhanxiang, (D) Zhaoxiang.

Panel (II) of Fig. 1 shows the second derivative spectra of the samples. From the spectra it can be seen that the difference at 1737cm^{-1} becomes more obvious. Compared with peaks at 1659cm^{-1} and 1606cm^{-1} , the peak at 1737cm^{-1} in Spectrum A is quite weak, but in the other three spectra, the peaks there are quite strong. This is coincident with the conventional IR spectra. But altogether, these four spectra still look quite similar and it is rather difficult to use second derivative IR spectra to distinguish these samples.

3.2 2D IR Spectra

Fig. 2 shows the synchronous correlation spectra of these four patchouli samples. By conventional IR spectra, we can easily discriminate Nanxiang from others. Here, by 2D IR spectra, Nanxiang is also easy to discriminate from the other three samples. See Panel (I) of Fig. 2, it's quite different from other panels. For examples, the strongest peak in Panel (I) is at 1148cm^{-1} , while the ones in the other three panels are at about 940cm^{-1} . And in the range from 850cm^{-1} to 1010cm^{-1} , Panel (I) only has a noticeable on-diagonal peak called auto-peak. All these make Panel (I) quite different from others.

Spectrum (II), (III), and (IV) are relatively similar. But differences between them are more remarkable than in conventional IR spectra and second derivative IR spectra. For example, the auto-peak at 1150cm^{-1} in Spectrum (IV) is much stronger than the ones in

Spectrum (II) and (III). And at $(1000, 946) \text{ cm}^{-1}$, the off-diagonal peaks called cross-peak in Spectrum (II) is relatively strong, but the one in Spectrum (III) is much weaker. Thus, we can discriminate these four patchouli samples, but not so easy.

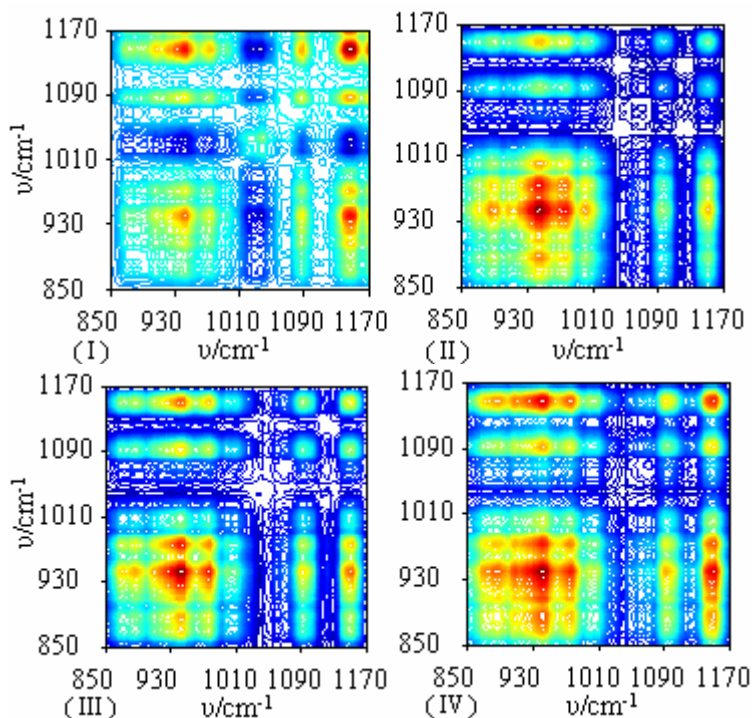


Fig. 2. Synchronous 2D IR correlation spectra of the patchouli samples. (I) Nanxiang, (II) Paixiang, (III) Zhanxiang, (IV) Zhaoxiang.

3.3 2D IR Spectra with Wavelet Transform

To improve the resolution of the 2D IR spectrum, we apply wavelet transform to the data process. Though wavelets can well express the signals that have non-stationary variations, sometimes different wavelet basic functions result quite differently in the same matter. After comparison with other wavelets, Symlet wavelets with order 8 (sym8 for short in Matlab) was chosen to be the optimized wavelet to analyze IR data.

In IR spectrum, all information comes from the position and intensity of the peaks. To enlarge the differences between the samples, we applied wavelet transform to each spectrum. Firstly, the spectrum signal was decompose into level 5, secondly, the detail and approximation coefficients at each level were extracted, and thirdly, these coefficients were processed. In this paper, to make the peaks in IR spectrum clear, obvious, and separate, the detail coefficients at level 3, level 4 and level 5 were multiplied by 2.5. At last, signal was reconstructed using the processed coefficients.

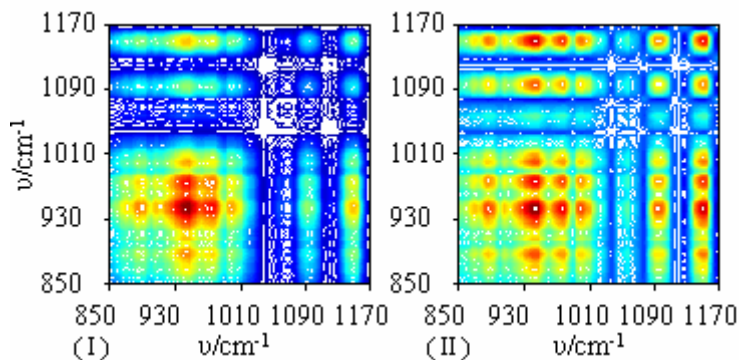


Fig. 3. Comparison of synchronous 2D IR spectrum of Paixiang patchouli (*I*) with processed one by wavelet transform (*II*)

For synchronous spectra, wavelet transform bring them many changes, which are shown in Fig. 3. Comparing these two spectra, it's apparent that after applying wavelet, the resolution is obviously enhanced. In Spectrum (*II*), the auto-peaks at 885cm^{-1} , 943cm^{-1} , 976cm^{-1} , 1000cm^{-1} , 1097cm^{-1} , and 1150cm^{-1} become more legible. The cross-peaks are also changed in the same way. For example, the ones at $(885, 976)\text{cm}^{-1}$, $(885, 1000)\text{cm}^{-1}$, $(976, 1000)\text{cm}^{-1}$, and $(1000, 1150)\text{cm}^{-1}$ are very difficult to identify in the Spectrum (*I*), but in Spectrum (*II*), these peaks are very clear and legible. As we known, in spectrum analysis, most of the information comes from peaks. Therefore, the more peaks appear, the more information could be obtained. This makes it easier to distinguish these four samples.

As mentioned above, with conventional IR spectrum, second derivative spectrum and 2D IR spectrum without wavelet transform process, Nanxiang and Zhaoxiang have been quite easy to be distinguished. With 2D IR, although the other two samples, Paixiang and Zhaoxiang, can be distinguished too, their spectra are rather similar so that it is not so easy to distinguish these two samples. So at this stage, we focus on these two samples.

Fig. 4 is the 2D IR spectra of Paixiang and Zhanxiang with wavelet transform process. With a superficial browse, these two spectra look a little similar. But when observe carefully, many differences can be found out. Look at the auto-peaks at first. At 867cm^{-1} , Zhanxiang has an auto-peak, Paixiang doesn't have. At 1000cm^{-1} , Paixiang has an auto-peak, Zhanxiang doesn't have. At 1095cm^{-1} , the auto-peak in Paixiang is weak, but the one in Zhanxiang is relatively strong. Now look at the cross-peaks. At $(867, 943)$, $(867, 976)$, $(867, 1000)$, and $(867, 1150)\text{cm}^{-1}$, the cross-peak are very weak in Paixiang, but relatively strong in Zhanxiang. And at $(1000, 1150)$ and $(1000, 1100)\text{cm}^{-1}$, the peaks in Paixiang is noticeable, but in Zhanxiang, they are nearly invisible.

It can be concluded from the above discussion that, with the help of 2D IR combined wavelet transform, these four similar patchouli samples of different geographical origins could be easily distinguished from each other.

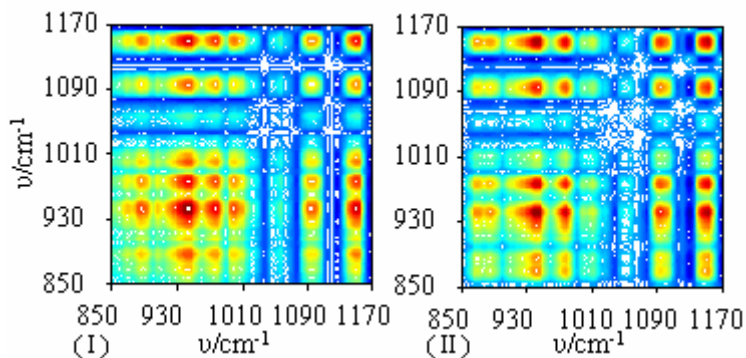


Fig. 4. Synchronous 2D IR correlation spectra of Paixiang and Zhanxiang samples with wavelet transform process. (I) Paixiang, (II) Zhanxiang.

4 Conclusions

This study showed a new method to discriminate patchouli, which is one of the complicated and similar traditional Chinese medicines, of different geographical origins. Their conventional IR spectra, second derivative IR spectra are rather similar and hardly to be differentiated. But the differences of their 2D IR spectra, especially after wavelet transform process, are so remarkable and easy to be used to distinguish the patchouli. The study also indicated that 2D IR with wavelet transform could be used in the discrimination of complicated mixture systems just like traditional Chinese medicines, and it would be a rapid, easy and economical analysis method.

Acknowledgements

This work was substantially supported by a grant from the Ministry of Science and Technology of the People's Republic of China (2002BA906 A29-4), and by the Research Grant Council of Hong Kong SAR under Project HKBU 2156/04E.

References

1. Zuo L, Sun SQ, Zhou Q, Tao JX, Noda I. 2D-IR Correlation Analysis of Deteriorative Process of Traditional Chinese Medicine 'Qing Kai Ling' Injection. *Journal of Pharmaceutical and Biomedical Analysis*, 30 (5): 1491-1498 JAN 1 2003
2. Lu GH, Sun SQ, Leung HW, Chan KKC, Study on the Identification of Danggui-tou and Danggui-wei by Two-Dimensional Infrared Correlation Spectroscopy. *Spectroscopy and Spectral Analysis* 24 (2004) 311-314
3. Albert Boggess, Francis J. Narcowich: *A First Course in Wavelets with Fourier Analysis*. Publishing House of Electronics Industry (2004) 6-10
4. Stephane Mallat. *A Wavelet Tour of Signal Processing*. China Machine Press (2003) 79-82

5. V. H. Segtnan, S. Sasic, T. Isaksson, and Ozaki. Studies on the Structure of Water Using Two-Dimensional Near-Infrared Correlation Spectroscopy and Principal Component Analysis. *Anal. Chem.* 2001, Vol. 73, 3153-3161
6. I. Noda, *Bull. Am. Phys. Soc.* 21 (1986) 520
7. I. Noda: *Appl. Spectrosc.* 47 (1993) 1329
8. I. Noda: *Advance in Two-Dimensional Correlation Spectroscopy. Vibrational Spectroscopy* 36 (2004) 143-165
9. Ma A, Stratt RM: Selecting the Information Content of Two-Dimensional Raman Spectra in Liquids. *Journal of Chemical Physics* 119 (2003) 8500-8510
10. Rui Hua, Suqin Sun, Qun Zhou, et al.: Discrimination of Fritillary According to Geographical Origin with Fourier Transform Infrared Spectroscopy and Two-Dimensional Correlation IR Spectroscopy. *Journal of Pharmaceutical and Biomedical Analysis* 33 (2003) 199-209

Gait Recognition Using View Distance Vectors

Murat Ekinici and Eyup Gedikli

Computer Vision Lab. Dept. of Computer Engineering,
Karadeniz Technical University, Trabzon, Turkey
ekinici@ktu.edu.tr

Abstract. This paper¹ presents a new approach for human identification at a distance using gait recognition. Binarized silhouette of a motion object is represented by 1-D signals which are the basic image features called the distance vectors. The distance vectors are differences between the bounding box and silhouette, and extracted using four view directions to silhouette. Based on normalized correlation on the distance vectors, gait cycle estimation is first performed to extract the gait cycle. Second, eigenspace transformation based on PCA is applied to time-varying distance vectors and then Mahalanobis and normalized Euclidean distances based supervised pattern classification is finally performed in the lower-dimensional eigenspace for human identification. Experimental results on two main database demonstrate that the right person in top two matches 100% of the times for the cases where training and testing sets corresponds to the walking styles for data set of 25 people, and other data set of 22 people.

1 Introduction

Gait is a behavioral biometric source that can be acquired at a distance. Gait recognition is the term typically used in the computer community to refer to the automatic extraction of visual cues that characterize the motion of a walking person in video and is used for identification purpose[1][2][3][6]. The main purpose and contributions of this paper are summarized as follows:

- We attempt to develop a simple but effective representation of silhouette for gait-based human identification using silhouette analysis. Similar observations have been made in [7][8], but the idea presented here implicitly more capture both structural (appearances) and transitional (dynamics) characteristics of gait.
- Instead of width/length time signal of bounding box of moving silhouette usually used in existing gait period analysis [3][9][10][2], here we analyze four distance vectors extracted directly from differences between silhouette and the bounding box, and further convert them into associated four 1D signals.

The novel approach is basically to produce view directions based 1-D distance vectors represent the distances between the outer contour of the binarized silhouette and the bounding box placed around silhouette. Thus, four 1-D signals

¹ This work is supported in part by KTU (Grant No:KTU-2002.112.009.1).

are extracted for each view directions, they are top-, bottom-, left-, and right-views. Then correlation-based a similarity function is executed to estimate gait cycle of moving silhouette. As following main purpose, depending on four view distance vectors, PCA based gait recognition algorithm is first performed. Both Mahalanobis and Euclidean distances based similarities are then achieved to obtain similarity measures on training and testing data. Next, a fusion strategy on that similarities is calculated to produce final decision. At the experiments on two main different databases, the fusion strategy has presented very robust and reliable results for human identification.

2 Feature Extraction

Given a sequence of images obtained from a static camera, we detect and track the moving person then compute the corresponding sequence of motion regions in each frame. Motion segmentation is achieved via background modeling/subtraction using a dynamic background frame estimated and updated in time, more details are given in [5]. Then a morphological filter operator is applied to the resulting images to suppress spurious pixel values. Once a silhouette generated, a bounding box is placed around silhouette. Silhouette across a motion sequence are automatically aligned by scaling and cropping based on the bounding box.

2.1 Silhouette Representation

Silhouette representation is based on the distance vectors for silhouette which is generated from a sequence of binary silhouette images $bs(t) = bs(x, y, t)$, indexed spatially by pixel location (x, y) and temporally by time t . The distance vectors are the differences between the bounding box and silhouette. Form a new 2D image $F_T(x, t) = \sum_y bs(x, y, t)$, where each column (indexed by time t) is the top-view distance vector of silhouette image $bs(t)$, as shown in figure 1-Middle. Each value $F_T(x, t)$ is then a count of the number of the row pixels between the top of the bounding box and the outer contours in that columns x of silhouette image $bs(t)$. The result is a 2D pattern, formed by stacking the top view together to form a spatio-temporal pattern. A second pattern which represents the bottom-view $F_B(x, t) = \sum_y bs(x, y, t)$ can be constructed by counting the number of the row pixels between the bottom of the box and its silhouette, as shown in figure1-Left:top-bottom. The third, $F_L(y, t) = \sum_x bs(x, y, t)$, and fourth, $F_R(y, t) = \sum_x bs(x, y, t)$, patterns are then constructed by taking the row differences (the number of the column pixels) from the left and the right of the box to silhouette, respectively, as shown in figure 1.Right:top-bottom. The variation of each component of the each view distance vectors can be regarded as silhouette signature of that object. From the temporal distance plots, it is clear that the view distance vector is roughly periodic and gives the extent of movement of the outer contours on the view direction of silhouette. The brighter a pixel in 2D patterns, shown in figure 1, the larger value is the value of the view-based distance vectors in that position.

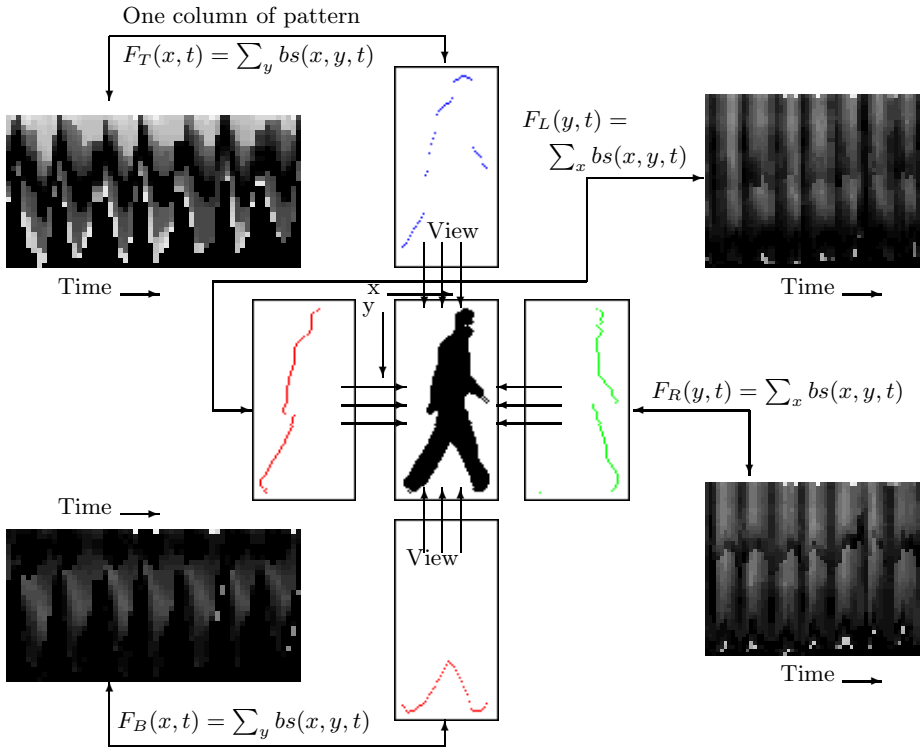


Fig. 1. Silhouette representation. (Middle) Silhouette and four views, (Left) temporal plot of the distance vectors for top and bottom views, (Right) temporal plot of the distance vectors for left and right views.

2.2 Gait Cycle Estimation

Gait cycle estimation is first step in the gait identification. Several vision methods have exploited this fact to compute the period of human gait from image features [9][10][2]. Different from them, here this paper presents four view distance vectors based variations on the moving silhouette as a function of time so as to enable them to cope effectively with both lateral view and frontal view. The gait cycle estimation process is as follows. The output of the detecting/tracking module gives a sequence of bounding boxes for every object [5]. Then each distance vectors is separately produced in each frames, and normalized correlation is performed on the distance vectors obtained from consecutive frames. To quantify the signals of the correlation results, the algorithm may first remove their background component by subtracting their mean and dividing by their standard deviation, and then smooth them with a symmetric average filter. Further, it computes their autocorrelation to find peaks in results [11]. Then, their first-order derivatives are computed to find peak positions by seeking the positive-to-negative zero-crossing points. Due to the bilateral symmetry of human gait, the

autocorrelation will sometimes have minor peaks half way between each major peaks [9][2]. Hence, the real period as the average distance between each pair of consecutive major peaks is estimated.

2.3 Training

First, we convert a two-dimensional silhouette shape into four one-dimensional distance vectors, which will significantly reduce the subsequent computational cost. The distance vector sequence is accordingly used to approximately represent a temporal pattern of gait. This process of original feature extraction is explained in section 2.1, and also illustrated in Figure 1. The training process similar to [2][4] is illustrated as follows: Given k class for training, and each class represents a sequence of the distance vector signals of a single person, as illustrated. Multiple sequences of each subject can be added for training, but we have used a sequence includes one gait cycle. Then the mean and the global covariance matrix of each view training set can separately obtained. Next the nonzero eigenvalues of covariance matrix and associated eigenvectors are computed based on theory of *singular value decomposition* [4]. Then a transform matrix to project an original distance vector signals into a point in the eigenspace are constructed. The projection average of each training sequence in the eigenspace is finally calculated to be able to represent to the individual characteristics of person.

Pattern Classification and Fusion. Normalized Euclidean, and Mahalanobis, distance measures have separately been selected for classification. The accumulated distance between the associated centroids obtained in the process of testing and of training can be easily computed by using Euclidean and Mahalanobis distance measurements. In the both distance measures, the classification result for each view is then accomplished by choosing the minimum of distance. At the using of each distance vectors alone without making a relationship between each others, we can see from experimental observations that recognition errors often happen when the two smallest values of the similarity function are very close for one view but not other view(s). Therefore, in order to increase the recognition performance, the classification results of the each view distance vectors are finally fused using a distance voting strategy.

In the fusion process, weight parameters of each view distance are determined by gait motion in different views with respect to the image plane, namely lateral, frontal, and oblique views for both ways (from left to right, vice versa). In the fusion strategy, each views is separately treated. Then the strategy is combining the distances of each views at the end by assigning equal weight. As a result processing, if any two of distance similarity values in four views give maximum similarities for same person, then the identification result is accepted positive. So, this fusion strategy has rapidly increased the recognition performance on the experiments.

3 Experimental Results and Conclusion

For the test sequences, the detection and tracking algorithm that we use is provided by the work in [5]. The Viterbi algorithm was used to identify the test

sequence, since it is efficient and can operate in the logarithmic domain using only additions [12]. The first experiments are obtained on the CMU database. This database has 25 subjects (23 males, 2 females) walking on a treadmill [13]. There are about 8 cycles in each sequence, and each sequences is recorded at 30 frames per second. But we used only 3 cycles in each sequence and slow walking type. The last cycle in the sequence was used for testing, others for training. The experimental results obtained using the proposed method are summarized in Table 1. Mahalanobis distance based classification performance has presented better than Euclidean distance based. Nevertheless, it can also be seen on both distance similarities that, the right person in the top three matches 100% of the times for the cases where the training and testing sets correspond to the slow walking styles.

As second experiment, our database established for gait recognition has 22 people (2 females, 20 males), and subjects are walking laterally to the camera, the directions of walking is from left to right, and from right to left. The database includes two sequences for each subject. One sequence includes 3 gait cycle for each direction, and the length of each gait cycle varies with the pace of the walker, but the average is about 26 frames. The subjects walk along a straight-line path at free cadences in lateral view with respect to the image plane, and 15 subjects were walking outside, seven subjects were walking inside platform. The results for the experiments along with cumulative match scores are summarized in Table 2. Two cycles were used for training, and next one cycle was used for testing. Walking from left to right and the other direction are separately tested to achieve initial experimental results. When the results of each view distance vectors are re-implemented by fusing strategy as explained in section 2.3, significantly improvements on the gait recognition has been achieved, as given in Tables 1 and 2. This is the robust implementation and is also one of the novelty presented by this paper.

Table 1. Performance on the first data set: classification rate using the fusion strategy

	Rank 1	Rank 2	Rank 3
Mahalanobis Distance	72	100	100
Euclidean Distance	44	76	100

Table 2. Performance on the second data set: Three gait cycles were used, two cycles for training, one cycle for testing. (The abbreviation used : M:Mahalanobis, E: Euclidean, L \Rightarrow R : From Left to Right).

Direction	Distance	Outdoor(15 person)			Indoor(7 person)		All(22 person)		
		Rank 1	Rank 2	Rank 3	Rank 1	Rank 2	Rank 1	Rank 2	Rank 3
L \Rightarrow R	M	67	100	100	86	100	68	95	100
	E	47	87	100	86	100	59	96	100
R \Rightarrow L	M	67	100	100	71	100	68	100	100
	E	47	87	100	71	100	36	68	100

As a conclusion, this paper has presented a novel approach to represent and recognize people by their gait. A recognition rates of 72% (Rank 1), and 100% (Rank 2) for a data set of 25 people, and 68% (Rank 1), and 100% (Rank 2) for a data set of 22 people. The novelty in the presented algorithm lies in its simplicity, the efficiency of the implementation, the usefulness in real-time applications, the effectiveness in view independent applications, the robustness to some factors such as motion object size and regular noise effects, and new approach includes multi-view and multi-purpose aspects to silhouette for silhouette based human identification.

References

1. Nixon M. S., Carter J. N.: Advances in Automatic Gait Recognition. Proc. of IEEE Int. Conf. on Automatic Face and Gesture Recognition, 2004.
2. Wang L., Tan T., Ning H., Hu W.: Silhouette Analysis-Based Gait Recognition for Human Identification. IEEE Trans. on Pattern Analysis Machine Intelligence, Vol. 25, No. 12, 2003.
3. BenAbdelkader C., Cutler R. G., Davis L. S.: Gait Recognition Using Image Self-Similarity. EURASIP Journal of Applied Signal Processing, pp. 1-14, April, 2004.
4. Huang P., *et. al.*, Human Gait Recognition in Canonical Space Using Temporal Templates. IEE Proc. of Vision Image and Signal Proc., Con. Vol.146, No.2, 1999.
5. Ekinçi M., Gedikli E.: Background Estimation Based People Detection and Tracking for Video Surveillance. Springer LNCS 2869, ISCIS 2003, Turkey, November, 2003.
6. S. Sarkar, *et.al.*, The HumanID Gait Challenge Problem: Data Sets, Performance, and Analysis. IEEE Trans. on Pat. Anal. and Mach. Intell., Vol.27, N0.2, Feb.2005.
7. Kale A., *et. al.*, Identification of Humans Using Gait. IEEE Trans. on Image Processing, Vol.13, No.9, September 2004.
8. Yanxi Liu, Collins R.T., Tsin T.: Gait Sequence Analysis using Frieze Patterns. Proc. of European Conf. on Computer Vision, 2002.
9. BenAbdelkader C., Culter R., and Davis L.: Stride and Cadence as a Biometric in Automatic Person Identification and Verification. Proc. Int. Conf. Aut. Face and Gesture Recognition, 2002.
10. Collins R., Gross R., and Shi J.. Silhouette-Based Human Identification from Body Shape and Gait. Proc. Int. Conf. Automatic Face and Gesture Recognition, 2002.
11. Ekinçi M.: Gedikli E.: A Novel Approach on Silhouette Based Human Motion Analysis for Gait Recognition. Springer Lecture Notes in Computer Science, ISVC 2005, Nevada, USA, December, 2005.
12. Phillips J.,*et.al.*, The FERET Evaluation Methodology for Face recognition Algorithm. IEEE Trans. Patt. Anal. and Mach. Int., vol.22, no.10, October, 2000.
13. Gross R., Shi J.: The CMU motion of body (MOBO) database. Tech. Rep. CMU-RI-TR-01-18, Robotics Institute, Carnegie Mellon University, June 2001.

HMM Parameter Adaptation Using the Truncated First-Order VTS and EM Algorithm for Robust Speech Recognition

Haifeng Shen¹, Qunxia Li², Jun Guo¹, and Gang Liu¹

¹ Beijing University of Posts and Telecommunications, Beijing 100876, China
shen_hai_feng@126.com, guojun@bupt.edu.cn, lg@pris.edu.cn

² University of Science and Technology Beijing, Beijing 100083, China
kellylqx@163.com

Abstract. This paper presents a framework of HMM parameter adaptation technique for improving automatic speech recognition (ASR) performance in the noisy environments, which online combines the clean hidden Markov models (HMMs) with the noise model. Based on the given composite HMM corresponding to the initial recognition pass result and truncated vector Taylor series, the noise model in the cepstral domain is updated and refined using iterative Expectation-Maximization (EM) algorithm under maximum likelihood (ML) criterion. Experiments results show that the presented approach in this paper is found to greatly improve recognition performance under mismatched conditions.

1 Introduction

Although the speech recognition systems are effective in the quiet condition, but suffer from major degradation that limits applications in the real-world environments due to a mismatch between the acoustic model and the test data. Attempts are made to overcome this limitation with the adoption of noise robust techniques. There are two ways to reduce the mismatch to achieve satisfactory performance in literature. One way is to create noise-robust features or compensate front-end noisy feature vectors [1]-[4] while keeping the acoustic model unchanged. Another way is to adapt the acoustic model to create the more accurate model for closely matching the noisy environment characteristics [5] [6]. The model-based approaches become the most interesting methods recently. They are based on the statistical model and employ different strategies to approximate environment model in the different feature domain. VTS [1] makes full use of an approximation of the nonlinear function in the log spectral domain for achieving noisy GMM, and iteratively estimates the noise parameters on the whole test sentence using EM algorithm. VTS can be viewed as a particular case of SLA [2] [3]. Although those approaches show the attractive recognition improvements in the noisy environment, they still incur the inherent drawbacks in many applications. In [1]-[3], the independence assumption of log-filterbank coefficients in the log spectral domain is not appropriate, so these compensations are not perfect. Gales have derived a PMC (parallel model combination) technique based on a log-normal

assumption [5]. It is shown to be more effective compared with feature compensation technique, but it needs a formidable computation. By ignoring the dependence among the log-filterbank coefficients, it can reduce much time, but will give some inherit aforementioned drawbacks. As can be seen, many speech recognition systems in literature generally apply the Mel frequency cepstral coefficients (MFCCs) as a feature vector since the cepstral coefficients are nearly incorrelate. To solve aforementioned problems, in this paper, we directly model the environment corruption in the cepstral domain as a nonlinear complicated function, and realize the HMM parameter adaptation. In this case, given a noise model, the goal is to adapt the acoustic models (HMMs) to well describe the corrupted input speech sequence. However, there is a fundamental problem associated with this technique, that is, the model adaptation needs huge computation in the large vocabulary continuous speech recognition (LVCSR) system which is impossible in the real-application conditions [6]. Instead, with only combination of the means of both the clean HMMs and noise model, this technique can greatly reduce computation complexity. Correspondingly, the model of the noise can also be updated using EM algorithm, which is more reasonable than that estimated in the nonspeech periods [6]. In this paper, we present a framework of HMM parameter adaptation technique based on EM algorithm. For simplicity, we use maximum likelihood criterion. However, it is easy to realize maximum a posteriori (MAP) estimation which only adds the conjugate prior density into the EM auxiliary function [4]. The remainder of the paper is organized as follows. The next section briefly describes the framework of the HMM parameter adaptation and employs EM algorithm to estimate the noise model. In section 3, we evaluate the performance of our presented method and some conclusions are drawn in section 4.

2 Framework of HMM Parameter Adaptation Approach

Given the corrupted observation sequence $Y = \{y_1, y_2, \dots, y_T\}$ and a set of trained models $\Lambda_X = \{\lambda_m\}$ where λ_m is the m^{th} subword context-dependent HMM unit trained from X embedded in Y and the aim of the recognition system is to recognize the word sequence $W = \{W_1 \dots W_L\}$ using the maximum a posteriori (MAP) decoder

$$\hat{W} = \arg \max_w P(Y | W, \Lambda_X)P(W), \tag{1}$$

where the first term is the likelihood of observation sequence Y given the word sequence W , the second term is the language model.

Λ_X is a K-states continuous density HMM (CDHMM) model with state transition probability $a_{p,q}$ ($0 \leq a_{p,q} \leq 1$) and each state s is modeled as Gaussian mixture distribution with parameter $\{p_{sj}, u_{xsj}, \Sigma_{xsj}\}_{j=1,2,\dots,M}$, where M and p_{sj} respectively denote the number of mixture components, the mixture gain in each state s , u_{xsj}, Σ_{xsj} are respectively the mean vector and the covariance matrix of each mixture component j at state s . Since there exists a mismatch between the corrupted speech Y and the acoustic model Λ_X , we can't achieve the best recognition performance. In this paper, in order to

reduce computation load, we only consider the mean parameters in the acoustic models. The noise corruption in the model space, as is well seen, resulting in the mean parameters mismatch in each mixture component j at state s , is given as [4] [6]

$$\begin{aligned} \mu_{ysj} &= D \cdot \log\{\exp(D^+ \mu_{xsj}) + \exp(D^+ \mu_n)\} = \mu_{xsj} + D \cdot \log\{1 + \exp\{D^+ \cdot \{\mu_n - \mu_{xsj}\}\}\} \\ &= \mu_{xsj} + f(\mu_{xsj}, \mu_n), \end{aligned} \tag{2}$$

where μ_{ysj} , μ_{xsj} and μ_n are respectively the means of the distorted HMM models Λ_Y , the clean HMM models Λ_X and the noise model Λ_N . D^+ denotes the generalized inverse of the non-square DCT matrix D . Λ_N generally can be estimated under ML:

$$\tilde{\Lambda}_N = \arg \max_{\Lambda_N} P(Y | \Lambda_N) = \arg \max_{\Lambda_N} \sum_S \sum_J P(Y, S, J | \Lambda_N), \tag{3}$$

where $P(\cdot)$ is the likelihood of the corrupted observation sequence Y given Λ_N , $S = \{s_1, s_2, \dots, s_T\}$ is the set of all possible state sequences, and $J = \{j_1, j_2, \dots, j_T\}$ is the set of all mixture component sequences. If we obtain the refined noise model $\tilde{\Lambda}_N$, according to the corruption function described in Eq.(2), we can derive the another optimal decoder by revising Eq.(1):

$$\hat{W} = \arg \max_W P(Y | W, \Lambda_Y = \{\Lambda_X, \tilde{\Lambda}_N\})P(W), \tag{4}$$

where Λ_Y is corrupted HMM obtained by HMM combination. As can be seen, it firstly converts the clean model Λ_X and the noise model $\tilde{\Lambda}_N$ into the linear spectral domain by taking $\exp(D^+)$ operation assuming that the noise is additive in the linear spectral domain, then takes inverse operation $D \cdot \log$ and back into the cepstral domain. In this paper, the system uses the first 13 cepstral coefficients $\{c_0, c_1, \dots, c_{12}\}$ as a acoustic vector where allows HMM parameters to be transformed to the linear spectral domain by using the generalized inverse D^+ , due to the dimensionality of the filterbank features is 24.

In the following step, we will estimate the noise by EM algorithm. Assuming the noise is a single Gaussian, the noisy speech is not hidden Markov model. This behavior prevents us to estimate the noise parameter using EM algorithm. Fortunately, it can be solved by employing the linear approximation strategies, such as the truncated first-order VTS approximation. Auxiliary function of EM algorithm is given as [7]

$$\begin{aligned} Q(\tilde{\Lambda}_N | \Lambda_N) &= E\{\log P(Y, S, J | \tilde{\Lambda}_N) | Y, \Lambda_N\} \\ &\propto - \sum_{t=1}^T \sum_{s=1}^K \sum_{j=1}^M \gamma_t(s, j) (y_t - \tilde{\mu}_{ysj})' (\Sigma_{ysj})^{-1} (y_t - \tilde{\mu}_{ysj}), \end{aligned} \tag{5}$$

where $\tilde{\mu}_{ysj} = A_{sj} \mu_{xsj} + B_{sj} \tilde{\mu}_n + C_{sj}$ in which A_{sj} , B_{sj} and C_{sj} are the coefficients dependent on the expansion vector point of means μ_{xsj} and μ_n [4], T is the frame number of Y , $\gamma_t(s, j) = p(Y, s_t = s, j_t = j | \Lambda_N) = \alpha_t(s) \beta_t(j) p(j_t = j | y_t; \mu_{ysj}, \Sigma_{ysj})$ is the joint likelihood of

Y and mixture component j from stat s producing the observation y_t given the models Λ_N , where $\alpha_t(s), \beta_t(s)$ are respectively the forward and backward probabilities. Computing the derivatives of $Q(\tilde{\Lambda}_N | \Lambda_N)$ with respect to $\tilde{\Lambda}_N = \{\tilde{\mu}_n\}$ and setting the derivatives to zero, the noise estimate $\tilde{\mu}_n$ is obtained, i.e.

$$\tilde{\mu}_n = \frac{\sum_{t=1}^T \sum_{s=1}^K \sum_{j=1}^M \gamma_t(s, j) B_{sj}'(\Sigma_{ysj})^{-1} (y_t - A_{sj} \mu_{x_{sj}} - C_{sj})}{\sum_{t=1}^T \sum_{s=1}^K \sum_{j=1}^M \gamma_t(s, j) B_{sj}'(\Sigma_{ysj})^{-1} B_{sj}} \quad (6)$$

3 Experiments and Results

The described approach in this paper is tested by speaker-independent LVCSR system. The utterances of 82 speakers (41 males and 41 females) from the clean mandarin Chinese corpus provided by the 863 plan ([8]) are trained for triphone-based HMM acoustic models, where each triphone unit is modeled as a three-emitting-state left-right topology with a mixture of 8 Gaussians per state and diagonal covariance matrices, and 100 utterances of the clean test set are artificially contaminated by adding the

Table 1. Recognition rates for the white noise (%)

SNR	0dB	5dB	10dB	15dB	20dB	clean	Avg.
baseline	0.54	2.24	10.99	38.31	66.95	86.53	34.26
VTS+LOG	1.55	6.50	26.78	54.26	70.74	83.75	40.60
SLA2+LOG	1.47	6.66	27.09	55.50	71.67	83.58	41.00
VTS+CEP	11.22	25.62	48.22	66.87	75.00	85.60	52.09
second pass	5.57	25.31	53.41	72.21	82.59	86.15	54.21

Table 2. Recognition rates for the babble noise (%)

SNR	0dB	5dB	10dB	15dB	20dB	clean	Avg.
baseline	1.55	8.05	28.64	71.36	82.51	86.53	46.44
VTS+LOG	6.42	24.07	51.94	72.52	81.04	83.75	53.29
SLA2+LOG	6.73	24.23	52.24	73.14	80.88	83.58	53.47
VTS+CEP	13.70	29.80	61.61	76.01	79.57	85.60	57.72
second pass	15.09	41.33	67.34	79.64	83.75	86.15	62.22

Table 3. Recognition rates for the factory1 noise (%)

SNR	0dB	5dB	10dB	15dB	20dB	clean	Avg.
baseline	1.01	9.29	40.94	71.36	82.43	86.53	48.59
VTS+LOG	1.47	9.29	40.56	70.51	82.59	83.75	48.03
SLA2+LOG	1.55	9.27	40.87	70.59	82.43	83.58	48.05
VTS+CEP	8.98	29.80	61.61	76.01	79.57	85.60	56.93
second pass	4.33	29.88	64.63	79.41	84.06	86.15	58.08

white noise, the babble noise and the factory1 noise from NoiseX92 [9] according to different SNR (signal-to-noise ratio) ranging from 0dB to 20dB. To extract the acoustic features, 39 dimensional features consisting of the first 13 cepstral coefficients and their time derivatives are computed and fed into the speech recognition system.

In order to show the effectiveness of the approach, we obtain five recognition systems with different approaches. Each system is performed in the different noise with different SNR conditions. In Table 1, Table 2 and Table 3, “baseline” denotes the recognition system don’t use any compensation technique. With different environment model approximation method (e.g. VTS and SLA) and different domain feature compensation (e.g. the log domain and the cepstral domain), we have different recognition system with different feature compensation approaches such as “VTS+LOG” “SLA2+LOG” and “VTS+CEP”. In addition, “second pass” denote the system embedded with the presented approach in this paper. Table 1 lists the experimental results of five recognition systems in the stationary white noisy environment at various SNRs. The performance of “baseline” under the clean environment is 86.53%. With an increasing amount of the additive white noise, “baseline” quickly deteriorates to 0.54%. With the feature approaches compensated in the log domain (“VTS+LOG” and “SLA2+LOG”), the systems achieve substantially improvements in all of noisy environment at the expense of relative degradation in the clean environment. Use of “VTS+CEP”, the performance is further improved with 11.49% and 11.09% compared with those with “VTS+LOG” and “SLA2+LOG”. When applying the approach described in this paper, the recognition system achieves the best performance with averagely 54.21% recognition rate. Although the performance is very excellent in the high SNRs, e.g. from 10dB to 20dB, it is clear that the recognition at 0dB is not very well with only 5.57% recognition rate. This result outperforms those recognized with “VTS+LOG” and “SLA2+LOG”, but it degrades drastically as compared with “VTS+CEP”. The reasonable fact is that “second pass” only considers the mean vector during the model combination process and EM noise estimation. Although the approach by ignoring the variance information is not effective in the low SNR environment, it can greatly reduce the computation load and time consumption. This issue doesn’t seriously influence the total recognition result. Table 2 and Table 3 list the results in the nonstationary babble and factory1 noisy environments at different SNRs. In Table 2, it is clear, in the different level babble noisy environments, that “second pass” achieves the best performance with averagely 62.22% recognition rate. It is very efficient especially in the 5dB and 10dB levels. Compared with “baseline”, “VTS+LOG”, “SLA2+LOG” and “VTS+CEP”, “second pass” achieve substantial improvements in all noisy environments from 0dB to 20dB. Moreover, the performance in the clean condition is well maintained at about 86% recognition rate. Unlike recognition in the white noise, “second pass” also obtains the best 15.09% recognition rate at 0dB condition and outperforms “VTS+CEP” about 1.3% recognition rate. In the case of the factory1 environments, it proves that the presented approach is effective under the SNR changing environment. We could obtain the improved results by applying HMM parameter adaptation technique. Although “VTS+CEP” obtain the better results compared with those with “VTS+LOG” and “SLA2+LOG”, it is shown in Table 3 that it averagely gives the lower performance than that of the presented approach. Here we notice that feature compensation approaches in the log domain actually don’t improve the recognition performance.

4 Conclusions

In this paper, we present a framework of HMM parameter adaptation technique for robust speech recognition. The acoustic-distorted environment resulting in the speech corruption is modeled by a nonlinear function and approximated by VTS. Given the unlabeled composite clean HMM, the noise model is effectively estimated using EM algorithm. Results show that it can further improve recognition performance. But HMM combination during speech recognition needs the expansive computation load. In the future, we will extend this approach into the feature space for decreasing the computation complexity and take the variances of the noise model into consideration.

Acknowledgement: This research was sponsored by NSFC under Grant No.60475007 and Foundation of China Education Ministry for Century Spanning Talent.

References

1. Moreno, P.J., Raj, B., Stern, R.M.: A Vector Taylor Series Approach for Environment-Independent Speech Recognition. In: Proceedings of IEEE (1995)733-736
2. Kim, N.S.: Statistical Linear Approximation for Environment Compensation. IEEE Signal Processing Letters, 1(1998)8-10
3. Shen, H., Liu, G., Guo, J., Li, Q.: Two-Domain Feature Compensation for Robust Speech Recognition. In: Wang, J., Liao, X., and Yi, Z. (eds.): Advance in Neural Network- ISNN 2005. LNCS 3497, Springer-Verlag(2005)351-356
4. Shen, H., Guo, J., Liu, G., Huang, P., Li, Q.: Environment Compensation Based on Maximum a Posteriori Estimation for Improved speech Recognition, Fourth Mexican International Conference on Artificial Intelligence, LNAI, Springer-Verlag(2005)
5. Gales, M.J.F.: Model-Based Techniques for Noise Robust Speech Recognition. Ph.D. thesis, University of Cambridge(1995)
6. Acero, A., Deng, L., Kristjansson, K., Zhang, J.: HMM Adaptation Using Vector Taylor Series for Noisy Speech Recognition. In: Proceedings of ICSLP 2000(2000)
7. Dempster, A.P., Laird, N. M. and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," J. of the Royal Statistical Society B, Vol.39, (1977)1-38
8. Zu, Y. Q.: Issues in the Scientific Design of the Continuous Speech Database. Available: http://www.cass.net.cn/chinese/s18_yys/yuyin/report/report_1998.htm.
9. Varga, A., Steenneken, H. J. M., Tomilson, M., Jones, D.: The NOISEX-92 Study on the Effect of Additive Noise on Automatic Speech Recognition. Tech. Rep. DRA Speech Research Unit(1992)

Model Type Recognition Using De-interlacing and Block Code Generation

Cheol-Ki Kim¹, Sang-Gul Lee², and Kwang-Baek Kim³

¹ School of Computer Engineering, Miryang National University, Korea

² Department of Computer Engineering, Pusan National University, Korea

³ Department of Computer Engineering, Silla University, Korea
ckkim@mnu.ac.kr, sglee@pnu.edu, gbkim@silla.ac.kr

Abstract. This paper presents a method that automatically recognizes the shoe's outsole products into model type, which flows through the conveyor belts from right to left. The interlaced pixels are displayed when we use the NTSC based camera in experiments. So, we require a suitable post-processing. For the purpose of this processing, it decides to find rectangle region of object by thresholding after removing interlaced pixels using de-interlacing method. And then, after rectangle region is separated into blocks through edge detection, we calculates pixel number per each block, re-classifies using its average, and classifies products into model type.

1 Introduction

De-interlacing method is a method which transform interlaced images into progressive images, it has been much of service in order to remove some visual problems based on interlaced method [1]-[3]. Recently, the most popular method is a directional dependent interpolation techniques such as the edge based line average (ELA), it is the method that perform interpolation in the direction of the highest sample correlation. The basic concept of motion adaptive schemes is to select the appropriate interpolation method according to the motion of the image-spatial filtering in motion area and temporal filtering in static area. Among these methods, the most advanced approaches to de-interlacing is motion compensation based de-interlacing in which interpolation is performed along the motion trajectory. The motion compensation may be useful to get high resolution and flicker-free pictures in motion area. But this approach is highly dependent on the accuracy of motion estimation. Therefore, it is important to develop robust motion estimation method suitable for interlaced image [4]-[6]. This paper proposes an enhanced method that enhances drawbacks ELA based method. These method improve performance as applying de-interlacing against it after detection of ROI(Region of Interest).

2 De-interlacing and Model Recognition

In this paper, we used NTSC camera for construction of low price system. But, when we used to the camera, the interlaced pixels are displayed by moving image.

Therefore, we propose an enhanced de-interlacing scheme for improving the problems. Because de-interlacing effect supply smooth image in captured image, there exist merits that eliminate drawbacks of H/W using S/W method. When the objects are horizontally flowed through conveyor belts, the interlaced pixels are displayed in captured image. Therefore, we eliminate interlaces after search object region. The proposed de-interlacing scheme is followed in this paper.

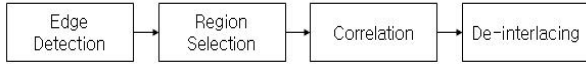


Fig. 1. De-interlacing scheme

First, after we apply Sobel operator to detect edges, then we calculated regions of object. These are related with improvement of processing speed and accuracy, and reduction of unnecessary computation in correlation. The region selection is established through followed three steps.

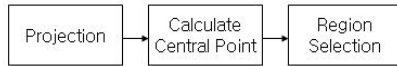


Fig. 2. Region selection scheme

The projection steps are calculated by equation (1)

$$x_{proj}(i) = \sum_{x=0}^{size\ of\ x} pixel(x, i) \quad y_{proj}(i) = \sum_{y=0}^{size\ of\ y} pixel(i, y) \tag{1}$$

After projection steps, central point of object is computed by equation (2).

$$center_x = \frac{\sum_{x=0}^{size\ of\ x} x_{proj}(x)}{size\ of\ x} \quad center_y = \frac{\sum_{y=0}^{size\ of\ y} y_{proj}(y)}{size\ of\ y} \tag{2}$$

The final region of object about projection image are computed by followings.

```

    if ( xproj(x) ≥ threshold )      x ++ ;
    else      break;
    if ( yproj(y) ≥ threshold )      y ++ ;
    else      break;
  
```

After region selection, we take a correlation computation procedure as third step for de-interlacing. When this procedure select a region of object, it moves as line as those coefficients after calculating auto correlation coefficients between even and odd lines. As well, when we offer some offset in odd line, we have to know correspondence between even and odd line in this procedure. To process it, equation (5) shows the final offset's computation method.

$$difference = \sum_{x=left}^{right} | pixel(x, y) - pixel(x + offset, y + 1) | \tag{3}$$

$$offset_{min} = \min\{difference\{ | offset | < range \}\} \tag{4}$$

$$offset_{sum} = \sum_{y=top}^{bottom} | offset_{min} | \quad offset_{final} = \frac{offset_{sum} \times 2}{bottom - top} \tag{5}$$

There are numbers of model in products produce in factory. In this case, when several models are inputted into estimation system through conveyer belt, the system have to distinct each model for difference of shoes outsole estimation condition about each model. For example, according as model, location or numbers of hole which engraved on outsole are different. And location of character region according as variety of shape is also different. For solving its problems, this paper proposes an enhanced model recognition scheme. Through its procedure, after we extract and classify features of each image, then we automatically produce model codes.

Although there exist several methods for classifying inputted images, we select an edge detection based method. Further, we reduce numbers of feature vectors for being easy to do, because of computational overhead. For this, we propose a block based code sequence generation scheme. First, after we select region of object, we divide as block of a fixed number (in this paper, we apply 3*9 input node). Then, we count pixel number of each block according equation(6), generate input patterns, and recognize models.

$$block_pixel_c = \sum_{y=y_c}^{block_y} \sum_{x=x_c}^{block_x} pixel(x, y) \tag{6}$$

Because directions of object are various, the model detection have to do through correction about slope image inputs. That is, when the object is slopped, we generate model input pattern after region selection as parallel rectangle.

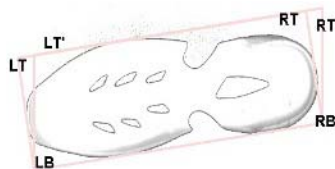


Fig. 3. Slope region selection

[Proposed Algorithm : slope region selection using tangent line]

- ① We get a tangent line as standard two points bordering object. As the points are shifted along the tangent line, we get two points(LB, RB) on contact with object and perpendicular line.

- ② Run parallel with this line, we find two points(LT', RT') that correspondence with upper corner of object.
- ③ Through following procedure, we find LT and RT (In the following equations, x, y represents coordinates and θ is a tangent angle).

$$\begin{aligned}
 h &= LT'.y - LB.y \\
 \theta &= \tan^{-1}\left(\frac{LB.x - LB.y}{LB.y - RB.y}\right) \\
 w &= h \times \cos \theta \\
 \Delta x &= w \times \sin \theta, \quad \Delta y = w \times \cos \theta \tag{7} \\
 LT.x &= LB.x + \Delta x \\
 LT.y &= LB.y - \Delta y \\
 RT.x &= RB.x + \Delta x \\
 RT.y &= RB.y - \Delta y
 \end{aligned}$$

- ④ After we apply first rotation transformation about four points(LT, RT, LB, RB) and transform parallel rectangle, we count each pixel, and apply the 1-st transformation again.

$$x_{dest} = x_{source} \cos \theta - y_{source} \sin \theta, \quad y_{dest} = x_{source} \sin \theta + y_{source} \cos \theta \tag{8}$$

- ⑤ We generate codes by counting pixels through edge detection on each block.

$$average = \frac{\sum_{c=0}^{pattern_size} block_pixel_c}{pattern_size}, \quad \begin{cases} code = 1 & \text{if } (block_pixel_c \geq average) \\ code = 0 & \text{otherwise} \end{cases} \tag{9}$$

- ⑥ Through comparison with learned code, the winner is chosen based on minimum distance by hamming distance.

$$difference = \sum_{i=0}^{||mod||} |node_{register}(number_{pattern}i) - node_{input}(i)| \tag{10}$$

$$winner = \min\{difference\{number_{pattern} < \max_{mod\ el}\}\} \tag{11}$$

3 Experimental Results

We implemented the proposed method in Pentium IV 2.3GHz CPU, 512MB RAM and Visual C++. Experimental data is 300 test images that are the outsole. Test Images have 640x480 resolutions that are acquired by the Samsung BW-2303 CCD camera. Experiments were focused on two major results. First, we analyzed pixel errors in the selection of real area which before and after applying the interlacing

method. Second, we measured the recognition rate when applied the proposed method for model recognition.

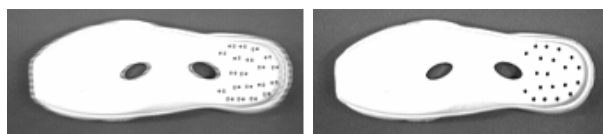


Fig. 4. Comparison of before and after de-interlacing (left: before, right: after)

For the performance test of de-interlacing method based on ROI that is the proposed method in this paper, we compared the proposed method with ELA-based method and the conventional ROI selection method. Table 1 shows about 64% efficiency of interlacing reduction and 0.76 pixel error in ELA-Based method. And the proposed method is more efficient as 96% interlacing reduction rate and 0.04 average pixel error than ROI selection method which is about 85% interlacing reduction rate.

Table 1. Experimental results of de-interlacing

	Method	De-interlacing ratio	Average pixel error
1 st test	Before region selection	64%	0.76 pixel
2 nd test	After region selection	85%	0.15 pixel
3 rd test	After slope compensation & region selection	96%	0.04 pixel

In this paper, we propose the method of block code generation for model code recognition in de-interlacing image. Whenever a new model is produced, we have to input the new model in database in factory. In that case, we have to generate the exact model code for the correct classifying of model. For experiments, we work with 100 images of outsole per each black, read, white color. Through experiments, we find that the average model recognition ratio about extracted models is about 95% as table 2. Further, we find that the extraction and the recognition ratio are different according to color of outsole. Because of similarity between background and outsole color, especially, the case of black shows the lowest extraction and recognition ratio.

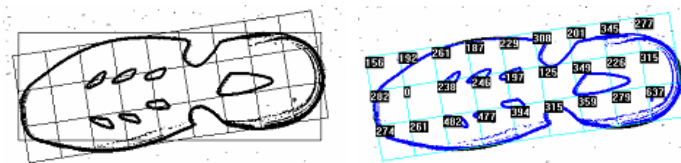


Fig. 5. Result of 1st transformation and result of model code generation

Table 2. Results for model recognition

Total number of data	Success of extraction	Failure of extraction	Number of recognized data & ratio (%)
Black (100)	87	13	80 (91.9 %)
Red (100)	95	5	91 (95.7 %)
White (100)	98	2	95 (96.9 %)

4 Conclusion

This paper proposed a model recognition scheme for length estimation of outsole in industry applications. As pre-processing procedure, we did de-interlacing processing based on edge detection and correlation for eliminating original interlacing of NTSC camera. And, through model recognition we make a possible for user to automatically select models for length estimation. The merit of proposed scheme is commonly applicable to various environments, color of objects and bright. It is expected that it is applicable to many application engineering in future.

References

1. G. S. Kim, C. K. Kim, E. Y. Cha.: Character Region Extraction using Model Recognition In Moving Images, Proceeding of JCM2000 (2000)
2. G. D. Haan, and E. B. Bellers.: Deinterlacing - an Overview, Proceeding of IEEE, Vol. 86, No.9 (1998) 1839-1857
3. C. J. Kuo, C. Liao, and C.C. Lin.: Adaptive Interpolation Technique for Scanning Rate Conversion, IEEE Transaction on Circuits System and Video Technology, Vol. 6. No. 3 (1996) 317-321
4. F.M. Wang, and D. Anastassiou, and A.N. Netravali.: Time-recursive deinterlacing for IDTV and pyramid coding, Signal Process. : Image Communication 2, (1990) 365-374
5. K.J. Sugiyama, and H.Y. Nakamura.: A Method of De-interlacing with Motion Compensated Interpolation, IEEE Transaction on Consumer Electronics, Vol.45. No.3 (1999) 611-616

R-functions Based Classification for Abnormal Software Process Detection

Anton Bougaev¹ and Aleksey Urmanov²

¹ Applied Intelligent Systems Laboratory, School of Nuclear Engineering,
Purdue University, 400 Central Dr., West Lafayette, IN, 47907, USA
bougaev@purdue.edu

² Sun Microsystems, Inc., 9515 Towne Centre Dr., San Diego, CA, 92121, USA
aleksey.urmanov@sun.com

Abstract. An R-functions based classification approach along with a regularization framework is proposed. The abnormal software process detection problem was used as the test bed. The R-functions based classification method is termed as the R-cloud method. The approach was validated both on synthetic and real-world data. Regularization allows to achieve good generalization and classification performance. In addition, the R-cloud approach gives the benefit of the analytical representation of the decision boundary. The introductory study on practical use of the R-cloud classifiers yielded promising results. The prototyping has shown that application of the R-functions based pattern recognition technique is a significant and practical tool for fault detection in providing fault tolerant computing.

1 Introduction

In the presented work, a piecewise linear classifier based on Rvachev functions (R-functions) is introduced. The method of R-functions based classification is termed as the R-cloud method. The R-cloud method was applied to the abnormal software process detection problem. Fault tolerance is the built-in capability of a computer system to provide continuous correct functionality in the presence of hardware or software faults. Fault detection is one of the important constituents of fault tolerance, which, in turn, defines the dependability of a computer system.

Various pattern recognition techniques using software metrics have been applied to diagnostics and prognostics in computer systems [1, 4]. However, there is no universal framework for the solution of these problems. One of the challenging issues in fault tolerant computing is the runaway process identification. A runaway process is a software process that no longer performs its function and continues to consume system resources. The overwhelming volume of information flows makes it virtually impossible for a system administrator to recognize precursors of the system performance degradation in a timely manner. Runaway processes can cause dramatic performance degradation of multi-task, multi-user systems. This calls for the usage of a reliable autonomous detection method to assist the system administrator in identifying runaway processes and in taking

appropriate actions. A detected runaway process can be proactively terminated to avoid degradation of the system's performance. The system administrator monitors the process statistics and makes decisions about terminating runaway processes using recommendation of a runaway process detector.

In order to achieve the required levels of availability without involving large number of system administrators, which is prohibitively expensive, the automatic abnormal process detection based on pattern classification can be used.

2 Classification Via R-functions

2.1 The R-cloud Method

R-functions are real valued functions whose sign is completely determined by the signs of their arguments [5, 6]. One of the important applications of the R-functions is the analytical representation of complex geometric shapes [8]. An object, which can be defined as a set of geometric primitives, can be represented by an R-function $R(\mathbf{x})$. The premise for the R-cloud classification method is to construct geometric objects in multi-dimensional space. A geometric object is represented by a domain in multi-dimensional space, which is bounded by surface $R(\mathbf{x}) = 0$. Multiple domains described by different R-functions can be combined into a single domain via R-conjunction and R-disjunction operations [5].

A standard two-class classification setting with samples obtained from a multi-dimensional input space \mathbf{X} is considered further. The training set \mathbf{D} is represented by labeled samples, $\mathbf{D} = \{\mathbf{C}_1, \mathbf{C}_2\}$, where \mathbf{C}_1 and \mathbf{C}_2 are the subsets of design (training) samples of Class 1 and Class 2 respectively; \mathbf{D} is a finite subset of \mathbf{X} . The class label function $c(\mathbf{x})$ specifies the class label for a sample $\mathbf{x} \in \mathbf{D}$ by mapping $c : \mathbf{D} \rightarrow \Lambda$, where $\Lambda = \{1, 2\}$ is the set of class labels. Given \mathbf{D} , it is required to construct an inference engine, which will be capable of classification of unknown samples or, in other words, of assigning appropriate class labels to previously unseen samples.

A function $\rho(\mathbf{x}, \mathbf{u}, \mathbf{v})$, where $\mathbf{x} \in \mathbf{X}$ and $\mathbf{u}, \mathbf{v} \in \mathbf{D}$, is called a separating primitive; the parameter vectors \mathbf{u} and \mathbf{v} is a pair of vectors from the training/design set, which belong to distinct classes. Only the separating primitives such that $\rho(\mathbf{u}, \mathbf{u}, \mathbf{v}) \geq 0$ and $\rho(\mathbf{v}, \mathbf{u}, \mathbf{v}) \leq 0$ are considered herein. For instance, in the case of a linear separating primitive in a multi-dimensional space, samples \mathbf{u} and \mathbf{v} will be separated by a hyperplane $\rho(\mathbf{x}, \mathbf{u}, \mathbf{v}) = 0$. The use of the linear separating primitive $\rho(\mathbf{x}, \mathbf{u}, \mathbf{v}, \gamma) = \mathbf{n}^T((\mathbf{v} + \mathbf{u})/2 - \gamma\mathbf{n} - \mathbf{x})$, where $\mathbf{n} = (\mathbf{u} - \mathbf{v}) / \|\mathbf{u} - \mathbf{v}\|$ is a vector such that $\|\mathbf{n}\| = 1$ and γ is an adjustable parameter, yields piecewise linear decision boundary.

A previously unseen vector $\mathbf{x} \in \mathbf{X}$ can be assigned with a class label using this separating primitive $\rho(\mathbf{x}, \mathbf{u}, \mathbf{v})$ in a following manner: \mathbf{x} is assigned the same class label as \mathbf{u} if $\rho(\mathbf{x}, \mathbf{u}, \mathbf{v}) \geq 0$, and \mathbf{x} is assigned the label of \mathbf{v} if $\rho(\mathbf{x}, \mathbf{u}, \mathbf{v}) < 0$. The separating primitive can be thought of as a basic (atomic) decision rule based on a pair of samples of distinct classes. Separating primitive is a function, which specifies the separating surface for previously unseen vectors \mathbf{x} , based on the information about only one pair of labeled samples \mathbf{u} and \mathbf{v} of distinct classes.

Next, utilizing the notion of the separating primitive, the notion of the separating bundle is introduced. The separating bundle function $\beta(\mathbf{x}, \mathbf{u})$ is defined as R-conjunction of all separating primitives of a vector $\mathbf{u} \in \mathbf{C}_k$ with respect to all other vectors \mathbf{v} of the training set such that $\mathbf{v} \notin \mathbf{C}_k$. Formally,

$$\beta(\mathbf{x}, \mathbf{u}) = \bigwedge_{\mathbf{v} \notin \mathbf{C}_k} \rho(\mathbf{x}, \mathbf{u}, \mathbf{v}). \tag{1}$$

The surface defined by $\beta(\mathbf{x}, \mathbf{u}) = \mathbf{0}$ separates \mathbf{u} of Class k ($\mathbf{u} \in \mathbf{C}_k$) from all vectors $\mathbf{v} \notin \mathbf{C}_k$. Further, the aggregation of all the separating bundles via R-disjunction yields an R-cloud classifier which is used to classify previously unseen samples or patterns. In the two-class pattern classification problem the R-clouds associated with Class 1 and 2 will be:

$$R_1(\mathbf{x}) = \bigvee_{\mathbf{u} \in \mathbf{C}_1} \bigwedge_{\mathbf{v} \in \mathbf{C}_2} \rho(\mathbf{x}, \mathbf{u}, \mathbf{v}), \quad R_2(\mathbf{x}) = \bigvee_{\mathbf{v} \in \mathbf{C}_2} \bigwedge_{\mathbf{u} \in \mathbf{C}_1} \rho(\mathbf{x}, \mathbf{v}, \mathbf{u}). \tag{2}$$

Once the R-clouds are constructed, previously unseen patterns can be classified by evaluating the corresponding R-cloud function. Simplified decision making can be performed in the following manner. An unknown sample \mathbf{x} is assigned the label of Class 1 if $R_1(\mathbf{x}) > 0$ and the label of Class 2 otherwise.

The locus of points \mathbf{x} such that $R_k(\mathbf{x}) = 0$ represents the surface bounding the R-cloud of class k . If linear separating primitive is chosen and parameter γ is set to 0, then the R-cloud classifier is equivalent to Euclidean distance based 1-nearest neighbor (1-NN) method. The R-function representation of the 1-NN classifier yields separating boundary with guaranteed differential properties [7]. The R-cloud associated with class k is a region in the multi-dimensional space, which is built around the design samples \mathbf{C}_k of this class using a given separating primitive function ρ and given operations of R-disjunction and R-conjunction. The term R-cloud was chosen to refer to this structure because in the cases of two or three dimensions the surface defined by $R(\mathbf{x}) = 0$ bounding these domains indeed resembles the cloud(s) visually.

2.2 Data Overfitting and Regularization

In statistics, data overfitting refers to the situation of the utilization of a statistical model that has too many adjustable parameters. Most of the pattern recognition methods are susceptible to overfitting. Overfitting leads to unnecessarily overcomplicated decision boundaries and decision making rules [2]. An overfitted classifier does not grasp the general structure of the data and demonstrates poor performance on previously unseen data. A straightforward application of the R-cloud classifiers may lead to overfitting. This situation is illustrated in Fig. 1a. A piecewise linear R-cloud was built to classify two-dimensional classes. The data in two classes are distributed normally. This plot depicts 150 test samples of each class and the separating surface corresponding to the R-cloud function $R_1(\mathbf{x})$ which is constructed using 40 labeled design samples of each class (not shown on the plot). The non-regularized separating surface is rough,

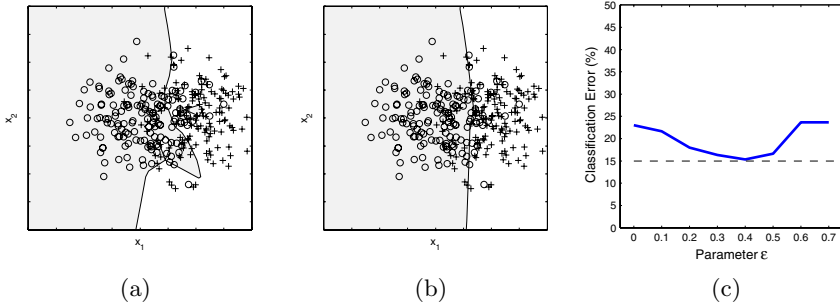


Fig. 1. Example of non-regularized (a) and regularized (b) R-cloud separating boundary for a two-class problem with two-dimensional Gaussian class distributions. Classification error curve (c). The optimal ϵ corresponds to the minimal error evaluated on the test set. The known Bayes error of the dataset is 15% (dashed line).

has cavities in it and does not grasp the underlying data structure. Thus, proper model complexity control has to be implemented to avoid possible overfitting. In order to construct a regularized R-cloud classifier the following procedure of sifting of a training data set is proposed. First, the value of the R-cloud function $R(\mathbf{x})$ is evaluated and compared to classification threshold ϵ . Next, all the data points that were misclassified are removed from the training set and a new R-cloud decision boundary is built using the reduced set of training vectors $\mathbf{D}_{R,\epsilon} = \{\mathbf{x} \in C_1 | R(x) > \epsilon\} \cup \{\mathbf{x} \in C_2 | R(x) < -\epsilon\}$.

In such a way the entire training/design set of samples will be divided into two subsets: subset $\mathbf{D}_{M,\epsilon}$ consisting of misclassified design samples and subset $\mathbf{D}_{R,\epsilon}$, consisting of samples, which were classified correctly with the threshold ϵ , $\mathbf{D} = \mathbf{D}_{R,\epsilon} \cup \mathbf{D}_{M,\epsilon}$. Each fixed value of ϵ produces one decision boundary. A manifold of decision boundaries is produced by variation of ϵ from 0 to some positive value ϵ_{max} . The classification performance evaluated as misclassification error on previously unseen (test) data will vary as ϵ varies. This procedure of the mean error evaluation is repeated for various values of ϵ . Then $\epsilon = \epsilon_o$ corresponding to the minimal error on the training set is identified. The R-cloud corresponding to ϵ_o yields a regularized classifier with the smoothed decision boundary.

2.3 Synthetic Data Experiment

Two-dimensional class data distributed normally were generated to have the Bayes classification error of 15%. The data generation procedures and the simulation experiment settings were kept close to the experiments described in [3]. The overfitted and regularized decision boundaries obtained using a piecewise linear R-cloud classifier are shown in Fig. 1a and Fig. 1b respectively. The shaded area corresponds to the R-cloud of Class 1.

The effect of regularization is illustrated in Fig. 1c. The minimal error evaluated on the training set corresponds to the minimum of the classification error

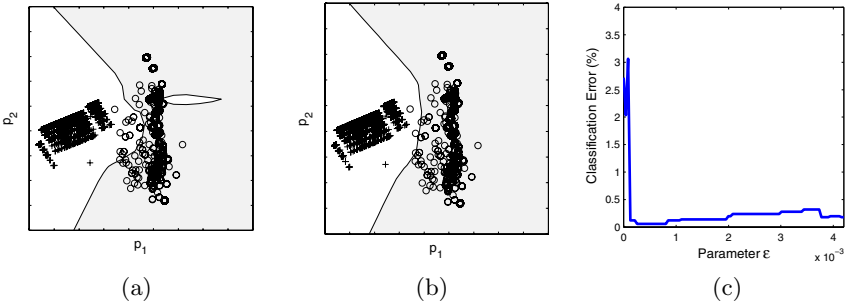


Fig. 2. Runaway software process detection: non-regularized (a), regularized (b) separating boundaries, and classification error curve (c) for the runaway software process detection problem

curve ($\epsilon = 0.4$). The classification error was reduced by about 7% with respect to the non-regularized classifier. The reduction of the classification error directly translates into the reduction of false and missed alarm probabilities.

2.4 Detection of Abnormal Software Processes

The abnormal software process detection problem with real-world data was chosen as the test bed. The snapshots of the software metrics associated with a given software process are taken to assess the validity of the state of this process. Historic data were collected for a number of Netscape Web browser software processes. The data points were labeled by a group of software experts. The labeled data set was utilized to set up an R-cloud classifier capable of discriminating new unlabeled data points into two classes of normal and abnormal process.

Using Principal Component Analysis (PCA) technique, in Fig. 2a and Fig. 2b the dimensionality of the dataset was reduced to only two principal components for illustration purposes. The shaded area corresponds to the R-cloud of the class of normal processes. Usually, in order to satisfy the classification performance requirements, 3 to 4 principal components should be used in the practical implementation.

The typical behavior of normal Netscape processes is shown by the circles. The abnormal process (shown by the crosses) initially was in the area where all the normal processes are. However, it has departed from the normal behavior at a certain point in time. The non-regularized decision boundary is shown in Fig. 2a. In this experiment, 40 process state snapshots corresponding to different Netscape processes were used as a design (training) data set and 2500 snapshots of each class were used as the test data set. The sampling rate was two snapshots per minute. After utilization of the regularization procedure proposed earlier, the irregularities of the original decision boundary were eliminated to a great extent (Fig. 2b). The mean classification error curve is shown in Fig. 2c. The classification performance was increased, which is expressed in reduction of the classification error from 2.7% to 0.1%.

3 Discussion and Conclusions

A novel nonparametric classification method termed R-cloud is introduced. The R-cloud gives an implicit representation for the separating surface with guaranteed differential properties.

A regularization framework for the R-cloud classification was introduced and validated. The regularization of the R-functions based classifier allows for the building of simpler decision boundaries and for achievement better classification performance. The R-functions representation of the separating surface was found to be a flexible technique. Further investigation is required in order to assess the potential of the R-cloud classification.

Acknowledgments

The authors thank Lefteri Tsoukalas for the insightful guidance of this research effort, Kenny Gross for many useful discussions, and Joshua Walter for valuable comments during the preparation of this manuscript.

References

1. Bougaev, A.: Pattern recognition based tools enabling autonomic computing. In Proceedings of the Second International Conference on Autonomic Computing, Seattle, WA, USA (2005) 313–314
2. Duda, R. O., Hart, P. E., Stork, D. G.: Pattern classification. 2nd edn. Academic Press, Boston (2001)
3. Fukunaga, K.: Introduction to statistical pattern recognition. 2nd edn. Wiley, New York (1990)
4. Lou, S.J., Budman, H, Duever, T.A.: Comparison of fault detection techniques. Journal of Process Control, **13(5)** (2003) 451–464
5. Rvachev, V.: Geometric applications of logic algebra. Tekhnika, Kiev (1967)
6. Shapiro, V.: Theory of R-functions and applications: A primer. Technical Report No. TR91-1219, Cornell University, Computer Science Department, Ithaca, NY (1991)
7. Shapiro, V., Tsukanov, I.: Implicit functions with guaranteed differential properties. In Proceedings of the Symposium on Solid Modeling and Applications, ACM, New York, NY, USA (1999) 258–269
8. Shapiro, V.: Real functions for representation of rigid solids. Computer Aided Geometric Design, **11(2)** (1994) 153–175

A Unified Framework for Shot Boundary Detection^{*}

Bing Han, Xinbo Gao, and Hongbing Ji

School of Electronic Engineering, Xidian University, Xi'an 710071, China
hanbing@lab202.xidian.edu.cn

Abstract. According to drawbacks of available algorithms, a new hierarchical and multiresolution approach to the detection and classification of scene breaks in video sequences is presented in this paper. This method gives a unified framework for different shots by using the multi-resolution analysis as mainstay. Firstly, the video clips are cut by FCM clustering method, then the fade and dissolve are detected using ECR and SCD algorithm in the high components and the low components respectively by integer-to-integer wavelet transform. Finally, according to the cut and the former detection of gradual changes, we use motion vectors in high-components of 3D-WT to detect the wipe transition. Experimental results with real video clips about 7h demonstrate that our method can detect and classify a variety of scene breaks, including cuts, fades and dissolves, even in sequences involving significant motions and flash.

1 Introduction

Generally, shot transitions include abrupt transition and gradual transition which includes Fadein, Fadeout, Dissolve and Wipe. Thus, it can be seen that gradual transition makes the shot boundary indistinctive. So far, many methods have been proposed for abrupt transition detection by using pixel-wise difference, histogram comparison, edge comparison and motion information. A comparison of these algorithms can be found in [1]. However, these methods available need to predetermine a threshold, if the measured dissimilarity is greater than this threshold, a shot boundary is assumed, which increases the interaction with human and cannot adapt to amount various video.

Then, few works for the detection of gradual transitions have been reported [2, 3, 4, 5]. Robust detection of gradual transitions is still an open issue. H. J. Zhang [2] proposed a twin-comparison algorithm for solving the problem of searching the positions of gradual transitions. However, it is difficult to select a proper threshold for all video sequences, while improper threshold affects the performance of the algorithm seriously. Histogram-based method [3] has been one of the most frequently used algorithms for video segmentation. Since it only uses the statistics of the luminance and color and does not take into account the information of pixel position, it has better noise tolerance characteristics than template matching-based

^{*} This work was supported by National Natural Science Foundation of China (No.60202004) and the Key Project of Chinese Ministry of Education (No.104173).

method. But it will fail in the case which similar histograms but different structures. Wu *et al.*[4] computed the standard deviation of projected pixelwise difference from each DC-image. The resulting 1-D plateau is compared with predefined threshold to identify wipe transition. Though wipe can be detected by this method, the type of wipes can not be identified and the corner-to-corner wipe transition is missed. Most of these methods do not take into account the three characteristic of video sequences, temporal-spatial and relativity in time domain, and they are sensitive to the motions of objects and camera. In addition, available methods are very efficient for one type of video shot transitions. There is not a universal and efficient algorithm for all types of shot segmentation. Certainly, some researchers have proposed a detection scheme by combining the different method for different shots simply [5], but a integrated detection system can-not be constructed. Otherwise, due to great amount of video data, computer cannot deal with these great amount data once. To this end, this paper presents a hierarchical and multiresolution scheme for different shot transition detection based on cutting before detection and fusion of various techniques by using the multiresolution characteristic of the wavelet.

2 A Unified Framework for Shot Boundary Detection

Figure 1 illustrates our unified framework for shot boundary detection, that is, hierarchical and multiresolution video segmentation system structure. According to the difficulty of different shot transition to detect, the scheme includes four hierarchy, that is, abrupt, fade, dissolve and wipe transition.

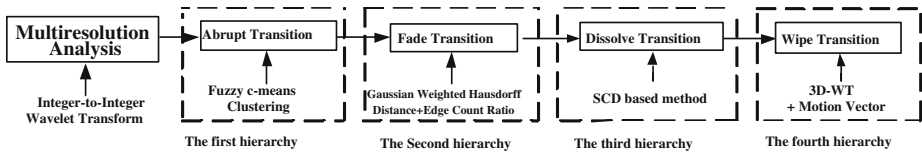


Fig. 1. The hierarchical scheme for shot boundary detection

2.1 Abrupt Transition Detection

Due to the amount of video data, the literature [6] presents the idea of subsection which there are most 400frames in every subsection to economize the computer memory. To partition video discretionally without considering the video content leads to the miss and false detection, which a gradual transition will be cut. However, this paper adopts the idea of cut before detection, that is, the video will first be cut several subsection by abrupt transition. This method cannot cut the gradual transition. So the storage of computer is decreased greatly and detection accuracy is increased. Here, we select the fuzzy c-means method [7] to cut shot in the feature space, which is spanned by the spatial difference based metric (SDM)

and the quantized histogram difference metric (QHDM). The definition of SDM can be found in authors other papers[7]. Through the experimental comparison, the color histogram we used is a $32 \times 10 \times 16$ 3D histogram in the HSV color space. Let $h_c(t)(c \in H, S, V)$ denote the color histogram of the hue, saturation or value component in HSV model of the t th frame. Then the quantized color histogram difference is defined as

$$D_h = \frac{1}{M \times N} \left[\sum_c \alpha_c \cdot \sum_{k=1}^{n_c} |h_c(t) - h_c(t + 1)|^2 \right]^{\frac{1}{2}} \tag{1}$$

where, k is one of possible colors levels n_c and α_c is the weighting coefficient. The weights can be used to tune the QHDM for the different video sequence.

For different types of video sequences, we can adjust α_c to make the dissimilarity function adapt to various conditions. For example, if we know that the video clips are taken under the same lighting conditions (such as news video clips), then we set $\alpha_h > \alpha_s$ and $\alpha_h > \alpha_v$ in order to penalize shifts in the average color. If video sequences are clips of the subsea world, it is very likely that the light is very weak. Generally, we first make the light compensation for this video in other algorithms. While, there are great changes in light and hue, especially light, to extract audience attention. So, effects of light sudden changes caused by flash and blast are decreased by adjusting weights, that is $\alpha_v > \alpha_h > \alpha_s$. For the films, due to the content of film is uncertainty, the weights is tuned according to the idiographic situation. Such as, the scene is often darkness in the war film. We enactment $\alpha_v > \alpha_h$, $\alpha_v > \alpha_s$, which make the light is dominant in shot changes. Now, the judgment of weights is by manual method above and the type of videos must be preknown, that is unblind judgment of weight coefficients.

2.2 Fade Transition Detection

Gaussian Weighted Hausdorff Distance(GW-HD)

Let $A = \{a_1, a_2, \dots, a_p\}$ and $B = \{b_1, b_2, \dots, b_q\}$ be two finite point sets, where p and q represent the numbers of elements in sets A and B , respectively. A Gaussian weighted HD (GW-HD) is computed as

$$h(A, B) = \sum_{m=1}^M \alpha_m \cdot h(A_m, B_m) \tag{2}$$

the directed measure $h(A, B)$ is defined by $\max_{a \in A} \min_{b \in B} \|a - b\|$, with $\|\cdot\|$ denoting the norm on the points of A and B [8].An image can be divided into m blocks, and the contributions of different blocks of the character image to HD are not equal. $\alpha_m(m = 1, 2, \dots, M)$ is weighted Gaussian coefficient defined as

$$\alpha_m = \frac{1}{\sqrt{2\pi}\sigma_m^2} \exp\left\{ - \frac{(i_m - i_0)^2 + (j_m - j_0)^2}{2\sigma_m^2} \right\} \tag{3}$$

where, (i_m, j_m) and (i_0, j_0) are the central pixels in the m th block and the original image, respectively. Psychological research reveals that the center of an

image is more important for human beings. The significance of central regions is strengthened by adjusting the weights, thus, the performance improvement of detection. Then motion compensation is made by GW-HD and Edge Change Ratio(ECR)[8] is used to detect a fade transition.

2.3 Dissolve Transition Detection

The dissolve transition is detected by the Similarity of Color Distribution(SCD) method in possible gradual transitions where fade transitions have been eliminated. This method can be found in our foregoing paper [7]. The dissimilarity function of color distribution between two successive frames is defined as:

$$dis(P, R) = \sum_{c \in \{H, S, V\}} (\omega_{1c}|E_c(P) - E_c(R)| + \omega_{2c}|\sigma_c(P) - \sigma_c(R)| + \omega_{3c}|S_c(P) - S_c(R)|). \tag{4}$$

where, E_c, σ_c and S_c is the first moments (mean), the second and the third central moments (variance and skewness) of each color channel respectively. $\omega_c, c \in \{H, S, V\}$ is the weighting coefficient. Now, we can tune the weights to make the dissimilarity function adapt to different videos. The coefficients adjustment can be dealt with through the flowchart in Fig. 2.

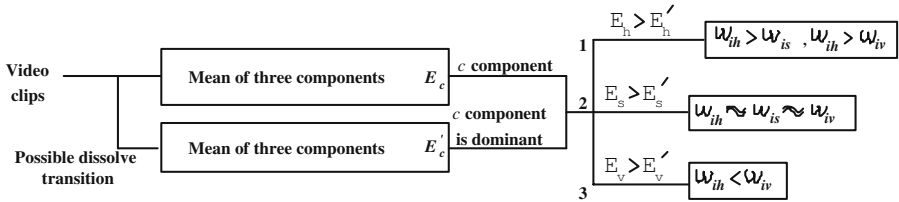


Fig. 2. The flowchart of automatic tuned coefficients

If changes of light in video clips is not obvious(such as news video), we set the weights $\omega_{ih} > \omega_{is}, \omega_{ih} > \omega_{is}$ to compensate the deficiency of color components. If the input video sequences is darkness video clips (such as sea ground), methods available must compensate the light component first before detection, but in our method, we only set a smaller value of ω_{ih} and a bigger value of ω_{iv} . So, the computation complexity is decreased. Generally, the weight of color component has a higher value than that of others in the HSV color space to match the original videos more better in color than in hue and saturation. Certainly, the judgment of weight ω_{ic} can refer to We call it the automatic judgment of weights, that is the blind judgment of weights.

2.4 Wipe Transition Detection

There are a number of more sophisticated variations and patterns of wipes including linear and nonlinear wipes. Here, the focus is placed on linear wipes

because of its common use in video editions, including left-to-right, right-to-left, top-down, bottom-up and corner-to-corner transitions. There are three parts in our detection scheme, which are the strip determination of wipe, the type judgment of wipe and the post-processing. The detailed content is referred to the authors other papers [9].

3 Experimental Results

Our method is applied to news, film and other commercial TV program video sequences. To verify the effectiveness of our method, we conduct an experiment with the method of the literature[8] on the same video clips. The experimental results are summarized in Table 1. Recall and Precision are chosen as the appropriate evaluation criteria. Their definitions are shown in the literature[7].

Table 1. The comparison of our method with the method of literature[8]

Program video		Film1	Film2	Advertisement	News	Total
Duration		136'28"	84'12"	50'24"	141'54"	412'58"
Video shots		1076	1040	772	1527	4415
Hits		1060	1160	832	1505	4557
Misses		58	40	8	53	159
Proposed method	False alarms	42	160	68	31	301
	Recall	94.7%	96.2%	98.9%	96.5%	96.4%
	Precision	96.1%	86.3%	91.8%	97.9%	93.5%
	Hits	1131	1100	839	1570	4640
	Misses	81	140	56	130	407
The method of literature[8]	False alarms	136	200	123	173	632
	Recall	92.4%	86.5%	92.7%	91.4%	90.8%
	Precision	87.9%	81.8%	85.3%	89.0%	86.4%

Based on the 4,415 shot boundary detected, we achieve 96.4% recall with 93.5%in precision. And there are three types of false detections in our method. One results from the existence of irregular camera operations during the gradual transitions. The second is due to a lot of flash effects in a shot. The misses are mainly due to the small content change between the frame pairs at some shot boundaries. The other is due to the "transparence" effect between two shots. Similarly, these false alarms and misses are occurred in the method of literature[8]. In addition, there exist other false alarms and misses in the method of literature [8]. Such as, there are a lot of false in the films caused by lots of object motions, which are detected as dissolves falsely. Nevertheless, misses of dissolve in the advertisement are so many that the accuracy is reduced because the duration of dissolve is too short to identify them and the feature of motion effects is much higher than that of dissolves in the curve shown in the picture.

4 Conclusions

This paper presents a unified framework based on the hierarchical idea for shot boundary detection by using the multi-resolution analysis as mainstay. This framework synthetically uses techniques of wavelet, FCM clustering and the proposed algorithms, Gaussian Weighted Hausdorff Distance, the method based on Similarity of Color Distribution and motion vector based on the three-dimension wavelet transformation. This hierarchical scheme detects not only abrupt and gradual transition but also their exact location. Then, this method presented in this paper does not be connected together simply but be combined efficiently by multiresolution analysis. In addition, the proposed method can restrain the effect of flash and various motions effectively. In this paper, we can tune the weighting coefficient to make the dissimilarity function adapt to various video sequences and the scheme of unblind and blind (automatic) adjustment for weighting coefficients are given. Consequently, this framework provides a new idea for shot boundary detection and makes a good base for video content analysis and video indexing in the following. Experimental results on about 7h videos significantly larger than most video experiments in the literature have shown that both the abrupt and gradual transitions are highly efficient.

References

1. Rainer Lienhart: Comparison of Automatic Shot Boundary Detection Algorithms. SPIE Conf.on storage and retrieval for image & video debase VII, **3656** (1999) 290–301
2. H. J. Zhang, S. W. Smoliar: Developing Power Tools for Video Indexing and Retrieval. Proceedings of the SPIE: Storage and Retrieval for Image and Video databases II, **2185** (1994) 140–149
3. Hampapur, A., Jain, R., Weymouth, T. E.: Production Model Based Digital Video Segmentation. Multimedia Tools and Applications, **1(1)**(1995) 9–46
4. Min Wu, Wyne Wolf, Bede Liu: An Algorithm for Wipe Detection. IEEE Inter. Conf. on Image Processing (ICIP'98), Chicago, (1998)
5. Qi wei, zhong yuzhuo: Approach for shot segmentation using Mpeg compressed data. Journal of Tsinghua University(Sci. & Tech.), **37(9)** (1997) 50–54 (in chinese)
6. Zhu Xingquan, Xue Xiangyang, Wu Lide: An Automatic Threshold Detection Method in Video Shot Segmentation. Journal of Computer Research & Development, **37(1)** (2000) 80–85(in chinese)
7. Han Bing, Gao Xin-bo, Ji Hong-bin: An efficient algorithm of gradual transition for shot boundary segmentation. SPIE on MIPPR,**5286(2)** (2003) 956–961
8. Ramin Zabih, Justin Miller, Kevin Mai: A Feature-Based Algorithm for Detecting and Classifying Production Effects. Multimedia Systems, **7**(1999) 119–128
9. Han Bing, Ji Hongbing, Gao Xinbo: A 3d Wavelet And Motion Vector Based Method For Wipe Transition Detection. International Conference on Signal Processing, **2**(2004) 1207–1210.

Image Recognition with LPP Mixtures

SiBao Chen, Min Kong, and Bin Luo*

Key Lab of Intelligent Computing & Signal Processing of Ministry of Education,
Anhui University, Hefei 230039, China
joysbc@163.com, minkong9546@sina.com, luobin@ahu.edu.cn

Abstract. Locality preserving projections (LPP) can find an embedding that preserves local information and discriminates data well. However, only one projection matrix over the whole data is not enough to discriminate complex data. In this paper, we proposed locality preserving projections mixture models (LPP mixtures), where the set of all data were partitioned into several clusters and a projection matrix for each cluster was obtained. In each cluster, We performed LPP via QR-decomposition, which is efficient computationally in under-sampled situations. Its theoretical foundation was presented. Experiments on a synthetic data set and the Yale face database showed the superiority of LPP mixtures.

1 Introduction

Principal component analysis (PCA)[7] and linear discriminant analysis (LDA) [1] are two fundamental methods for image recognition. PCA aims to find a mapping which preserves total variance by maximizing the trace of feature variance. LDA seeks a mapping which preserves discriminant information by maximizing between-class scatter meanwhile minimizing within-class scatter. Given sufficient sample, LDA is superior to PCA. While for a small sample size problem, PCA can outperform LDA because LDA is sensitive to the training data set [5].

Now a new technique, namely Laplacianfaces [3], has appeared in recent literature. Laplacianfaces is based on a technique called Locality Preserving Projections (LPP), which finds an embedding that preserves local information, and obtains a face subspace that best detects the essential face manifold structure. Experiments showed that LPP has a good discrimination performance.

However, LPP has only one transformation matrix. For the problem of many classes with high variations, only one transformation matrix is not sufficient for a good discrimination. Motivated by the idea of LDA mixtures [4], we propose LPP mixtures, which give several projection matrices. We partition the set of all data into several clusters and apply LPP technique for each cluster.

Another drawback of LPP is that if training samples are not enough and data dimension is high especially for image data, LPP can't be used directly in each cluster due to singularity of matrices. Recently Ye and Li [8] proposed LDA/QR algorithm, which avoids singularity by introducing QR-decomposition

* Corresponding author.

on a small-size matrix. Motivated by the idea of it, we perform LPP via QR-decomposition in each cluster, which shows its efficiency and high recognition performance in under-sampled situations with high dimension data.

2 LPP Mixture Model

LPP mixture model needs the partition of the set of all classes data into several clusters. We take PCA mixture model for data partitioning as it can divide the entire data accurately into many constituent components, which will make the formulation of LPP mixture model simpler. The detail of PCA mixture model is in [6], which applied EM algorithm [2] to estimate parameters. A simplified form of it is in [4]. PCA mixture model is used to partition the set of all data into an appropriate number of clusters and LPP is applied to each cluster, independently.

After applying PCA mixture model, we obtain several clusters of training samples by posterior probability. For each cluster, we then apply the locality preserving projections algorithm via QR decomposition (LPP/QR). Note that QR decomposition is more efficient than SVD numerically. It takes QR decomposition of original data matrix and turns to solve generalized eigenvector problem of matrices with $(N_k \times N_k)$ size at most, where N_k is the number of training samples in the k th cluster. This algorithm is especially efficient for under-sampled problem of high dimension data such as images and text data, where the dimension of sample n is greater than the number of training samples N .

2.1 The Algorithm of LPP Via QR Decomposition (LPP/QR)

LPP/QR is aimed to apply LPP method in under-sampled situations. The procedure of it in the k th cluster can be stated formally as the following:

1. **Constructing nearest-neighbor graph:** For the k th cluster, let G_k denote a graph with N_k nodes, the i th node corresponding to the i th data point x_i in k th cluster. We put an edge between nodes i and j if x_i and x_j are "close". There are several methods to measure "close". Here are two:
 - (a) k nearest neighbors. Nodes i and j are connected by an edge if i is among k nearest neighbors of j or j is among k nearest neighbors of i .
 - (b) ε -neighborhoods. Nodes i and j are connected if $\|x_i - x_j\| < \varepsilon$ where $\|\cdot\|$ is just the L^2 norm in \mathbf{R}^n .

Note: Here we can add label information (if there is) of training samples to improve discriminant performance, which can be done by restricting k nearest neighbors of each data point to be from the same class.
2. **Choosing the weights:** If there is an edge between nodes i and j in graph G_k , put a similarity weight s_{ij} on it, otherwise let $s_{ij} = 0$. Then we get a sparse symmetric $(N_k \times N_k)$ similarity matrix S_k for the k th cluster. The similarity weight s_{ij} can be:
 - (a) Simple-minded. $s_{ij} = 1$ iff nodes i and j are linked by an edge.
 - (b) Heat kernel. If nodes i and j are linked, put $s_{ij} = \exp\{-\frac{\|x_i - x_j\|^2}{t}\}$, where t is a suitable constant. Here in this paper, we choose t to be a constant multiplied by the maximal distance of pairwise neighbor data points.

3. **QR decomposition:** Decompose the data matrix of the k th cluster $X_k = [x_1, x_2, \dots, x_{N_k}]$ using QR decomposition method, $X_k = Q_k R_k$, where $Q_k \in \mathbf{R}^{n \times r}$ has orthogonal columns, $R_k \in \mathbf{R}^{r \times N_k}$ is a full row rank matrix, and $r = \text{rank}(X_k)$.
4. **Eigenmap:** Compute the eigenvectors and eigenvalues for the generalized eigenvalue problem:

$$R_k L_k R_k^T v = \lambda R_k D_k R_k^T v \quad (1)$$
 where D_k is a diagonal matrix with $D_{ii} = \sum_j s_{ij}$. $L_k = D_k - S_k$ is the Laplacian matrix for the k th cluster.
5. **Computing projection matrix:** Let $V_k = [v_1, v_2, \dots, v_d]$, where $v_i, i = 1, \dots, d$ are the eigenvectors associated with the smallest d generalized eigenvalues of equation (1), ordered according to their magnitude $0 \leq \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_d$. Then the optimal projection matrix for the k th cluster is $W_k = Q_k V_k$.

These generalized eigenvalues of equation (1) are nonnegative because matrices $R_k L_k R_k^T$ and $R_k D_k R_k^T$ are both symmetric and positive semidefinite. Thus the embedding for the k th cluster is as the following:

$$x_i \mapsto y_i = W_k^T x_i. \quad (2)$$

where y_i is a d feature vector of data point x_i in the k th cluster. W_k is the $(n \times d)$ transformation matrix for the k th cluster.

After training by LPP/QR, a feature vector of each data point and a transformation matrix for each cluster is obtained. Then a nearest neighbor classifier can be used for classification (k -nearest neighbor classifier can also be used here). Suppose that y_1, y_2, \dots, y_N are feature vectors of training data x_1, x_2, \dots, x_N , and each data point has a class (identity) label C_k . Given a test sample x , we assign it to the class C of corresponding to the transformed training data that is nearest to the transformed test data by the corresponding transformation matrix W_k as

$$C = L \left(\arg \min_{x_i} \|W_{I(x_i)}^T (x - x_i)\| \right), \quad (3)$$

where x_i is a training sample, $L(x_i)$ indicates the class label and $I(x_i)$ indicates the mixture component label, which is generated by PCA mixture model.

2.2 Justification

From matrix computation knowledge, we know, for any $A \in \mathbf{R}^{m \times n}$, it can be decomposed into $A = Q L_1 L_2^T V^T$, where $Q \in \mathbf{R}^{m \times r}$ and $V \in \mathbf{R}^{n \times r}$ have unitary and orthogonal columns, $L_1 \in \mathbf{R}^{r \times r}$ and $L_2 \in \mathbf{R}^{r \times r}$ are nonsingular upper triangular matrices. $r = \text{rank}(A)$ is the rank of matrix A . Rewriting the formula we get $A = QR$, where $R = L_1 L_2^T V^T \in \mathbf{R}^{r \times n}$ is a full row rank matrix.

Now the training data matrix in the k th cluster $X_k = [x_1, x_2, \dots, x_{N_k}] \in \mathbf{R}^{n \times N_k}$ can be decomposed into $X_k = Q_k R_k$, where $Q_k \in \mathbf{R}^{n \times r}$ has unitary and orthogonal columns, $R_k \in \mathbf{R}^{r \times N_k}$ is a full row rank matrix. $r = \text{rank}(X_k)$ is the rank of matrix X_k . So $X_k L_k X_k^T = Q_k R_k L_k R_k^T Q_k^T$ and $X_k D_k X_k^T = Q_k R_k D_k R_k^T Q_k^T$.

First it can be seen that if $\hat{V} = \arg \max_{V^T V = I} \text{trace}(V^T R_k D_k R_k^T V)$ then $Q_k \hat{V} = \arg \max_{W^T W = I} \text{trace}(W^T X_k D_k X_k^T W)$. This is due to $Q_k^T Q_k = I_r$, and if v is the eigenvector of $R_k D_k R_k^T$ associated with eigenvalue λ , then

$$R_k D_k R_k^T v = \lambda v \Rightarrow Q_k R_k D_k R_k^T Q_k^T Q_k v = \lambda Q_k v \Rightarrow X_k D_k X_k^T Q_k v = \lambda Q_k v.$$

which says that $Q_k v$ is the eigenvector of $X_k D_k X_k^T$ associated with eigenvalue λ . This implies that maximizing $\text{trace}(W^T X_k D_k X_k^T W)$ can be achieved by maximizing $\text{trace}(V^T R_k D_k R_k^T V)$, $W_k = Q_k V$.

Specifically, we take a relaxation on V . That is, we seek for a transformation matrix W_k such that $W_k = Q_k V$ for any matrix V . Now the original problem of finding W_k is equivalent to looking for V . Since $W^T X_k L_k X_k^T W = V^T R_k L_k R_k^T V$ and $W^T X_k D_k X_k^T W = V^T R_k D_k R_k^T V$, the original optimization on finding optimal W_k is now turned to finding V , such that

$$V_k = \arg \min_V \text{trace}((V^T R_k D_k R_k^T V)^{-1} (V^T R_k L_k R_k^T V)). \tag{4}$$

Note that D_k is full rank and R_k is full row rank, so $R_k D_k R_k^T$ is nonsingular, hence the optimization problem above is well defined and can be solved easily. It is exactly a generalized eigenvector problem and the optimal solutions are d generalized eigenvectors associated with the d smallest generalized eigenvalues of equation (1).

2.3 Model Selection

When using mixture models, we are faced with model selection problem. That is, how many mixture components we should use. One strategy to find the proper number of mixture components is cross-validation. We can set aside some of a training set as a validation set, and evaluate the performance for the validation set over different number of mixture components. We can use the number of mixture components which give the best performance for the validation set.

3 Experimental Results

The first experiment is performed on a synthetic two-class two-dimensional data set. The first class consists of points distributed uniformly on four small circles and the second class on two big ellipses uniformly as shown in Figure 1. Now we compare the discriminant performance of LPP mixture model with that of PCA, LDA, LPP, PCA mixture model and LDA mixture model by reducing the feature space dimension to one. Figure 1 shows their corresponding projection axes. PCA, LDA and LPP all obtain horizontal projection axis and so fail to discriminate the two class well. Three mixture models all partition the data set into two clusters. In each cluster, PCA mixture model seeks axis which maximizes the feature variance and obtains the primary axes shown as tilted dashed lines in Figure 1. LDA mixture model fails to obtain good projection axis in each cluster. The nearly vertical dotted lines are its projection axes. This is because that in each cluster, two classes share the same mean and between-class variance is zero.

The tilted solid lines are the projection axes obtained by LPP mixture model. Here we can see that projecting data points on tilted solid line in each cluster will partition the two class data perfectly well, which shows the discriminant ability of LPP mixture model outperforms that of other methods.

The second experiment is performed on the Yale face database (<http://cvc.yale.edu/projects/yalefaces/yalefaces.html>), which contains 165 images of 15 individuals, each person has 11 different images under various lighting conditions (center-light, left-light, right-light), facial expressions (happy, normal, sad, sleepy, surprised, wink) and wearing glasses (glasses,no-glasses). The size of original images is 243×320 . We cropped each image manually and resized to 64×64 pixels for this experiment.

Several methods are tested: Eigenface, Fisherface, Laplacianface, PCA mixture, LDA mixture and LPP mixture. Cross-validation strategy is adopted for

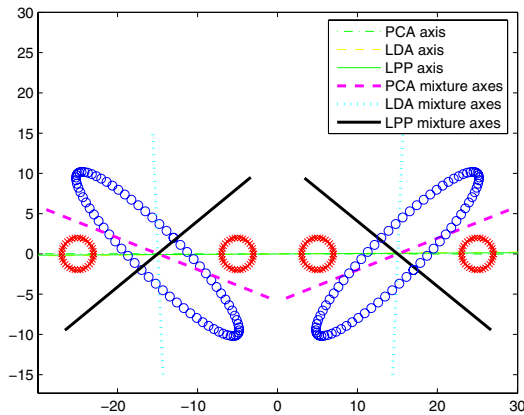


Fig. 1. Synthetic two-class data to test LPP mixture

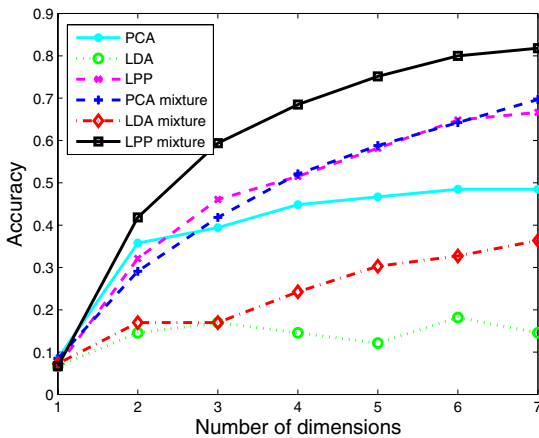


Fig. 2. Accuracy vs. number of features

recognition. In Laplacianface and LPP mixture we choose 3-neighborhood to construct similarity matrices. Experiment shows that there is no significant improvement in recognition performance when using more than two mixture components, which is just like the depiction in [4]. So we use only two mixture components for learning PCA mixture model, LDA mixture model and LPP mixture model. Here we evaluate the effect of reduced dimension on the classification accuracy. Figure 2 shows the results. As can be seen, LPP mixture shows its high recognition accuracy. In this experiment, the training samples are not enough and don't show representative due to the random choice method, Fisherface and LDA mixture fail to outperform Eigenface and PCA mixture. We can also note that PCA mixture, LDA mixture and LPP mixture outperform Eigenface, Fisherface and Laplacianface, respectively.

4 Conclusions

We proposed locality preserving projections mixture models (LPP mixtures), where the set of all data were partitioned into several clusters and a projection matrix for each cluster was obtained. In each cluster, We performed LPP via QR-decomposition, which is efficient computationally in under-sampled situations, especially when the number of dimension of data is greater than the number of training samples. Its theoretical foundation was presented. Experiments on a synthetic data set and the Yale face database showed that LPP mixtures outperform PCA, LDA, LPP, PCA mixtures and LDA mixtures.

Acknowledgements. This research is supported by the National Natural Science Foundation of China (No.60375010), the Excellent Young Teachers Program of the Ministry of Education of China, and Innovative Research Team of 211 Project in Anhui University, 2001Z021.

References

1. Belhumeur,P.N., Hespanha,J.P. and Kriegman,D.J.: Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Trans. PAMI* **19** (1997) 711-720
2. Dempster, P., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser.* **39** (1977) 1-38
3. He,X.F., Yan,S., Hu,Y., Niyogi,P. and Zhang,H.J.: Face Recognition Using Laplacianfaces. *IEEE Trans. PAMI* **27** (2005) 328-340
4. Kim,H.C., Kim,D. and Bang,S.Y.: Face recognition using LDA mixture model. *Pattern Recognition Letters* **24** (2003) 2815-2821
5. Martinez,A.M. and Kak,A.C.: PCA versus LDA. *IEEE Trans. PAMI* **23** (2001) 228-233
6. Tipping, M., Bishop, C.: Mixtures of probabilistic principal component analyzers. *Neural Comput.* **11** (1999) 443-482
7. Turk,M. and Pentland,A.P.: Face Recognition Using Eigenfaces. *IEEE Conf. Computer Vision and Pattern Recognition* (1991)
8. Ye,J.P. and Li,Q.: A Two-Stage Linear Discriminant Analysis via QR-Decomposition. *IEEE Trans. PAMI* **27** (2005) 929-941

Line-Based Camera Calibration

Xiuqin Chu, Fangming Hu, and Yushan Li

Department of Electronic and Engineering, Xidian University,
Xi'an 710071, China
Xqchu@mail.xidian.edu.cn

Abstract. In this paper, a new line-based camera calibration technique is presented. Firstly, the relation between dimensional line parameters and projective line parameters is deduced. Using more than three dimensional lines which are not parallel with each other, the camera intrinsic parameters can be determined by solving linear equations. Experiment results show that in same noise level the mean attained by this method is more closer to factual value and MSE is smaller by about 65% than classic point-based technique.

1 Introduction

Camera calibration is a way to determine the parameters in camera-imaging's geometry model by experimental and calculating methods^[5].

Researchers have presented many methods to calibrate the camera in the recent years. These methods may classify two kinds approximately^[1]. One kind is using the projection of objection of known structure in image to calculate the camera intrinsic parameters^[5]; the other obtains many images by controlling camera's motion accurately to calibrate the intrinsic parameters^[2-4]. Both kinds have advantage as well as shortcoming. One objection of known structure is needed in the first method and not needed in the other method; but camera should be controlled accurately to do some horizontal motion which are orthogonal with each other. So the requirement of camera platform's motion is much hypercritical in this kind.

The method presented in this paper belongs to the first one, but it is some different from the classic one. In the classic method, character point in calibration block and its projection in image are used to calculate the camera parameters. The noise of image point's effect is much big to the result. Firstly, the relation between the line in calibration block and its projection in the image is deduced in this paper. On the base of this foundation, using the correspondence of the line, the camera parameters are calculated by solving linear equations. Experimental result indicates that this method is simple and what's more, the precision and robustness are higher than the classic point-based technique.

2 Camera-Imaging's Geometry Model

Given the classic pinhole model, the rectangular coordinate system which is composed by the camera's optical axis and the image plane is called the camera

coordinate system. Because the camera may place in the environment's any position, a normalized coordinate system should be chosen in the environment to describe the camera's position, and this coordinate system is called the world coordinate system. The relation between the camera coordinate system and the world coordinate system may use a rotation matrix \mathbf{R} and a translation vector $\boldsymbol{\mu}$ to describe. Therefore, point P in the space has the following relation between camera coordinate system and the world coordinate system:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \boldsymbol{\mu} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \mathbf{M}_1 \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{1}$$

where, $[X_c \ Y_c \ Z_c \ 1]$ and $[X_w \ Y_w \ Z_w \ 1]$, the point of vector format, respectively, are described by the camera coordinate system and the world coordinate system; \mathbf{R} , 3×3 unitary matrix, are set to be orthogonal to each other; $\boldsymbol{\mu}$ is three dimensional translation vector; $\mathbf{0}=(0, 0, 0)^T$; \mathbf{M}_1 is 4×4 matrix.

(x, y) corresponding image pixel coordinates in the image plane, is produced by some point in object P (X_c, Y_c, Z_c) after perspective projection. Both of them have the following proportional relationship:

$$Z_c \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \tag{2}$$

The relation between the pixel element coordinates (u, v) and physical coordinates (x, y) is deduced:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3}$$

Substituting (1) and (2) to (3), we can obtain:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & u_0 & 0 \\ 0 & f_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \boldsymbol{\mu} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \mathbf{M}_1 \mathbf{M}_2 \mathbf{X}_w = \mathbf{M} \mathbf{X}_w \tag{4}$$

where \mathbf{M}_1 , only related to the camera's intrinsic construct, is called camera's intrinsic parameter matrix; \mathbf{M}_2 is determined by the position and orientation of the camera relative to the world coordinate system, called camera extrinsic parameter matrix. Camera calibration means that determining the intrinsic and extrinsic parameter matrix.

3 Algorithm of Line-Based Camera Calibration

Given one dimensional line in the world coordinate system and image plane, the equation is:

$$a_w x + b_w y + c_w z + d_w = 0 \quad a_\xi x + b_\xi y + c_\xi = 0 \tag{5}$$

Random point $[X_w \ Y_w \ Z_w \ 1]$ out of dimensional line in world coordinate system corresponds with some determined point $[X_\xi \ Y_\xi \ w_\xi]$ in the image plane. Furthermore:

$$\begin{bmatrix} a_w & b_w & c_w & d_w \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = 0 \quad \begin{bmatrix} a_\xi & b_\xi & c_\xi \end{bmatrix} \begin{bmatrix} X_\xi \\ Y_\xi \\ w_\xi \end{bmatrix} = 0 \tag{6}$$

Submitting equation(4) to equation(6) , then:

$$\begin{bmatrix} a_\xi & b_\xi & c_\xi \end{bmatrix} \begin{bmatrix} X_\xi \\ Y_\xi \\ w_\xi \end{bmatrix} = \begin{bmatrix} a_\xi & b_\xi & c_\xi \end{bmatrix} M \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = 0 \tag{7}$$

thereby, $\begin{bmatrix} a_\xi & b_\xi & c_\xi \end{bmatrix} M = k_0 \begin{bmatrix} a_w & b_w & c_w & d_w \end{bmatrix}$ (8)

Where k_0 is a proportional factor, and has no influence to the discussion followed. Shortening equation above as:

$$KM = W \tag{9}$$

where, 12 unknown quantities require to solve, and every line in the calibration block has four linear equations. Then matrix M only can be determined by having at least three known line. When $n \geq 3$, the above one can be solved by LMS:

$$M = (K^T K)^{-1} K^T W \tag{10}$$

Then, we can get from (4):

$$M = M_1 M_2 = \begin{bmatrix} f_u & 0 & u_0 & 0 \\ 0 & f_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & \mu \\ O^T & l \end{bmatrix} = \begin{bmatrix} m_1^T & m_{14} \\ m_2^T & m_{24} \\ m_3^T & m_{34} \end{bmatrix} \tag{11}$$

Where, m_1^T 、 m_2^T 、 m_3^T are the vectors composed by the first three lines and first three rows in the M matrix respectively. The parameter in matrix M1 and M2 can be determined by follow equations:

$$\begin{aligned}
 \mu_z &= m_{34} \\
 r_3 &= m_3 \\
 u_0 &= m_1^T m_3 \\
 v_0 &= m_2^T m_3 \\
 f_u &= \sqrt{(m_1 - u_0 r_3)^T (m_1 - u_0 r_3)} \\
 f_v &= \sqrt{(m_2 - v_0 r_3)^T (m_2 - v_0 r_3)} \\
 r_1 &= (m_1 - u_0 m_3) / f_u \\
 r_2 &= (m_2 - v_0 m_3) / f_v \\
 \mu_x &= (m_{14} - u_0 m_{34}) / f_u \\
 \mu_y &= (m_{24} - v_0 m_{34}) / f_v
 \end{aligned} \tag{12}$$

Then we can get the camera’s whole intrinsic and extrinsic parameters.

4 Simulation Experiment

In camera calibration, the noise of image point has much influence in the result of calibration. In the simulation experiment, we examine the noise of image point to the classic point-based algorithm and line-based one presented in this paper separately and compare the influence. The parameters in simulation experiment are: $f_u=1000$, $f_v=1000$, $u_0=0$, $v_0=0$, $\mu_x=60$, $\mu_y=80$, $\mu_z=40$, the rotation Φ in R matrix is 30° . Four corresponding lines are adopted in experiment and they are not parallel with each other. Adding random noise to the image point, we make every 100 times experiments in different noise level.

Graph1 and graph2 denote the mean and square error separately obtained by both methods in the different noise level. From both graphs, we can see that, the mean of parameter out of both methods much deviate from the factual value and the MSE also become much bigger, along with the increase of the random noise level. But the degree of mean’s deviation in line-based method is smaller than that in the classic one . Moreover, the increase of f_u and f_v ’s MSE in the line-based method is slower than the classic one ,along with random noise level ’s increasing gradually.

Results show that the MSE of f_u and f_v out of the line-based method are smaller by 67% and 65.6% than those out of the point-based one separately. The MSE of u_0 and v_0 are the same in basically. So line-based method presented in this paper has much less noise’ influence than the classic one. Therefore, the method in this paper has better robustness.

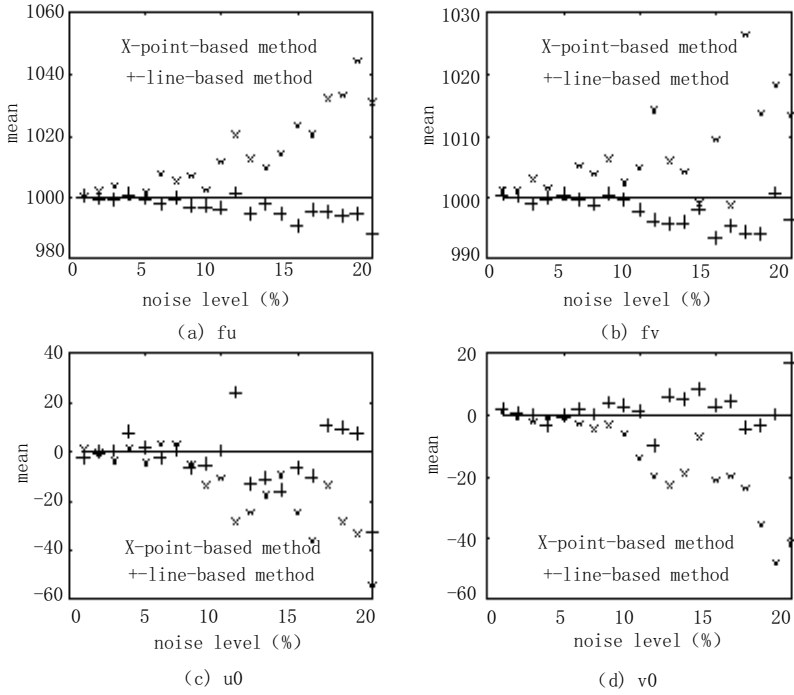


Fig. 1. Comparing with means in different noise level

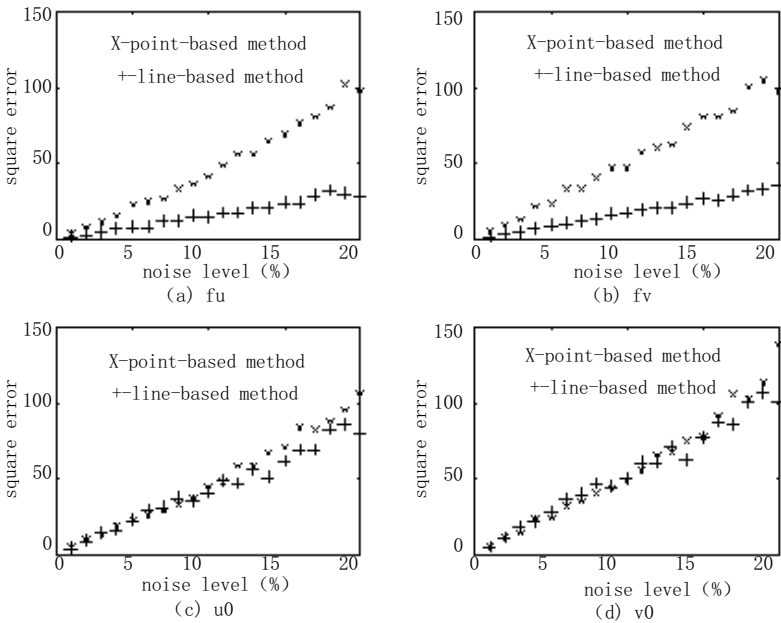


Fig. 2. Comparing with MSE in different noise level

5 Conclusion

Under the foundation of the relation between the dimensional line in the world coordinate system and the corresponding line in the image plane, the line-based camera calibration is presented in the process of deducing perspective projection. This method only require three more dimensional lines on the calibration block which has the known coordinate parameter and are not parallel with each other. The perspective projection is made in the calibration block. The corresponding line parameter in the image plane is obtained from the corresponding image. Using both line parameters, we can calibrate camera's intrinsic and extrinsic parameters.

From the result of experiment in this paper, we can see that under the same image point's noise level, mean out of the method presented in this paper is more close to the factual value and MSE is much smaller. Therefore, this method has higher precision and better robustness.

References

1. Zhang Z.: A Flexible New Technique for Camera Calibration. Microsoft Corporation:Technique Report MSR-TR-98-71 (1998)
2. Stephen J Maybank, Olivier D Faugeras. : A Theory of Self-calibration of a Moving Camera. Interational Journal of Computer Vision, Vol.8(2) (1992) 123-151
3. Quan L.: Conic Reconstruction and Correspondence from Two Views. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.18(2) (1996) 151-160
4. Yang Chang-Jiang, Wang Wei, Hu Zhan-Yi : An Active Vision Based Camera Intrinsic Parameters Self-calibration Technique. Journal of computers, Vol.21(5) (1998) 428-435
5. O.D. Faugeras and G. Toscani: The Calibration Problem for Stereo. Proc. Of IEEE Conference of Computer Vision and Pattern Recognition (1986) 15-20
6. Song-de Ma, Zheng-you zhang: Computer Vision---Compute Method and Algorithm Basis. Beijing Science and Technology Press (1998)
7. Zheng Nan-Ning: Computer Vision and Pattern Recognition. Beijing National Defence Industry Press (1998)

Shot Boundary Detection Based on SVM and TMRA

Wei Fang¹, Sen Liu², Huamin Feng³, and Yong Fang³

¹Telecommunication Engineering College, BUPT University, Beijing 100876, China
fangwei@besti.edu.cn

²Communication Engineering College, YanShan University, Qinhuangdao 066004, China
liusen@besti.edu.cn

³Key Laboratory of Security and Secrecy of Information, BESTI, Beijing 100070, China
{fenghm, fyong}@besti.edu.cn

Abstract. Video shot boundary detection (SBD) is an important step in many video applications. In this paper, previous temporal multi-resolution analysis (TMRA) framework was extended by first using SVM (Supported Vector Machines) classify the video frames within a sliding window into normal frames, gradual transition frames and CUT frames, then clustering the classified frames into different shot categories. The experimental result on ground truth, which has about 26 hours (13,344 shots) news video clips, shows that the new framework has relatively good accuracy for the detection of shot boundaries. It basically solves the difficulties of shot boundaries detection caused by sub-window technique in video. The framework also greatly improves the accuracy of gradual transitions.

1 Introduction

Since shots are the basic temporal units of video, SBD is the groundwork for video applications [4]. To fulfill the task of partitioning the video, video segmentation needs to detect the joining of two shots in the video and locate the position of these joins. These joins appear to be of two different types, abrupt transition (CUT) and gradual transition (GT).

Zhang et al. [9] developed one of the most successful early methods, called twin-comparison method, which detects CUTs and GTs by applying different thresholds based on differences of color histograms between successive frames. However, this, together with most existing methods, suffers a lot from threshold selection and noise.

Chua et al. [8] proposed the TMRA model. They extract the temporal feature of video; calculate the feature wavelet coefficients. By setting the adaptive threshold, they made use of characteristics of different resolutions to detect CUTs and GTs.

In this paper, we extend previous TMRA method [8] to incorporate Blocked Color Histogram (BCH) as feature vector. The video frame series have the temporal multi-resolution characteristics of shot presented by the wavelet transition coefficients. We use SVM classifier for pattern recognition. Tests on more than 26 hours (over 13,344 shots) video clips show that the resulting system is able to improve the precision of SBD while retaining high recall.

2 TMRA and Wavelet

2.1 Video Representation

We model the video according to the content of the video frames in the stream. The feature for representing the content of video frames could be of any types: color, shape, texture et al. Thus the video is modeled in a N-dimensional feature space. The dimension of the space depends on the dimensionality of the chosen feature space.

In our current work we chose color-based feature space. If we expressed every frame of the video stream as: $f = (x_1, x_2, \dots, x_n)$, where x_i is the i th bin in a histogram representation, then we could use $v = f_i = (x_{1i}, x_{2i}, \dots, x_{ni})$ to represent the video stream.

2.2 Applying Wavelet

Wavelet provides a good mathematical basis for video analysis. The Gaussian scale-space approach is widely adopted. This is because the Gaussian function is the unique kernel which satisfies the causality property as guaranteed by the scaling theorem. It states that no new feature points are created with increasing scale [8]. Because the first order derivative of the Gaussian function could be a mother wavelet, one can easily show that the sharper variation points of the signal correspond to the local maxima of the wavelet transform. Thus a maxima detection of the wavelet transform is equivalent to boundary detection. The magnitude of the maxima will show the strength of the transitions. We select the Canny-like B-spline wavelets since they have fast algorithm independent of the resolution, and the good features of Canny [8]. We project the video streaming onto the N-dimension space, do the wavelet transition to each dimension, and add the energies. By analyzing the sum coefficients, we can get the positions of CUTs and GTs.

3 SVM and Its Application in SBD

3.1 SVM

Support Vector Machines combine generalization control with a technique to address the curse of dimensionality. The formulation results in a global quadratic optimization problem with box constraints, which is readily solved by interior point methods. The kernel mapping provides a unifying framework for most of the commonly employed model architectures, enabling comparisons to be performed. In classification problems, generalization control is obtained by maximizing the margin, which corresponds to minimization of the weight vector in a canonical framework. The solution is obtained as a set of support vectors that can be sparse. These lie on the boundary and as such summarize the information required to separate the data [1].

We choose RBF kernel for three reasons: One reason is that the RBF kernel nonlinearly maps samples into a higher dimensional space, so it, unlike the linear kernel, can handle the case where the relation between class labels and attributes is nonlinear. The linear kernel is a special case of RBFs as in [7] showed that the linear kernel with a penalty parameter \tilde{C} has the same performance as the RBF kernel with

some parameters. In addition, the sigmoid kernel behaves like RBF for certain parameters [5]. The second reason is that the number of hyperparameters influences the complexity of model selection. The polynomial kernel has more hyperparameters than that of the RBF kernel. Finally, the RBF kernel has less numerical difficulties.

3.2 Applying SVM in SBD

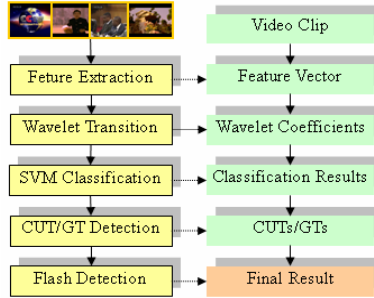


Fig. 1. System function blocks

3.2.1 System

Figure 1 shows schematic steps of our system function blocks. The following subsections discuss each of these functional blocks in details.

3.2.2 BCH Extraction

Color Histogram has been proved to be a good feature for SBD [3] [8] [9], and is easily computed, so we adopt color histogram as the basic feature. Our system first extracts histograms in the YUV color space. Global frame histograms and block histograms are extracted; the block histograms use 4×3 uniform spatial image grids. Then we can denote the BCH by vector V_H , respectively: $V_H = [v_{H1}, v_{H2}, \dots, v_{H13}]$.

3.2.3 Resolution Selection

All the BCHs corresponding with frames were jointed. Denote the matrix constructed by BCH vectors by M_H , respectively.

$$M_H = \begin{bmatrix} V_{1H1} & V_{1H2} & \dots & V_{1H13} \\ V_{2H1} & V_{2H2} & \dots & V_{2H13} \\ \vdots & \vdots & \ddots & \vdots \\ V_{NH1} & V_{NH2} & \dots & V_{NH13} \end{bmatrix} \Rightarrow M_W = \begin{bmatrix} V_{1W1} & V_{1W2} & \dots & V_{1W13} \\ V_{2W1} & V_{2W2} & \dots & V_{2W13} \\ \vdots & \vdots & \ddots & \vdots \\ V_{NW1} & V_{NW2} & \dots & V_{NW13} \end{bmatrix} \quad (1)$$

We calculate the wavelet coefficients of the column vectors in M_H . Denote the coefficients matrix as M_W . Each column vector of M_W is the wavelet coefficients of each block in frames.

It is very important to choose resolution in video temporal multi-resolution analysis. In a general way, if we choose low resolution, we will lose the details of the frame transitions; if we choose the high resolution, we will suffer the influence of

camera motions, and focus on changing or objects moving in shots. The third resolution was chosen in [2]. By testing through 0 to the 5th resolution, we found that the 4th resolution is better than the 3rd resolution combining with BCH and SVM.

3.2.4 Sliding Window Selection

Shot transitions are temporal random processes. We set a moving sliding window in the wavelet coefficient vectors. The durations of GTs are 2 to 60 frames, and the length of noise is no more than 10 frames. By experimenting, we set the window size to 11 frames. Matrix (2) illustrates the sliding window vector (SWV).

$$M_w \Rightarrow [SWV_1 \dots SWV_N \dots] \tag{2}$$

$$SWV_N = [V_{(N-5)W1} \quad V_{(N-5)W13} \quad \dots \quad V_{NW1} \quad V_{NW2} \quad \dots \quad V_{NW13} \quad \dots \quad V_{(N+5)W13}]$$

3.2.5 SBD Using SVM

It can be regarded as a problem of classification where a video frame is a CF/GF. There are three frame classes labeled NF, CF, and GF separately. We construct a 3-class SVM, and use the “one-against-one” [6] approach. The features of SVM include all the wavelet coefficients of blocks in the sliding window.

In classification we employ a voting strategy, in which 3 classifiers are constructed and each one train’s data from two different classes. Each binary classification is considered to be a voting where votes can be cast for all data points x - in the end point is designated to be in a class with maximum number of votes.

We join the result of SVM classification and get a series of frame labels which indicate NF, CF and GF, separately. Thus CUTs and GTs can be detected (Figure 2).

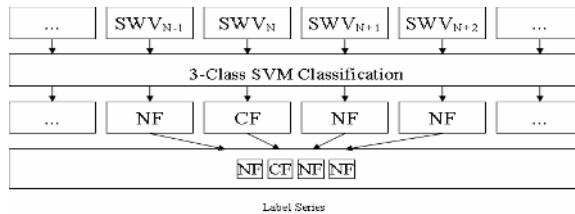


Fig. 2. The flow of SVM classification

3.2.6 Flash Detection

With TMRA, flash is easily detected by testing the changes of frame’s wavelet coefficients at all resolutions.

4 Experiment Results and Analysis

The effectiveness of the algorithm was evaluated on the CCTV9 (China Central TV Channel 9) news and CNN news (TRECVID 2005) using the precision, recall and F1 measures. We trained two SVMs on two sets of TV channels video clips separately.

4.1 Experiment 1

SWVs of 20 hours CCTV9 News videos were generated. The SVM was given the SWVs as the training set.

There are about 21 hours CCTV9 news videos (Data Set 1) contained about 8,365 CUTs and 1,885 GTs and about 5 hours CNN news videos testing data set (Data Set 2) with 2,289 CUTs and 805 GTs. Table 1 shows our system improves the F1 measure over the previous TMRA [8], where Data Set 1 was detected. Also the system is able to maintain high recall with only slight degradation in precision.

Table 1. Comparison of testing CCTV9 news videos results with the TMRA

Transition Type	TMRA			TMRA-SVM		
	Precision	Recall	F1	Precision	Recall	F1
CUTs	77%	98%	0.87	93.1%	95.4%	0.94
GTs	65.2%	97.1%	0.78	82.7%	80.1%	0.81
Average	71.5%	97%	0.82	91.5%	90.4%	0.91

There are 452 sub-windows in Data Set 1. The precision of sub-window detection in TMRA-SVM is 97.1%, while the precision in TMRA is merely 11.9%.

Table 2 shows the measure of our system. Data Set 2 was the testing set. The SVM was trained with CCTV9 news videos.

Table 2. The measure of testing CNN news

Transition Type	Precision	Recall	F1
CUTs	80.3%	96.2%	0.88
GTs	88.7%	57.5%	0.70
Average	81.9%	88.4%	0.85

The precision and recall are much lower, because two kinds of GTs, which are not found in CCTV9 news, appeared in the CNN news frequently.

4.2 Experiment 2

SWVs of 10 hours CNN News videos were generated. The SVM was trained with these SWVs.

We used the Data Set 2 to test the second SVM. Table 3 shows that the result is much better than the result in experiment 1 when tested the same data set. It's obvious that the SVM in our framework should be retrained while detecting video clips in different TV channels. This is the basic knowledge for learning based approach.

Table 3. The measure of testing CNN news

Transition Type	Precision	Recall	F1
CUTs	95.6%	98.0%	0.97
GTs	88.4%	88.4%	0.88
Average	93.7%	95.4%	0.945

5 Conclusions and Future Work

In this paper, an improved TMRA algorithm was developed for SBD. SBD can be regarded as a problem of pattern recognition. By incorporating of SVM classifier, the system classifies SWVs into NF, CF and GF. It's a general framework for all kinds of video transition. Results of experiments show that our method has good performance for SBD. It basically resolves the difficulties of detection caused by sub-window. The framework also greatly improves accuracy of gradual transitions of shot.

Acknowledgments. This research work is supported in part by National Natural Science Foundation of China under grant No.60472082 and also supported by Key Lab Research Project of Beijing Electronic and Science Technology Institution (No. YZDJ0430).

References

1. Naiyang Deng, Yingjie Tian: A New Method in Data Mining-Support Vector Machine. Beijing: Science Publisher (2004)
2. HuaMin Feng, A Chandrasekhara, T.S. Chua: An Automatic Temporal Multi-resolution Analysis Framework for Shot Boundary Detection. MMM 2003, the 9th International Conference on Multi-Media Modeling, (2003), 224
3. YiHong Gong: An Accuracy and Robust Method for Detecting Video Shot boundaries, ICMCS, (1999), Vol. 1 850-854
4. F Idris, S. Panchanathan: Review of Image and Video Indexing Techniques. Journal of Visual Communication and Image Representation, (1997 June), 8(2), 146-156.
5. Lin, H.-T. and C.-J. Lin: A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods. Technical report, Department of Computer Science and Information Engineering, National Taiwan University. (2003)
6. S. Knerr, L. Personnaz, and G. Dreyfus: Single-layer learning revisited: a stepwise procedure for building and training a neural network. In J. Fogelman, editor, Neurocomputing: Algorithms, Architectures and Applications. Springer-Verlag, (1990)
7. S. S. Keerthi and C.-J. Lin: Asymptotic behaviors of support vector machines with Gaussian kernel. Neural Computation (2003)15 (7), 1567-1589.
8. Yu-Ping Wang, S.L. Lee: Scale-Space Derived From B-Spline, PAMI 1998, Vol.20, No.10.
9. HongJiang Zhang, A. Kankanhalli, S. W. Smoliar: Automatic partitioning of full-motion video. ACM Multimedia Systems, (1993), 1 (1), 10-28.

Robust Pattern Recognition Scheme for Devanagari Script

Amit Dhurandhar¹, Kartik Shankarnarayanan², and Rakesh Jawale³

¹ University of Florida, Gainesville, Florida, USA
amit.dhurandhar@gmail.com

² Persistent Systems Pvt. Ltd, Pune, India
kartik9@gmail.com

³ Persistent Systems Pvt. Ltd, Pune, India
rakesh.jawale@gmail.com

Abstract. In this paper, a Devanagari script recognition scheme based on a novel algorithm is proposed. Devanagari script poses new challenges in the field of pattern recognition primarily due to the highly cursive nature of the script seen across its diverse character set. In the proposed algorithm, the character is initially subjected to a simple noise removal filter. Based on a reference co-ordinate system, the significant contours of the character are extracted and characterized as a contour set. The recognition of the character involves comparing these contour sets with those in the enrolled database. The matching of these contour sets is achieved by characterizing each contour based on its length, its relative position in the reference co-ordinate system and an interpolation scheme which eliminates displacement errors. In the Devanagari script, similar contour sets are observed among few characters, hence this method helps to filter out disparate characters and narrow down the possibilities to a limited set. The next step involves focusing on the subtle yet vital differences between the similar contours in this limited set. This is done by a prioritization scheme which concentrates only on those portions of character which reflect its uniqueness. The major challenge in developing the proposed scheme lay in striking the right balance between definiteness and flexibility to derive optimal solutions for out of sample data. Experimental results show the validity and efficiency of the developed scheme for recognition of characters of this script.

Keywords: Devanagari script recognition, feature extraction, interpolation, contour sets, horizontal scanning, vertical scanning, priority matching.

1 Introduction

Devanagari script is the base script of many languages in India such as Sanskrit, Hindi and Marathi besides other Indian languages that use variants of this script. Since the words in Devanagari are two dimensional compositions of characters and symbols, it becomes fundamentally different from Roman and ideographic scripts. Consequently, simple adaptation of the algorithms which perform well for other scripts are rendered ineffective in case of Devanagari. R. Sinha et al [1],[2],[3],[4] have reported various aspects of Devanagari script recognition where the post-processing system is based on

contextual knowledge which checks the composition syntax. Sethi et al [5] have described Devanagari recognition based on the structural approach using primitives like horizontal and vertical line segment. A decision tree is employed to perform the analysis based on the presence/absence of these primitives and their interconnection. Neural network approach for isolated characters has also been reported. However, the accuracy rates reported by these methods are still relatively low. Also none of these methods have shown good recognition rates in case of out of sample data. The rest of the paper describes a new approach towards this problem of Devanagari script recognition. The basic strength of this proposed approach is its simplicity since it exploits the inherent cursive nature of the script unlike previous approaches which relied on applying complex mathematical techniques towards the problem.

2 Feature Extraction

The aim of this phase is to extract the significant curves from the given character. The input to this phase is a preprocessed character bitmap obtained using Median Filtering [6]. This bitmap is considered to be placed on a co-ordinate system, where each pixel has integer co-ordinates. The X-axis proceeds along the width and the Y-axis along the height of the image. The character is scanned from left top to right bottom first, and then from right bottom to left top (Horizontal scanning). The curves extracted from these two scans constitute the contour set for this character (refer to Fig.2).

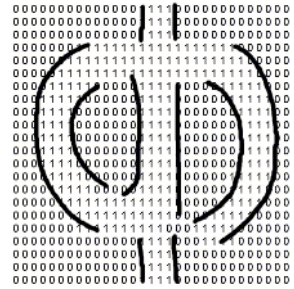


Fig. 2. Extraction Of Curves

Further information about the character curves could be extracted by similarly scanning from top to bottom and vice-versa (Vertical scanning). However, in Devanagari, most characters share strikingly similar attributes on Vertical scanning, such as the *Shirorekha*. Thus Vertical scanning is skipped since the complexity added by it far outweighs the information it helps to retrieve. So, we concentrate only on the curves extracted from horizontal scanning. These curves are stored in terms of their normalized centroids, normalized lengths (in terms of the number of pixels spanned by it) and normalized co-ordinates of the points that form them. This normalization is performed apropos the character width (for Y co-ordinates) and height (for length and X co-ordinates). To

$$X_c = \frac{\sum_{k=1}^n x_k}{n} \quad Y_c = \frac{\sum_{k=1}^n y_k}{n} \quad (1)$$

calculate the normalized centroids, the formulae used are shown in (1) where (X_c, Y_c) denote the co-ordinates of the centroid, (x_k, y_k) denote the co-ordinates of the points that form the curve and n denotes the length of the curve.

3 Pattern Matching

Once the features of the input character have been extracted, they are compared with the features of the template characters, which are stored in a database. For each curve of a template character, the database contains the centroid of the curve, its length and

points of the curve required for interpolation. Each template character in the database is matched with the input character, curve by curve, as follows:

3.1 Centroid Matching

In centroid matching, for a curve in the input character a corresponding curve in the template character is determined such that the difference in distances between the centroids of the two curves is lesser than,

1. the difference in distances between the centroids of the input character curve and other curves in the template character.
2. the difference in the distances between the centroids of the template character curve and other curves in the input character.

If this difference lies within a given threshold, then this centroid difference value is saved as $\Delta_{centroid}$ and then *Length Matching* of the two centroid matched curves is done. If not, the two curves are marked as not matched.

3.2 Length Matching

Length Matching involves determining whether the centroid-matched curves are comparable in length. The centroid-matched curves are said to be matched in length only if the difference in their lengths lies within a threshold. If so, then the points of these two curves are matched using *Interpolation*. The threshold used here is newly calculated for each unique pair of input and template character.

3.3 Interpolation

The aim of interpolation is to superimpose the two curves to be compared and then observe the extent to which they cleanly fit over each other. From the two curves that satisfy the length matching criterion, the curve having shorter length is determined and is kept as the reference curve. Now, every point of this reference curve is shifted by a distance of $\Delta_{centroid}$ to eliminate displacement errors. After this, the points of the other curve are interpolated with respect to the reference curve. For every curve of the input character, the extent to which this curve has matched with the corresponding curve of the template character is determined. Based on this, a priority is assigned to each template character that satisfies the length and centroid matching criteria (Fig.3).

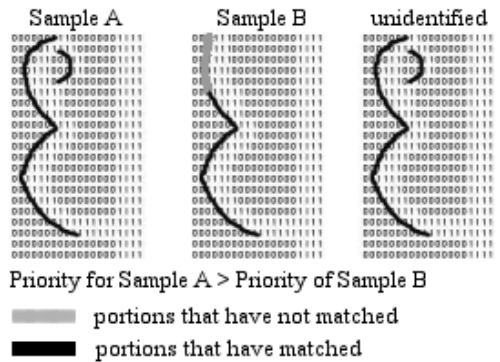


Fig. 3. Priority Matching

This priority determines the probability of the input character being the template character. Once the comparison with all the template characters is done, the template

character having the highest priority is identified as the input character. This step helps in improving the accuracy of character recognition by correct identification of the input character from the available set of characters that have already satisfied the length matching and centroid matching criteria

Once all curves of significant length have been matched then the character is said to have matched in this scan. If not, then the unmatched curves are taken from both the input and the database characters and checked

for joining. The curve joining operation (Fig.4.) is required in order to make the technique more robust as it was found while testing that for large variation in fonts, the curves extracted were not identical to those of the template characters.

While performing the curve joining operation, two unmatched curves of a character (from input or database) that are close to each other are considered and the distance between their end points is calculated. This distance is compared with a pre-determined threshold value for decision making. If the distance lies within this threshold, then the two curves are joined. The resulting curve is matched with an unmatched curve of the corresponding database character or input character. Thus, this threshold value, which is calculated in accordance with the training data, acts as a recognition parameter. Following this, a comparison of various curves is carried out as previously mentioned. If the significant curves now match, then the character is said to have matched.

4 Experimental Results

The proposed approach was experimentally evaluated for a range of Devanagari fonts. Throughout the experimentation stage, the characters of the Font *Osho* were chosen for training and the recognition of all fonts was done using these only. The first test carried out was primarily aimed at establishing the effectiveness of the proposed approach. The test data used for this test involved three largely varying fonts viz. *Osho*, *Surekh* and *Yogesh*. Since 60 was the bare minimum number of samples needed to cover the character set, the test started with a training set of 60 samples. After this, the size of training set was continuously increased and accuracy was measured at every stage. The test concluded with the training set having 91 samples (Fig.5.). After this, the test also covered an extra phase in which the trained set count was increased from 91 to 94 and then 97. The accuracy obtained with this remained stable showing that the recognition system was robust when trained with a basic character set of 91 and that training duplicate samples neither improved nor degraded the accuracy.

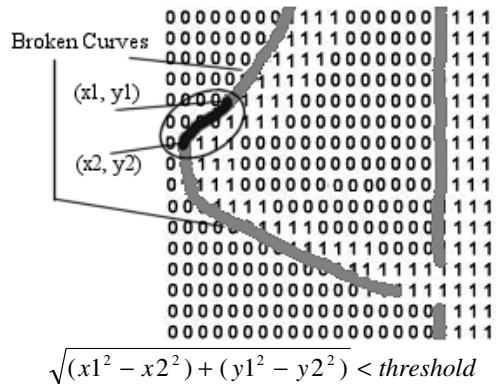


Fig. 4. Curve Joining operation

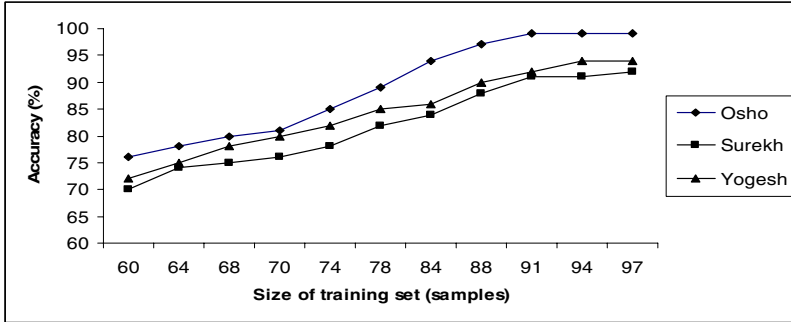


Fig. 5. Accuracy vs Size of training set

The recognition rate in terms of characters per second is directly proportional to the size of trained data set in terms of number of samples (time complexity $O(n)$, where n stands for number of samples trained). The next test was aimed at establishing this proposition. It was observed that the recognition rate increased linearly with the size

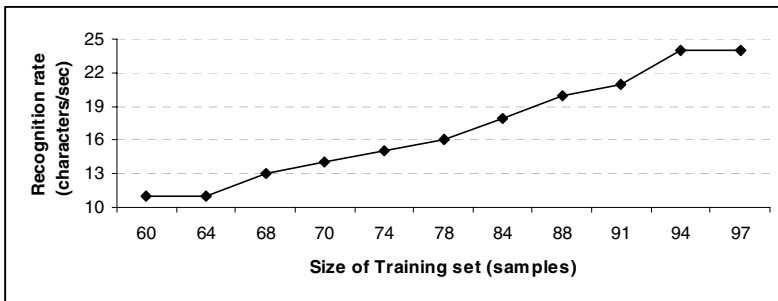


Fig. 6. Recognition Rate vs Size of Training set

of trained set. The system showed a recognition rate of 11 characters per second for a training set of 60 samples and 21 characters per sec for 91 samples. Thus a robust database showing a good balance of speed and accuracy was achieved with a trained set of 91 samples. Using this trained data set of 91 samples, the test was extended to detect characters from many unknown fonts obtained from various sources (Table 1).

Table 1. Results using a trained data set of 91 samples

(No of Chars/ Font)	Accuracy (%)	(No of Chars/ Font)	Accuracy (%)
8415 / Osho	99.07	2047 / Surekh	92.00
1947 / Yogesh	94.00	1278 / unknown	91.15
1478 / unknown	91.88	1021 / unknown	90.10

After this, the system underwent an informal field test with inputs from different real-life sources such as newspapers, old books etc. The results obtained were encouraging. Accuracy rates in most cases were upward of 85%. It should be mentioned that by varying few of the recognition parameters, the accuracy did not vary significantly, thus reflecting the robustness of the proposed approach.

5 Conclusion

We have proposed a novel approach for Devanagari script recognition. The proposed approach has two major contributions in this direction. The first is to represent each character as a contour set and use this set for matching based on its length, its relative position in the reference co-ordinate system, and an interpolation scheme. The next involves focusing on the minutiae among similar contours in this limited set. This is done by a prioritization scheme which concentrates only on those portions of character which reflect its uniqueness. Experimental results show usefulness of the proposed scheme. They constitute a significant improvement over contemporary approaches which aim to achieve the same goal.

References

1. S.Marwah, S.K.Mulick and R.M.K.Sinha, *Recognition of Devanagari characters using a hierarchical binary decision tree classifier*. IEEE International Conference on Systems, Man and Cybernetics, (October 1994).
2. R.M.K.Sinha and H.N.Mahabala, *Machine recognition of Devanagari script*, *IEEE Transactions on Systems, Man and Cybernetics*, vol.SMC-9, (1979) 435-441.
3. R.M.K.Sinha, *Rule based contextual post-processing for Devanagari text recognition*, *Pattern Recognition*, vol.20, no.5, (1987) 475-485.
4. R.M.K.Sinha, *Visual text recognition through contextual processing*, *Pattern Recognition*, vol.21, no.5, (1998) 463-479.
5. I.K.Sethi and B.Chaterjee, *Machine recognition of handprinted Devanagari Numerals*, *Journal of Institution of Electronics and Telecommunication Engineers, India*, vol.2, (1976) 532-553.
6. Anil K. Jain, *Fundamentals of Digital Image Processing*. Prentice Hall, (1989).

Credit Evaluation Model and Applications Based on Probabilistic Neural Network

Sulin Pang

Department of Mathematics, Jinan University, Guangzhou 510632, China
pangsulin@163.com

Abstract. The paper introduces the method of probabilistic neural network (PNN) and its classifying principle. It constructs two PNN structures which are used to recognize both the two patterns and the three patterns respectively. The structure of the two patterns classification of PNN is used to classify the 106 listed companies of China in 2000 into two groups. The classification accuracy rate is 87.74%. The structure of the three patterns classification of PNN is used to classify the 96 listed companies of China in 2000 into three groups. The classification accuracy rate is 85.42%.

1 Introduction

Probabilistic Neural Network (PNN) is a artificial neural network model by Specht^[1] proposed in 1990. It has not only the function of both statistics and classification but also it is not submitted to the multivariate normal condition. Tyree et al^[2](1995) constructed a PNN which has 4 layers: the input layer, the pattern unit layer, the collection unit layer and the output layer. Yang and Marjorie^[3](1999) utilized the PNN technology established the bankruptcy predicting model for companies, and then used the model to study the predicting problem of the financial crisis to the 122 companies of the oil industry in USA. Hajmeer^[4](2002) and Simon^[5](2001) also used PNN to study the problems of pattern recognition. Yiuming Cheung and Rongbo Huang^[6] (2005) proposed a divide-and-conquer RBF, which has an outstanding performance compared to the existing one. Sulin Pang^[7,8] (2005) used radial function network and the method of fuzzy cluster analysis to study the problem of the credit evaluation for listed companies of China respectively.

The present paper establishes a credit evaluation model of listed companies based on probabilistic neural network. The model is applied to classify both two patterns on 106 Chinese listed companies and three patterns on 96 Chinese listed companies respectively.

2 Selection of the Samples

To select the samples of the two patterns, 106 listed companies of China in 2000 are adopted, in which, there are 53 “bad” companies consist of all the ST &

PT companies published in 2000, matched with the other 53 “good” companies which are not in loss.

Two classes of mistakes are defined here: The 1st class of mistake means that mistake a ”bad” company for a “good” company. The 2nd class of mistake means that mistake a “good” company for a “bad” company.

To select the samples of the three patterns, 96 listed companies in Chinese Stock Exchange in 2000 are adopted as data samples. The 96 data are separated into three groups: a “good” group, a “middle” group and a “bad” group, in which, there are 32 “bad” companies consist of the ST & PT companies published in 2000, 32 “good” companies, and 32 “middle” companies whose finance are between the “good” group and the “bad” group.

Six classes of mistakes are defined here: The 1st class of mistake means that mistake a “bad” company for a “median” company. The 2nd class of mistake means that mistake a “bad” company for a “good” company. The 3rd class of mistake means that mistake a “median” company for a “bad” company. The 4th class of mistake means that mistake a “median” company for a “good” company. The 5th class of mistake means that mistake a “good” company for a “bad” company. The 6th class of mistake means that mistake a “good” company for a “median” company.

Whether the two patterns or the three patterns, to each listed company, the following four major financial indexes are still considered: x_1 is earning per share, x_2 is net asset per share, x_3 is return on equity, x_4 is cash flow per share.

3 Probabilistic Neural Network(PNN)

Assume that the populations made up of n groups (or classes) are G_1, G_2, \dots, G_n respectively. The prior probability of the i th group (or class) is $P(G_i)$ ($i = 1, 2, \dots, n$). To the input vector x , the conditional probability of each class is $P(x|G_i)$. According to Bayes theorem, the posterior probability $P(G_i|x)$ can be expressed by.

$$P(G_i|x) = \frac{P(x|G_i)P(G_i)}{P(x)} = \frac{P(x|G_i)P(G_i)}{\sum_{k=1}^n P(x|G_k)P(G_k)} \tag{1}$$

The discriminant rule is:

$$\begin{cases} x \in G_i & P(G_i|x) > P(G_j|x), i \neq j, i, j = 1, 2, \dots, n \\ \text{unkown} & \text{otherwise} \end{cases} \tag{2}$$

If the prior probability $P(G_i)$ is known, the Bayes discriminant rule becomes into:

$$\begin{cases} x \in G_i & P(x|G_i)P(G_i) > P(x|G_j)P(G_j), i \neq j, i, j = 1, 2, \dots, n \\ \text{unkown} & \text{otherwise} \end{cases} \tag{3}$$

If the data samples of each class is the same, that is, $P(G_i) = \frac{1}{n}$, the posterior probability of becomes into:

$$P(G_i|x) = \frac{P(x|G_i)P(G_i)}{P(x)} = \frac{P(x|G_i)P(G_i)}{\sum_{k=1}^n P(x|G_k)P(G_k)} = \frac{P(x|G_i)}{\sum_{k=1}^n P(x|G_k)} \tag{4}$$

So we can get simpler discriminant rule as follows:

$$\begin{cases} x \in G_i & P(x|G_i) > P(x|G_j) \\ \text{unkown} & \text{otherwise} \end{cases} \tag{5}$$

In which, $P(x|G_i)$ is used the following estimating equation ^[12]

$$\hat{P}(x|G_i) = \frac{1}{(2\pi)^{\frac{n}{2}} |\sum_i|^{-\frac{1}{2}}} \exp\left\{\frac{1}{2}(x - \mu_i)^T \sum_i^{-1} (x - \mu_i)\right\} \tag{6}$$

where μ_i is a mean vector, \sum_i is a covariance matrix of G_i , \sum_i^{-1} is an inverse matrix of the covariance matrix of $G_i (i = 1, 2 \dots, n)$.

To arbitrary two populations G_i, G_j whose have the same number of the samples, the working principle of PNN is illustrated in Fig. 1^[3], in which, the solid line is the conditional probability density function $P(x|G_i)$ of G_i and the dash line is the conditional probability density function $P(x|G_j)$ of G_j . To a testing sample x which is in the range of δ , because $P(x|G_i) > P(x|G_j)$, we can decision $x \in G_i (i, j = 1, 2, \dots, n, \text{ but } i \neq j)$ according to discriminant rule (5).

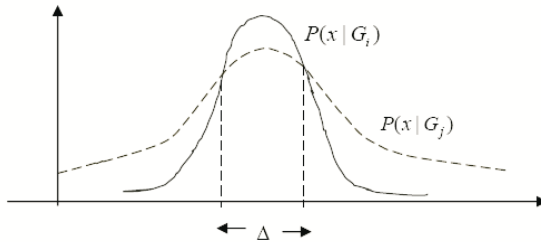


Fig. 1. The Working Principle of PNN

4 Two Patterns Classification

In this section, we will construct a PNN with 4 layers: one input layer, two hidden layers and one output layer, to solve the problem of the two patterns classification. The PNN network structure is illustrated in Fig. 2. The first layer is an input layer which accepts an input vector $x = (x_1, x_2, x_3, x_4)$ and includes 4 nodes. The first hidden layer consists of the 63 training samples. They makes up of the two patterns which are both G_1 and G_2 , in which, $P(G_1) = 32$, $P(G_2) = 31$. The state function of the layer's units is the conditional probability density function $\hat{P}(x|G_i) (i = 1, 2)$ which are evaluated equation (6). The second hidden layer is a repeated plus layer and includes 2 nodes. Each node representatives a class. It is evaluated various classes of the posterior probability $\hat{P}(G_i|x)$. The fourth is the output layer, it will give the decision result according to the discriminant rule (3). In Fig.2, a small rectangle on the left of

each pattern in the first hidden layer representatives the conditional probability $\hat{P}(x|G_i)$. The rectangle is higher, the corresponding conditional probability is larger. A small rectangle on the left of each class $G_i(i = 1, 2)$ in the second hidden layer representatives the posterior probability $\hat{P}(G_i|x)$. The rectangle is higher, the corresponding posterior probability is larger. It is easy to know that, if $\hat{P}(x|G_1) > \hat{P}(x|G_2)$, because $P(G_1) > P(G_2)$, we have $\hat{P}(G_1|x) > \hat{P}(G_2|x)$. The simulating results discriminated both the training set and the testing set

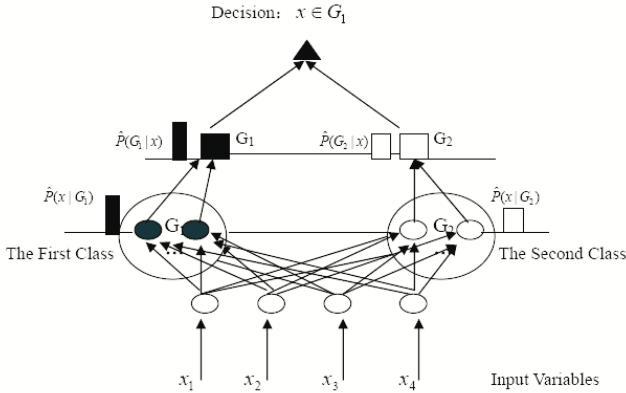


Fig. 2. PNN Structure of the Two Patterns

of the two patterns data by using PNN is shown in table 1. Table 1 shows that, to the training set, the misclassification number of both the 1st class and the 2nd class are all 0. So the classification accuracy rate to the training set of PNN is 100%. To the testing set, the misclassification number of the 1st class is 13, and the 2nd class is 0. So the overall misclassification rate is 30.23%. And the classification accuracy rate to the testing set of PNN is 69.77%.

Summarily, to the total 106 data, when PNN is used to classify the two patterns data, the total misclassification number is 13, the overall misclassification rate is 12.26%, so the classification accuracy rate is 87.74%(See table 1).

Table 1. Classification Accuracy Rate of the Two Patterns

PNN Method	1 st Class of Error	2 nd Class of Error	Overall	Accuracy Rate
Training Set (63)	0 (0.00%)	0 (0.00%)	0 (0.00%)	100%
Testing Set (43)	13 (30.23%)	0 (0.00%)	13 (30.23%)	69.77%
Population(106)			13 (12.26%)	87.74%

5 Three Patterns Classification

This section, we will construct a PNN with 3 layers: one input layer, one hidden layer and one output layer, to carry out the three patterns classification. The

PNN network structure is illustrated in Fig. 3. The first layer includes 4 nodes. The hidden layer includes 60 nodes, which are separated into three classes: G_1 , G_2 and G_3 . The state function of the layer's units is conditional probability density function $\hat{P}(x|G_i)(i = 1, 2, 3)$, which are evaluated equation (6). The output layer gives the decision result according to the discriminant rule (5). In Fig. 3, it

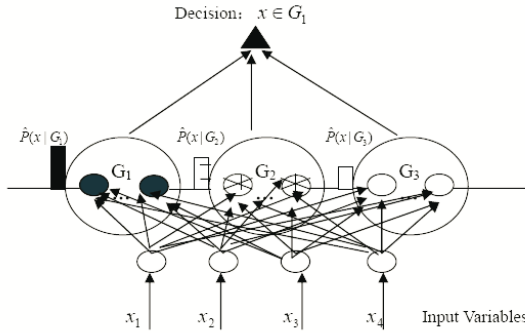


Fig. 3. PNN Structure of the Three Patterns

is assumed that $\hat{P}(x|G_1) > \max(\hat{P}(x|G_2), \hat{P}(x|G_3))$, the decision result will be $x \in G_1$ from the discriminant rule (5).

The simulating results discriminanted both the training set and the testing set of the three patterns by using PNN is shown in table 2. Table 2 shows that, to the training set, the misclassification number of the 6 classes are all 0. So the classification accuracy rate to the training set is 100%. To the testing set, the misclassification number of the 1st class is 12, the 6 th class is 2 and the others are all 0. So the total misclassification number are 4, and the overall misclassification rate is 38.89%. The classification accuracy rate to the testing set is 61.11%.

Summarily, to the total 96 samples, when PNN is used to classify the three patterns, the total misclassification number are 14, the overall misclassification rate is 14.58%, so the classification accuracy rate is 85.42%(See table 2).

Table 2. Classification Correct Rate of the Three Patterns

PNN Method	1 st Class of Error	2 nd Class of Error	3 rd Class of Error	4 th Class of Error	5 th Class of Error	6 th Class of Error	Overall	Accuracy Rate
Training Set	0(0.00%)	0(0.00%)	0(0.00%)	0(0.00%)	0(0.00%)	0(0.00%)	0(0.00%)	100%
Testing Set	12(3.33%)	0(0.00%)	0(0.00%)	0(0.00%)	0(0.00%)	2(5.56%)	14(38.89%)	61.11%
Population(96)							14(4.58%)	85.42%

6 Conclusions

The research shows that, whether the two patterns classification or the three patterns classification, to the training samples, the classification accuracy rates

of PNN are all very high, they are all 100PNN are all very low. What the cases are appeared is because PNN adopts a classifying method of the self-supervised forward network. The method does not need to train the connected weight of the network. Its hidden units are composed of all the training samples directly. Therefore, it has very strong recognition capability to the training samples. But its testing performance is relatively bad and its recognition capability is very weak to the testing samples.

Acknowledgments

The research is supported by the Natural Science Foundation of Guangdong Province (31906), the Key Programs of Science and Technology Bureau of Guangzhou (2004Z3-D0231) and the Key Programs of Science and Technology Department of Guangdong Province (2004B10101033).

References

1. Specht D. F. Probabilistic neural network. *Neural Network*, 1990, 3(2): 109-118.
2. Tyree Eric W, Long J. A. Assessing financial distress with probabilistic neural network. *Proceedings of the Third International Conference on Neural Networks in the Capital Market*, 1995.
3. Z. R. Yang, Marjorie B. Platt and Harlan D. Platt. Probabilistic neural networks in bankruptcy prediction. *Journal of Business Research*, 1999, 44: 67-74.
4. M. Hajmeer, I. Basheer. A probabilistic neural approach for modeling and classification of bacterial growth/no-growth data. *Journal of Microbiological Methods*. 2002, 51: 217-226
5. Laurent Simon, M. Nazmul Karim. Probabilistic neural network using Bayesian decision strategies and a modified gompertz model for growth phase classification in the batch culture of bacillus subtilis. *Biochemical Engineering Journal*. 2001, 7: 41-48
6. Yiuming Cheung and Rongbo Huang. A divide-and-conquer learning approach to radial basis function networks, *Neural Processing Letters*, 2005, 21:189-206
7. Sulin Pang. Credit scoring model based on radial function network. *Proceedings of 2005 IEEE International Workshop on VLSI Design and Video Technology*, 2005, 389-392.
8. Sulin Pang, Bijnu Hu and Yuanhuai Bai. Application of Fuzzy Cluster Analysis in Credit Scoring. *Journal of Systems Science and Information*, 2005, 3 (1): 157-168
9. Hugo Guterman, Youval Nehmadi, Andrei Chisyakov, Jean F. Soustiel, Moshe Feinsod. A comparison of neural network and Bayes recognition approaches in the evaluation of the brainstem trigeminal evoked potentials in multiple sclerosis. *International Journal of Bio-Medical Computing*, 1996, 43: 203-213.

Fingerprint Ridge Line Extraction Based on Tracing and Directional Feedback

Rui Ma¹, Yaxuan Qi², Changshui Zhang³, and Jiaxin Wang¹

¹ State Key Laboratory of Intelligent Technology and Systems,
Department of Computer Science, Tsinghua University, Beijing 100084, China
mr02@mails.tsinghua.edu.cn

² Network Security Lab, Research Institute of Information Technologies,
Tsinghua University, Beijing 100084, China

³ State Key Laboratory of Intelligent Technology and Systems,
Department of Automation, Tsinghua University, Beijing 100084, China

Abstract. Fingerprint recognition and verification are always the key issues in intelligent technology and information security. Extraction of fingerprint ridge lines is a critical pre-processing step in fingerprint identification applications. Although existing algorithms for fingerprint extraction work well on good-quality images. Their performance decrease when handling poor-quality images. This paper addresses the ridge line extraction problem as curve tracking processes under the framework of probabilistic tracking. Each ridge line is modeled as sequential frames of a continuous curve and then traced by standard CONDENSATION algorithm in the area of computer vision. Additionally, local directional image is rectified with a feedback technique after each tracking step to improve the accuracy. The experimental results are compared with those obtained through existing well-known algorithms, such as local-binarization and sampling-tracing methods. In spite of greater computational complexity, the method proposed performs better both in terms of efficiency and robustness.

1 Introduction

Fingerprint recognition and verification are always the key issues in intelligent technology and information security. Extraction of fingerprint ridge lines is a critical pre-processing step in the identification of minutia in fingerprint images. Because of the practically poor quality of fingerprint images, it is a hard task to extract coherent and smooth ridge lines.

Thus, the problem of extracting fingerprint ridge lines in poor quality images is challenging. Conventional methods follow the binarization and thinning steps, which are only adequate providing the fingerprints are clear and continuous. While in most practical situation, the acquired images often suffer from noise, deformation and contrast deficiency. Therefore, it is considerably difficult to achieve reliable fingerprint ridge lines from poor quality images. A ridge line following algorithm had been proposed in [1]. However the accuracy depends on the directional image and the traced points may diverge from the ridges when the local information is absent. One effective approach is to use random sampling. The research in [4] introduced a sampling based

method to effectively trace the ridge lines. In this paper, we extended this work by extracting ridges under the framework of Probabilistic Tracking. Main contributions of this paper include:

- An Extended Sampling Algorithm: Dynamic tracing concept is employed to the ridge line extraction in static images. Based on the dynamic sequential frames, fingerprint ridge lines are traced by a sampling method, which utilizes the pixels gray levels and direction information. However, ridge line tracing in this paper adopting the standard CONDENSATION algorithm, which is more efficient and robust than the simple “prediction-expectation” algorithm used in [4].
- More Accurate Probabilistic Model: In order to further improve the accuracy of ridge line extraction results, the criterion for evaluating the weight of each sample is revised by including the gray level distribution along the orthogonal direction through each sample. In spite of greater computational complexity, the novel ridge line model guarantees the ideal performance of the tracking algorithm.
- More Accurate Directional Image: Due to the effective sampling technique and direction feedback, more accurate directional images are obtained.

2 Probabilistic Tracking

The effectiveness of probabilistic tracking of objects in computer vision has been revolutionized by the development of particle filtering. The technique of Particle Filters has gained a lot of popularity in recent years to solve visual tracking problems. Their popularity stems from their simplicity, flexibility, ease of implementation, and modeling success over a wide range of challenging application [3]. The pioneering work by Isard and Blake was termed CONDENSATION[2]. In the framework of target tracking, the target state distribution is iteratively estimated over time by random samples and the learned dynamical models.

The CONDENSATION algorithm actually retains a set of weighted samples, denoted as $s_t^{(i)}, i = 1 \dots N$ with weights $\pi_t^{(i)}$, representing approximately the conditional state-density $p(X_t | Y_t)$ at time t . X_t is the state and Y_t is the observation at time t . In the fingerprint ridge line tracing problem, the moving target in visual tracking is taken as a segment on the ridge line. The segment is represented by its center point at $P_t = (i_t, j_t)$ which is the coordinates of the center point in the fingerprint image. Target state X_t is defined as the ridge line direction D_t at point P_t and the observation Y_t is defined as the gray level distribution Z_t in a certain area relevant to point P_t . The key problem is to estimate $p(D_{t+1} | Z_{t+1})$ along the traced ridges as if time evolves. CONDENSATION solves this following a sampling-diffusion way, and outputs a weighted sample-set $\{s_{t+1}^{(i)}, \pi_{t+1}^{(i)}, i = 1 \dots N\}$, representing $p(D_{t+1} | Z_{t+1})$. Once the sequence of sample distribution is known, by Monte Carlo integration, the estimated point along the ridge line at time $t+1$ is:

$$E(D_{t+1} | Z_{t+1}) = \int D_{t+1} p(D_{t+1} | Z_{t+1}) dD_{t+1} = \sum_{i=1}^N \pi_{t+1}^{(i)} s_{t+1}^{(i)} \tag{1}$$

Following this framework, to trace the ridge lines, our algorithm locates a point representing the local ridge line segment in a section along the ridge line direction in each step. By connecting all the traced points, a polyline approximation of the ridge line can be obtained.

3 Ridge Line Model

The gray scale fingerprint images can be seen as the combination of surface ridges and ravines. Denote $G(i, j)$ as the gray level at pixel (i, j) of the $a * b$ image I with 0 to $g - 1$ gray levels. Let $z = G(i, j)$ be the discrete surface corresponding to the image I . Then the ridges are the connected dark pixels with the gray levels near $g - 1$ and the ravines are the connected bright pixels with the gray levels near zero.

Fingerprint ridge lines have some specific characteristics that would facilitate tracking. First one is *uniformity*, which means that the ridges are distributed with even density throughout the whole image. The second is *continuity*, which denotes that the ridge lines go smoothly and coherently and the directions of a series of ridge line segments do not change abruptly. The third one is *correlation*, which indicates that the neighboring ridge lines often distribute by nearly the same direction. All these characteristics make it practical and feasible to track the ridge lines.

4 Tracking Framework

To get start the tracking, we use the method similar to [1] to obtain the initial rough directional image $imgD$. Thus the fingerprint ridge line can be traced one by one as the starting points on the edges of an image are given.

According to the CONDENSATION algorithm, the tracking process output a weighted, time-stamped sample-set, $\{P_t^{(i)}, D_t^{(i)}, \omega_t^{(i)}\}, i = 1, \dots, N$, representing approximately the conditional state-density $p(D_t | Z_{1:t})$. Direction samples $D_t^{(i)}$ are formed by samples

point $P_t^{(i)}$ and $P_{t-1}^{(i)}$. Then the expected ridge point at time t is $E(P_t) = \sum_{i=1}^N \omega_t^{(i)} P_t^{(i)}$.

4.1 Sampling

From a selected $P_t^{(i)}$ (according to $\omega_t^{(i)}$), a set of expectation points $\{y_k\}$ are obtained by detecting all the maximum and saddle points in area $A_t^{(i)}$. Here $A_t^{(i)}$ is defined as a quarter-ring oriented to the direction of $D_t^{(i)}$ with radius r_1 and r_2 . For each point y_k in $A_t^{(i)}$, the absolute value of acute angle $|\theta_k|$ formed by $D_t^{(i)}$ and $\overline{P_t^{(i)}y_k}$ is computed. Assume that $\{y_k\}$ has Gaussian distribution over $|\theta_k|$, that is

$p(y_k) = (\frac{1}{2\pi\sigma}) \exp(-\frac{|\theta_k|^2}{2\sigma^2})$, then the sample $P_{t+1}^{(i)}$ is: $P_t^{(i)} = (\sum_k p(y_k) \cdot y_k) / (\sum_k p(y_k))$.

4.2 Weighting

Section $B_t^{(i)}$ is defined as the rectangle determined by $P_t^{(i)}$ and $P_{t+1}^{(i)}$. Let $D_{t+1}^{(i)}$ be the direction defined by the starting point $P_t^{(i)}$ and sample point $P_{t+1}^{(i)}$. Section $C_t^{(i)}$ is defined as a rectangle orthogonal to $D_{t+1}^{(i)}$ through $P_{t+1}^{(i)}$, with length a and width b . The weight $\omega_{t+1}^{(i)}$ is determined by the following values:

- $g - 1$ The maximal pixel intensity level of the whole image.
- $G^{(i)}$ The average gray level within $B_t^{(i)}$.
- $V_1^{(i)}$ The variance of the gray level within $B_t^{(i)}$.
- $V_2^{(i)}$ The variance of the gray level within $C_t^{(i)}$.
- $I^{(i)}$ The included acute angle between $D_{t+1}^{(i)}$ and $D_t^{(i)}$.

Thus the weight is computed by

$$\omega_{t+1}^{(i)} = \exp\left(-\frac{K(g - G^{(i)})V_1^{(i)} | I^{(i)} |}{V_2^{(i)}}\right) \tag{2}$$

The weight $\omega_{t+1}^{(i)}$ which is confined within $[0,1]$ is inferred from local pixels intensity distribution through the exponential function. Parameters K , a and b are determined empirically in experiments. Thus the iterative process outputs a weighted sample set $\{P_{t+1}^{(i)}, D_{t+1}^{(i)}, \omega_{t+1}^{(i)}\}, i = 1, \dots, N$. By computing $E(P_{t+1}) = \sum_{i=1}^N \omega_{t+1}^{(i)} P_{t+1}^{(i)}$, the expected ridge point at time $t + 1$ is figured out.

4.3 Direction Feedback

Since the directions in area $A_t^{(i)}$ around point $P_t^{(i)}$ are similar to a certain extent, according to ridge line characteristics of *continuity* and *correlation*, a feedback technique is adopted to correct the directions in area $A_t^{(i)}$ by convoluting the newly obtained included angle $I^{(i)}$ with a 2D Gaussian mask and applying the result to update the rough initial directional image. This approach would in turn promote the precision of the following tracking steps.

4.4 Starting and Ending

These starting points can be easily obtained according to the distribution of edge pixels intensities, and the false points can be eliminated during tracing process. During the tracing process, the algorithm automatically stops the tracing of one single ridge line when encounters with one of following situations:

- Exterior $E(P_{t+1})$ is external to the window of interest.
- Termination Most of the searching area $A_t^{(i)}$ is lack of local maxima as compared with the number of local minima. Then the algorithm stops

independently on the gray level of the current region, and it can work on both saturated regions and contrast-deficient regions without particular enhancement.

Intersection $E(P_{i+1})$ has been previously labeled as a point belonging to another ridge line.

5 Experiments and Performance Evaluation

To objectively assess the performance of the proposed algorithm, the proposed algorithm is compared to existing methods, including local-binarization [7] and tracking [4] algorithms.

Fig.1 illustrates the ridge line extraction results on good-quality images. We can see from the figure that both tracking (Fig.1 (c)) and CONDENSATION (Fig.1 (d)) perform well, extracting clear and continuous ridge lines. However in Fig. 2, where the three algorithms are applied to a poor quality fingerprint image, the result obtained by the proposed algorithm is superior to the other two algorithms. Note that the ridge lines extracted by each method do NOT contain any post-processing step. The listed figures of ridge line extraction in Fig.1 and Fig.2 are directly obtained from the experiments.

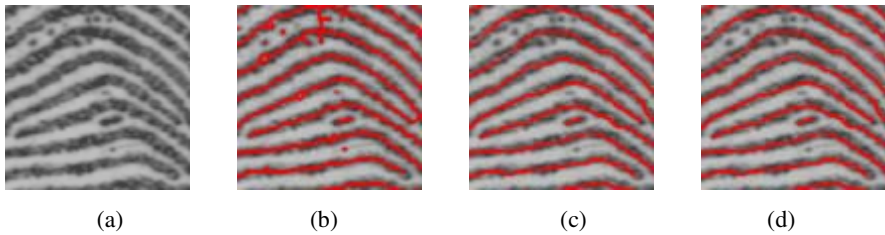


Fig. 1. Ridge line extraction on a good-quality fingerprint image. Fig.1 (a) is the original fingerprint image. Fig.1 (b) shows the ridge lines extracted using local binarization [7]. Fig.1 (c) is the ridge lines extracted using the algorithm in [4]. Fig.1 (d) depicts the extraction result using the proposed algorithm.

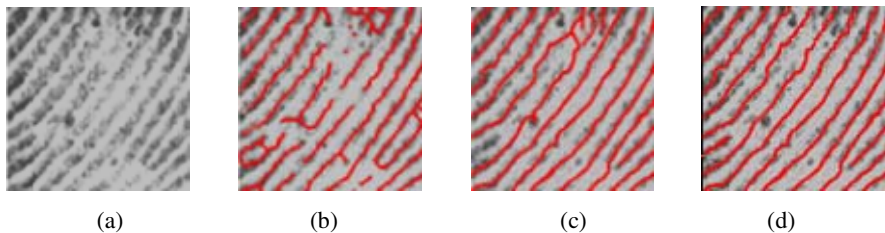


Fig. 2. Ridge line extraction on a poor-quality fingerprint image. Fig.2 (a) is the original fingerprint image. Fig.2 (b) shows the ridge lines extracted using local binarization [7]. Fig.2 (c) is the ridge lines extracted using the algorithm in [4]. Fig.2 (d) depicts the extraction result using the proposed algorithm.

Through exhaustive experiments on a vast set of NIST [8] fingerprint images, the proposed algorithm is proved to be much more effective and robust than existing schemes.

6 Conclusion

Extraction of fingerprint ridge lines is a critical pre-processing step in the identification of minutia in fingerprint images. The problem of extracting fingerprint ridge lines in poor quality images is challenging. Conventional methods follow the binarization and thinning steps, which can only deal with high-quality fingerprint images. However in this paper, we propose an effective ridge-line extraction approach by adopting the standard CONDENSATION algorithm in computer vision to *track* the ridge lines. In spite of greater computational complexity, this method outperforms the existing best-known techniques when tested on both good-quality and poor-quality fingerprint images.

Our future work is to generate fingerprint representations normalized with respect to scale and rotation. Such representations are to be worked out to avoid rotating fingerprint images and to match fingerprints among all possible orientations in print-to-print matching.

References

1. D. Maio and D. Maltoni: Direct Gray-scale Minutiae Detection in Fingerprints. *IEEE Trans. Pattern Anal. Mach. Intell.* 19(1997) 27-40
2. A.Blake and M.Isard. The CONDENSATION algorithm – conditional density propagation and applications to visual tracking. *Advances in Neural Information Processing System*. 9(1996) 361-368
3. Patrick Perez, Jaco Vermaak, and Andrew Blake. Data Fusion for Visual Tracking with Particles. Invited paper in proceedings of the IEEE, 92(2004) 495-513
4. Rui Ma, Yaxuan Qi, et al. A Novel Approach to Fingerprint Ridge Line Extraction. To appear in Proc. of International Symposium on Communications and Information Technologies 2005
5. Carlotta Domeniconi, Sibel Tari, and Ping Liang: Direct Gray Scale Ridge Reconstruction in Fingerprint Images. *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*. 5 (1998) 2941 – 2944
6. Michael Isard and Andrew Blake: Contour Tracking by Stochastic Propagation of Conditional Density. In Proc. of European Conf. on Computer Vision. 1(1996) 343-356
7. B. Moayer and K. Fu: A Tree System Approach for Fingerprint Pattern Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (1986) 376-387
8. NIST fingerprint databases:
<http://www.itl.nist.gov/iad/894.03/databases/defs/dbases.html#finglist>

A New Method for Human Gait Recognition Using Temporal Analysis

Han Su^{1,2} and Fenggang Huang¹

¹ School of Computer Science and Technology, Harbin Engineering University,
Harbin 150001, China

susuhan1016@yahoo.com.cn, fgghuang@163.com

² School of Computer Science, Sichuan Normal University, Chengdu 610066, China

Abstract. Human gait recognition is the process of identifying individuals by their walking manners. The gait as one of newly coming biometrics has recently gained more and more interests from computer vision researchers. In this paper, we propose a new method for model-free recognition of gait based on silhouette in computer vision sequences. The silhouette shape is represented by a novel approach which includes not only the spatial body contour but also the temporal information. First, a background subtraction is used to separate objects from background. Then, we represent the spatial shape of walker and their motion by the temporal matrix, and use Discrete Fourier analysis to analyze the gait feature. The nearest neighbor classifier is used to distinguish the different gaits of human. The performance of our approach is tested using different gait databases. Recognition results show this approach is efficient.

1 Introduction

Biometrics technologies verify a person's identity by analyzing human characteristics such as fingerprints, facial images, irises, gait, heat patterns, keystroke rhythms, and voice recordings. Gait as a biometric has particular advantages over other biometrics: it can be performed at a distance or at low resolution, while other biometric need higher resolution. Apart from this, it is difficult to disguise, and it requires no body-invading equipment to capture gait information. Medical studies [1] suggest that gait is unique if all gait movements are considered. In these cases, gait recognition is an attractive biometric and becoming increasingly important for surveillance, control area etc. More and more researchers have devoted to this field.

Early approaches to automated recognition by gait used marked-based technology, which needs expensive specialized hardware. Most recently approaches based on computer vision extract features for recognition from image sequences. Current approaches to gait recognition can be broadly divided into two categories: model-free ones [2-6] that deal directly with image statistics and model-based ones [7],[8] that recover the structure of a person in a model, interpret the model and then analyze the variation of its parameters. The majority of current approaches are model-free. The model-free approaches making no attempt to recover a structural model of a human subject are simple and fast. But the silhouette appearance information is only indirectly linked to gait dynamics. The dynamics and temporal information are important

element to improve the performance of recognition. And some researchers [2-6] try to extract these information from video sequences.

In this paper, a new gait recognition approach combining static with temporal information for analyzing and extracting human gait is presented. To obtain the full temporal information that is easy to extract by the model-based approaches, we propose a novel method to describe the silhouette shape and analyze shape changes. Such a description is useful in video surveillance. Using this method, we can extract the temporal information with low cost of computation by analyzing the variation of spatial contour matrix. First, a background subtraction is used to separate objects from background. Then, we represent the spatial shape of walker and their motion. And we use Discrete Fourier analysis to analyze the spatial contour feature matrixes and the temporal matrixes and represent the gait features. Our approach assumes that the camera is static, our view is invariant and the persons walk with constant velocity.

2 Silhouette Extraction

In our experiments, we use a background subtraction technique to extract the motion silhouette in each frame. Firstly, we model the background image. Note that the static background pixels sometimes include non-background pixels mainly due to the shadows, and the silhouettes can contain holes while directly subtracting the foreground image from original image. Then our foreground images are obtained by thresholding and applying morphology after background subtraction. Fig. 1 shows an example of silhouette extraction.

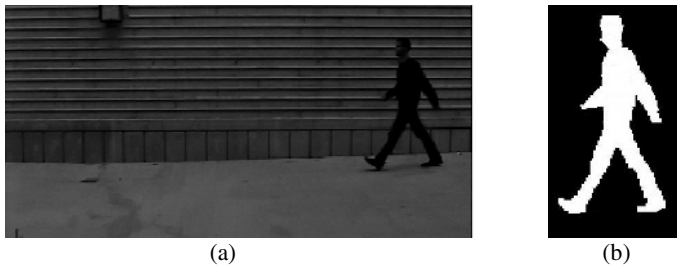


Fig. 1. An example of silhouette extraction and contour extraction. (a) is the original image and (b) is the extracted object

We place a window around the subject in order to reduce the computational complexity. The silhouette walks within this bounding box. And the silhouette contour is extracted by our preview method.

Murray et al.[1] studied the walking pattern and suggested that human gait is a form of periodic motion. And our previous work found the variance of contour is the important information for gait analysis based on silhouette. We consider gait cycle as a function of the silhouette width and height over time and detect gait cycle from the successive peak values of the contour width (height).

3 Gait Feature Analysis

3.1 Gait Representation and Feature Extraction

The silhouette shape contains structural as well as dynamical information of gait. For the gait recognition approaches based on silhouette analysis, how to describe the shape is a key element to the following feature extraction and feature analysis process. Effective shape descriptions should include full spatial gait features and easy to extract temporal variances. [6] describes spatio-temporal shape by a generalized symmetry operator. [4] describes the shape of the motion with a set of features derived from moments of a dense flow distribution. Liang Wang et al.[5] compute the shape centroid and describe shape as a vector of ordered complex numbers with boundary elements. They use spatial edge contours to approximately represent temporal patterns of gaits.

Different from them, we represent shapes as a set of boundary pixel. Fig.2 shows the representation of the silhouette's edge points.

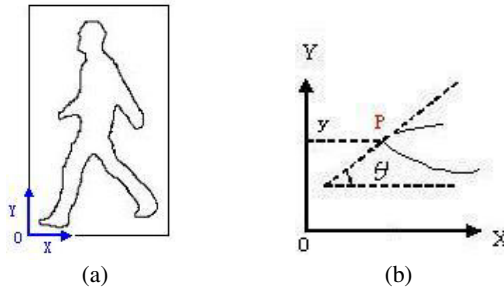


Fig. 2. (a) and (b) are the representations of the silhouette's boundary

One point P_j on the boundary can be representing as $P_j = y_j + \theta_j i$, where y_j is the vertical distance between the origin and P_j , θ_j is the angle from P_j to x-axis in a plane. The variance of θ_j describes the variational orientation of one edge point. For the following effective description of the shape changes over time, we represent each boundary point as the complex number. Y-axis represents the real part of complex number and angle represents the imaginary part. The distance between the boundary point and the origin is absolute distance which would not change with the shape centroid. Therefore, each image will be converted into an associated matrix of shape contour and each gait sequence will be accordingly converted into temporal pattern matrix sequence. We normalize the bounding box of all frames to the same dimensions $H \times W$ before represent the silhouette contour.

We name the matrix of shape contour X the spatial contour matrix. $X_i(t)$ is the spatial contour matrix of the t th frame in one's i th sequence. $X_i(t)$ includes many structure information such as body width, height and rough somatological proportion of walker etc. But it merely indirectly describes the changes of silhouette shape over

time. For including the temporal information in our gait feature, we propose a novel method to represent the temporal changes based on spatial contour matrix with low cost of computation.

We define the algorithm of one moving boundary point in the $t1$ th frame and $t2$ th frame as:

$$x_{nm}(t1) \cdot x_{pq}(t2) = |y_{t1} - y_{t2}| + |\theta_{t1} - \theta_{t2}| i \quad (1)$$

where $x_{nm}(t1) = y_1 + \theta_1 i$, $x_{pq}(t2) = y_2 + \theta_2 i$, $x_{nm}(t1)$ and $x_{pq}(t2)$ are the element of $X(t1)$, $X(t2)$, respectively.

According to the idea of Kronecker product, we define $X(t1) \otimes X(t2)$ as:

$$X(t1) \otimes X(t2) = \begin{bmatrix} x_{11}(t1)X(t2) \cdots x_{1w}(t1)X(t2) \\ x_{21}(t1)X(t2) \cdots x_{2w}(t1)X(t2) \\ \vdots \\ x_{h1}(t1)X(t2) \cdots x_{hw}(t1)X(t2) \end{bmatrix} \quad (2)$$

where $x_{ij}(t1)X(t2)$ is in term of (1). $X(t1) \otimes X(t2)$ represents the whole shape change between the $t1$ th frame and $t2$ th frame. Here, $X(t1) \otimes X(t2)$ is not the Kronecker product. To reduce the computation, we calculate $x_{nm}(t1)X^{\wedge}_{nm}(t2)$ instead of $x_{nm}(t1)X(t2)$. $X^{\wedge}_{nm}(t2)$ is a subset of $X(t2)$. It changes with the $x_{nm}(t1)$. $X^{\wedge}_{nm}(t2)$ is a $d \times d$, $d \ll H, W$ sub-matrix and center of $x_{nm}(t1)$. d is decided by interval between two frames. Different from the Kronecker product,

$$x_{nm}(t1)X^{\wedge}_{nm}(t2) = \min_{p,q} \{x_{nm}(t1) \cdot x_{pq}(t2)\} \quad (3)$$

where $n - d + 2 \leq p \leq n + d - 2, m - d + 2 \leq q \leq m + d - 2$.

Then we obtain the temporal matrix

$$T(t1, t2) = X(t1) \otimes X(t2) = (t_{nm}) \in C^{hw} \quad 1 \leq t1, t2 \leq N \quad (4)$$

where $t_{nm} = x_{nm}(t1)X^{\wedge}_{nm}(t2) = \Delta y + \Delta \theta | i$, N is the number of spatial contour matrixes in a sequence. (4) gives the effective temporal information from the gait sequences. $T(ti, tj)$ can not only analyze walker by their spatial shape but also by their temporal shape motion.

The temporal matrix $T(ti, tj)$ has two properties: $T(ti, ti) = 0$ and $T(ti, tj) = T(tj, ti)$.

3.2 Feature Analysis

In order to eliminate the influence of spatial scale, different walking pace, sequence length and original pose state, we normalize these sequences firstly. The parameters we used in this paper include $X(t1)$ and T . $X(t1)$ is the averaging of a person's all

original contour matrixes and T derives by summing all the temporal matrixes . To remove the disadvantage of unbalanced numerical values between $|\Delta y|$ and $|\Delta \theta|$, we need normalize t_{nm} :

$$\hat{t}_{nm} = \frac{|\Delta y|}{\min_{T(i,tj)}(|\Delta y|)} + \frac{|\Delta \theta|}{\min_{T(i,tj)}(|\Delta \theta|)} i . \tag{5}$$

Having normalized, we replace t_{nm} by \hat{t}_{nm} and replace $T(ti,tj)$ by $T^{\wedge}(ti,tj)$.Then we calculate T .

Our approach adopt Discrete Fourier Transform to analyze the gait feature $X(t1)$ and T for a given image sequence.

$$F(u, v) = \sum_{t=0}^{N-1} \left[\sum_{t=0}^{N-1} T(x, y) e^{-j(\frac{2\pi}{N})(vy)} \right] e^{-j(\frac{2\pi}{N})(ux)} . \tag{6}$$

We select the magnitude spectrum of Discrete Fourier frequency as the gait feature because the phase spectrum includes less information. The analysis results still contain a lot of descriptors. But we can see that lots of frequency parts are centralized at the low frequency position. Most of the visually significant information is concentrated in just a few coefficients of the DFT. The general features which have the highest inter-class variance are found in the low frequency. Then we discard the high frequency, and use the low-order descriptors to be the gait features. Using these gait features not only hold the primary information, but reduce the dimensions of data as well. The distance between classes is measured by the Euclidean distance.

4 Experiments

To evaluate our method, we tested it on some gait databases. We use the nearest neighbor classifier and leave-one-out cross validation technique. First, we evaluate our approach for the data set of Little and Boyd. The data set consist of 40 image sequences and 6 different subjects. Then, we train and test on the CMU MoBo data set. The data set contain 6 simultaneous motion sequences of 25 subjects walking on a treadmill. Image resolution is 640×480 . Each subject is recorded performing four different types of walking. We test our method on the slow walk. To reduce the computation, we normalize the bounding box resolution to 200×300 . The recognition rates on the two databases are 100% and 93.1%, respectively. We also test our method on the data set of Little and Boyd and NLPR gait database for varying low resolution analysis. NLPR gait database include 80 sequences from 20 subjects and four sequences per subject. The original resolution is 352×240 . We choose some images and normalize them to 128×128 , 64×64 and 32×32 , respective. Table 1 shows a better recognition rate. This suggests that our approach can work on low resolution condition.

Table 1. Recognition results for reduced resolution

Image size	CCR(%)	
	Little and Boyd	NLPR(0 ⁰)
128× 128	98%	80.1%
64× 64	91.7%	76.4%
32× 32	80%	67.1%

5 Conclusions

In this paper, we present an automated model-free approach for human identification combining static and temporal information of walking gait from low resolution video. To obtain the full temporal information that is easy to extract by the model-based approaches, we propose a novel method to describe the silhouette shape and analyze shape changes. Using this method, we can extract the temporal information with low cost of computation by analyzing the variation of spatial contour matrix. Our approach is tested on some gait databases and recognition results show this approach is efficient. Furthermore, our method shows a better recognition rate in low resolution condition. In future, we will pay more attention to the feature space for describing and recognizing the human gait on the larger database of subjects.

References

1. Murary, M.P., Drought, A.B., Kory, R.C.: Walking Pattern of Movement. *American Journal Medicine* 46 (1967) 290–332
2. Wang, L., Tan, T.N., Hu, W.M., Ning, H.Z.: Automatic Gait Recognition Based on Statistical Shape Analysis. *IEEE Transactions on Image Processing* 12 (2003) 1120–1131
3. Niyogi, S.A., Adelson, E.H.: Analyzing and Recognizing Walking Figures in XYT. In: *Computer Vision and Pattern Recognition 1994*, Seattle, USA (1994) 469–474
4. Little, J., Boyd, J.: Recognising People by Their Gait: the Shape of Motion. *Videre* 1 (1998) 1–32
5. Wang, L., Tan, T.N., Ning, H.Z., Hu, W.M.: Silhouette Analysis-Based Gait Recognition for Human Identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25 (2003) 1505–1528
6. Hayfron-Acquah, J., Nixon, M.S., Carter, J.N.: Human identification by spatio-temporal symmetry. In: *Proc. of Intl. Conf. on Pattern Recognition*, Quebec, Canada (2002) 632–635
7. Yam, C.Y., Nixon, M.S., Carter, J.N.: Extended model-based automatic gait recognition of walking and running. In: *Proc. of 3rd Intl. Conf. on Audio and Video-based Biometric Person Authentication*, Halmstad, Sweden (2001) 278–283
8. Cunado, D., Nixon, M.S., Carter, J.N.: Automatic Extraction and Description of Human Gait Models for Recognition Purposes. *Computer Vision and Image Understanding* 90 (2003) 1–41

Microcalcification Patterns Recognition Based Combination of Autoassociator and Classifier*

Wencang Zhao¹, Xinbo Yu², and Fengxiang Li³

¹ College of Automation and Electronic Engineering,

Qingdao University of Science & Technology, Qingdao 266042, China

² Department of Oral Medicine, the Affiliated Hospital of Medical College,

Qingdao University, Qingdao 266003, China

³ Department of Gynecology, People's Hospital of Jimo, Qingdao 266200, China

zhaocenter-conference@yahoo.com.cn

{xinboyu, fengxiangli}@qingdaonews.com

Abstract. This paper presents a microcalcification patterns recognition method based autoassociator and classifier to detect the breast cancer. It studies the autoassociative and classification abilities of a neural network approach to classify the microcalcification patterns into Benign and Malignant using some certain image structure features. The proposed technique used the combination of two kinds of neural networks, autoassociator and classifier to analyze the microcalcification. It could obtain 88% classification rate for testing dataset and 100% classification rate for training dataset.

1 Introduction

Breast cancer remains the most common cause of cancer deaths among women. The stage of development depends upon the detection time, early detection prevents patient to be pass through high stage traumatic treatment and increases chance of survival. There has been a considerable decline in mortality from breast cancer in women because of the Breast Screening programs. Although the mortality rate from breast cancer has been decreasing at the rate of 3.7% per year, the incidence of breast cancer is rising at 1.5% per year [1]. Thus prevention and an early detection of breast cancer tumors are immediate demand of society.

Mammography continues to be regarded as one of the most useful technique for early detection of breast cancer tumor. The presence of microcalcifications in breast tissue is one of the main features for its diagnosis [2, 3]. In many cases of screening database, microcalcifications exhibit both class (benign and malignant) characteristics. Interpretation of such cases often produce screening errors; either to miss malignant cases or more unnecessary biopsies. A higher prognostic rate is anticipated by combining the mammographer's interpretation and computer analysis.

Recent revolution in image processing techniques makes even smaller microcalcification detection easier, though classification of malignant and benign microcalcifications is still a challenging issue for researchers. It's clear from

* This research was supported by the National 863 Natural Science Foundation of P. R. China (2001AA636030) and the Doctoral Fund of Qingdao University of Science & Technology.

previously proposed methods [2-18]; selection of significant features, type of classifiers, and topography for particular classifiers are most important factors in pattern classification process.

The most used features are region-based features [9], shape based features [10], image structure features [11-13, 18], texture based features [16, 14-15], and position related features.

Different methods have been used to classify microcalcifications in digital mammograms. Most common are Statistical methods [3, 11] and Artificial Neural Networks (ANN) [2, 5, 9-18]. Shen et al [10] achieved 94% classification for benign calcifications and 87% for malignant calcifications using one-hidden layer Backpropagation artificial neural network. A comparative study of statistical methods and ANN for classification carried out by woods et al. [11]. Quin et al [9, 17] used a Multistage Neural Network.

The main objective of this research work is to investigate the autoassociative and classification abilities of a neural network approach for the classification of microcalcification patterns into benign and malignant using some certain image structure features. Two neural networks such as an autoassociator and a classifier are combined and used to classify benign and malignant microcalcifications. All the features are set as input and desired output of autoassociator neural network (AANN). Hidden neuron values of trained AANN are taken as input to classifier NN. Final output of classifier NN evaluate into two classes: benign and malignant.

2 Research Methodology

2.1 Digital Mammograms

Digital Mammograms in this paper is taken from Digital Database for Screening Mammography (DDSM) established by University of South Florida. DDSM provides mammograms with ground truth and other information. Suspicious area is already marked in all digital mammograms of DDSM by at least two expert radiologists. Boundary of each abnormality is specified by chain code. DDSM contains total of 126 calcification areas; 84 (43 benign, and 41 malignant) areas in training set, and 42 (21 benign, and 21 malignant) areas in testing set.

2.2 Feature Extraction

14 image structure features: number of pixels, histogram, average grey, average boundary grey, contrast, difference, energy, modified energy, entropy, modified entropy, standard deviation, modified standard deviation, skew and modified skew are calculated using extracted grey level values of each suspicious areas and surrounding boundary areas [6, 18].

2.3 Classifier

The research method combines two neural networks: autoassociator and classifier for classification of microcalcification pattern into benign and malignant. Both neural networks have a single hidden layer architecture and uses backpropagation learning algorithm for training.

Autoassociative Neural Network (AANN) takes input and output the same feature vector. 14 extracted features were feed to autoassociator as input and same considered as desired output for training. Autoassociative networks are typically used for tasks involving pattern completion. A trained AANN returns a composite of its learned patterns most consistent with the new information. Network parameters like number of hidden neuron and iterations, learning rate and momentum adjusted during network training to obtain unique patterns for benign and malignant classes for classification.

Hidden neuron values from trained AANN are taken as input to classifier neural network. Classifier neural network has 2 output nodes in output layer, which represents each class. The desired output set as (0.9, 0.1) for malignant and (0.1, 0.9) for benign class. Different network topologies tried, adjusting number of hidden neurons in hidden layer and iterations, in classifier network training to achieve higher classification rate.

3 Experiments

Five modules have been defined separately and integrated together to perform entire classification process. The five modules are 1) area extraction, 2) feature extraction, 3) feature normalization, 4) autoassociation of given patterns, and 5) classification.

A total of 126 calcification areas, containing 84 (43 benign, and 41 malignant) for training and 42 (21 benign, and 21 malignant) for testing, are used for experiments. All Experiments are carried out with Learning rate 0.1 and momentum 0.1 for both the network: Autoassociator and classifier.

For AANN many experiments using different network parameters were run to generate unique patterns for both classes which enable classifiers to produce higher classification rate. RMS error is considered as evaluating criteria for AANN. Different number of hidden neurons (6-26) used for AANN training. To check RMS error values respective to number of hidden neuron for AANN training some experiments carried out with fixed 100000 iterations.

The hidden neurons value obtained after AANN training is used as input to classifier neural network. Different number of hidden neurons and iterations were used for both neural networks to produce higher classification rate on both training and testing dataset. In many experiments we achieved 100% classification rate on training dataset.

The patterns obtained from AANN trained with 10 hidden neurons and 50000 iterations achieved very good classification on training and testing datasets using various hidden neurons and iterations for classifier network. Table 1 shows the results of experiments performed on patterns obtained from AANN trained with 10 hidden neurons for 50000 iterations. For training dataset 100% classification rate obtained for all hidden neuron from 6 to 20 (using even numbers). While for testing dataset classification rate reached up to 88% using 10 hidden neurons and 20000 iterations.

Experimental results show (Table 1) that increase in number of iterations improves training classification rate with same number of hidden neurons but it affects testing classification rate. Fig. 2 shows the performance of classifier network on training and testing dataset with increase in iterations for training.

Table 1. Classification rate when 10 hidden neuron and 50000 iterations were used for AANN Training

Classifier NN			
Network Parameters		Classification Rate (%)	
Hidden Neurons	Iterations	Training	Testing
6	10000	94	85.7
6	70000	100	73.8
8	10000	95.2	81
8	50000	100	71.4
10	20000	96.4	88.1
10	50000	100	69
12	10000	94	83.3
12	40000	100	73.8
16	5000	90.5	85.7
16	50000	100	69
14	5000	90.5	81
14	30000	100	66.7
18	5000	90.5	83.3
18	40000	100	76.2
20	5000	89.3	85.7
20	50000	100	69

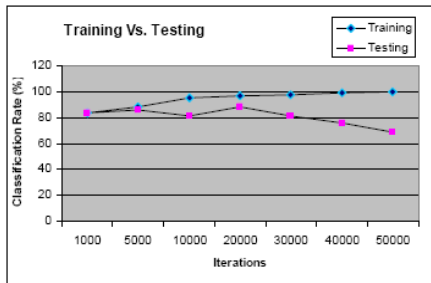


Fig. 1. Performance of Classifier Network on increasing in number of iterations in training

4 Results Analysis

Many experiments were run using 14 image structure features to classify microcalcifications into benign and malignant. The highest classification rate obtained was over 88% for testing dataset and 100% for training dataset.

With most of the experiments with increase in iterations during classifier training leads to higher training classification rate but results in decrease in testing classification rate (see Fig. 1). This is because the neural network is overtrained on the training data.

It is prime importance for all the classification system to differentiate the given breast abnormalities in to proper classes minimizing the false positive errors (Benign classified as Malignant: increase in unnecessary biopsies) and false negative errors (Malignant classified as Benign: early misclassification leads to last stage invasive cancers), rather to achieve highest classification rate.

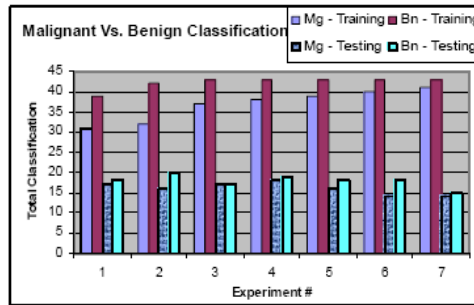


Fig. 2. Benign and Malignant Classification for both dataset

Fig. 2 shows that classification of benign microcalcifications is higher than malignant microcalcifications on both training dataset and testing dataset.

In many experiments (Table 1) maximum classification rate achieved with small number of iterations. It shows that Autoassociative network is producing distinct patterns for given classes thus classifier network needs less training to converge onto solution.

5 Conclusions

In this paper we proposed a method which combined two neural networks: autoassociator and classifier for the classification of microcalcification patterns to detect the breast cancer. We obtained highest classification rate over 88% on testing dataset and 100% on training dataset. Over 85% classification rate for testing dataset was consistent with many network configurations for both the networks. The results obtained from our work explored AANN capabilities experimenting more with different network topologies and learning algorithms to produce unique patterns for given classes. For such unique patterns some non-iterative classifiers can be used for classification process, to save both time and cost on classifier network training.

References

1. <http://www.nbcc.org.au>
2. Halkiotis S., Mantas J., Botsis T.: Computer-Aided Detection of Clustered Microcalcification in Digital Mammograms. Proceedings of the fifth European Systems Science Congress (2002)
3. Salifty M. F., Kaufmann- G. H., Granitto P., Ceccatto H. A.: A Computer-Aided Diagnosis Method for Automated Detection and Classification of Microcalcifications in Mammograms. Proceedings of the Argentine Symposium on Healthcare Informatics, Tandil (2000) 41-47
4. Kupinski M. A., Giger M. L.: Feature Selection with limited datasets, Medical Physics. Vol. 26, No. 10 (1999) 2176-2182

5. Tourassi G. D., Frederick E. D., Markey M. K., Jr. Floyd C. E.: Application of the mutual information criterion for feature selection in computer-aided diagnosis. *Medical Physics*, Vol. 28, No. 12 (2001) 2394-2402
6. Zhang P., Verma, B., Kumar K.: Neural vs. Statistical Classifier in Conjunction with Genetic Algorithm Feature Selection in Digital Mammography. *IEEE Congress on Evolutionary Computation*, Vol. 2 (2003) 1206-1213
7. Yang J., Honavar V.: Feature Subset Selection using A Genetic Algorithm. *IEEE Intelligent Systems*, Vol. 13 (1998) 44-49
8. Kwak N., Choi C-H: Input Feature Selection for Classification Problems. *IEEE Transaction on Neural Networks*, Vol. 13, No 1 (2002) 143- 159
9. Zheng B., Qian W., Clarke L.: Multistage Neural Network for Pattern Recognition in Mammogram Screening. *International Conference on Neural Networks, IEEE International Conference on Neural Networks (ICNN)*, Orlando, FL, USA (1994)
10. Shen L., Rangayyan R. M., Leo Desautels J. E.: Detection and Classification of Mammographic Calcifications. *State of The Art in Digital Mammographic Image Analysis*. World Scientific Publication, Vol. 9 (1994) 198-212
11. Woods K. S., Solks J. L., Priebe C. E., Kegelmeyer W. Phillip, Doss C. C., Bowyer K. W.: Comparative Evaluation of Pattern Recognition Techniques for Detection of Microcalcifications in Mammography. *State of The Art in Digital Mammographic Image Analysis*, World Scientific Publication, Vol. 9 (1994) 213-231
12. Verma, B. K.: A Neural Network Based Technique to Locate and Classify Microcalcifications in Digital Mammograms. *Proceedings of IEEE World Congress On Computational Intelligence, WCCI'98, USA (1998) 2163-2168*
13. Verma B.K.: Comparative Evaluation n of Two Neural Network Based Techniques for Classification of Microcalcifications in Digital Mammograms. *Knowledge and Information Systems: An International Journal*, Vol. 1, No.1, Springer-Verlag (1999) 10-20
14. Christoyianni L., Dermatas E., Kokkinakis G.: Fast Detection of Masses in Computer-Aided Mammography. *IEEE Signal Processing Magazine*, Volume: 17, Issue: 1 (2000) 54-64
15. Zhao, W.C., Ji, G.R., Feng, C., Nian, R.: Blindly Selecting Method of Training Samples Based Hyper-spectral Image's Intrinsic Character for Object Recognition. *Proceedings of 2005 IEEE International Workshop on VLSI Design and Video Technology, Suzhou, China (2005) 113-116*
16. Bovis K., Singh S.: Detection of Masses in Mammograms using Texture Measures. *Proceedings of 15th International Conference On Pattern Recognition, Barcelona, IEEE Press, Vol. 2 (2000) 267-270*
17. Qian W., Sun X.J.: Digital Mammography: Wavelet Transform and Kalman-Filtering Neural Network in Mass Segmentation and Detection. *Academic Radiology*, Vol. 8, No. 11 (2001).
18. Verma B.K. Zakos J.: A Computer-Aided Diagnosis System for Digital Mammograms Based On Fuzzy-Neural and Feature Extraction Techniques. *IEEE Transactions On Information Technology in Biomedicine*, Vol. 5 (2001) 46-54

Improved Method for Gradient-Threshold Edge Detector Based on HVS

Fuzheng Yang, Shuai Wan, and Yilin Chang

National Key Lab. on ISN, Xidian University, Xi'an 710071 China
{fzhyang, shwan, ylchang}@mail.xidian.edu.cn

Abstract. This paper presents an improved method which is suitable for gradient-threshold edge detectors. The proposed method takes into account the basic characteristics of the human visual system (HVS) and precisely determines the local masking regions for the edges with arbitrary shape according to the image content. Then the gradient image is masked with the luminance and the activity of local image before edge labelling. The experimental results show that the edge images obtained by our algorithm are more consistent with the perceptive edge images.

1 Introduction

Edge detection plays important roles in various areas of image analysis and computer vision. The edge representation of an image drastically reduces the amount of data to be processed, yet it retains important information about the shapes of objects in the scene. This description of an image is easy to integrate into a large number of object recognition algorithms used in computer vision and other image processing application. A lot of methods for detecting edges have been proposed [1] [2]. Today the gradient-threshold based edge detectors are widely used, for example, Sobel, Prewitt, and Canny detectors [3]. The experiment results showed that the Canny algorithm had the highest performance when the parameters were adapted for each image [4].

The most commonly proposed schemes for gradient-threshold edge detection include three steps: smoothing image, image differentiation and edge labelling [1]. The last step is critical for the quality of the edge map. Usually a global threshold is used for edge labelling. An improvement has been achieved by the use of local threshold techniques, which use local feature for selecting thresholds and operate on local image [5]. However, the characteristics of human visual system (HVS) are not well considered in the existing gradient-threshold edge detectors. The edge maps obtained by these detectors may be inconsistent with human perception. We proposed an improved method suitable for gradient-threshold edge detectors, where the basic characteristics of the HVS are taken into account [6].

This paper proposes an improved method suitable for gradient-threshold edge detectors based on the one proposed in [6]. Taking HVS characteristics into account, the proposed method in this paper precisely determines the local masking regions for the edges with arbitrary shape according to the image content. Then

the gradient image is masked with the luminance and the activity of local image before edge labelling. The edges detected by this method are much more consistent with the perceptive edges than those by traditional methods. The rest of this paper is organized as follows. Section 2 presents the basic idea of the improved edge detection method. The description of the edge detection algorithm is given in Section 3. The experimental results presented in Section 4 demonstrate the performance of our approach. Section 5 concludes this paper and points out future research directions.

2 Basic Idea

The human observer is the end user for the most general vision system so the human visual system was selected for evaluating edge detectors at the present time [4]. However, the characteristics of HVS are not well taken into account in the existing gradient-threshold edge detectors. The edge maps obtained by these detectors may be inconsistent with human perception.

We have proposed a HVS based method suitable for gradient-threshold edge detectors [6], where the gradient image is masked with luminance and activity before edge labelling. Spatial activity masking is one of the key techniques of this kind method, and the determination of the local masking regions is of great importance in activity masking. In traditional methods [6], the direction of the gradient of a pixel is considered as vertical to the edge, and the neighbor rectangle regions vertical to the direction of the gradient will be used for local masking, as shown in Fig. 1(a). The arrow indicates the computed direction of the gradient at pixel (x^*, y^*) , and the line cross the pixel is the edge to be masked. Then R1 and R2 are the determined local masking regions.

However, the direction of the gradient is easily affected by noise, which often results in a slanting direction not vertical to the actual edge. As shown in Fig. 1(b), the computed direction of the gradient at pixel (x^*, y^*) is not vertical to the actual edge, so the determined local masking regions R1 and R2 are the inconsistent with the actual local activity.

On the other hand, the actual edges have arbitrary shapes. The rectangle local masking regions always disaccord with the perceptive image activity in activity masking. For a curving edge, the masking regions should be properly determined according to the edge shape and the HVS characteristics. For example, in Fig. 1(c) the regions R1 and R2 along with the curving edge should be chosen for local activity masking.

3 The Proposed Method

Like the other gradient edge detectors, the proposed algorithm smoothes and differentiates the image to get the gradient image $grad(x, y)$ (magnitude and direction) firstly. Then we label the edges with a low threshold to get the prime edge image $pedge(x, y)$, which includes all the potential edges. To get the perceptive edges, the gradient image $grad(x, y)$ is further masked with the local

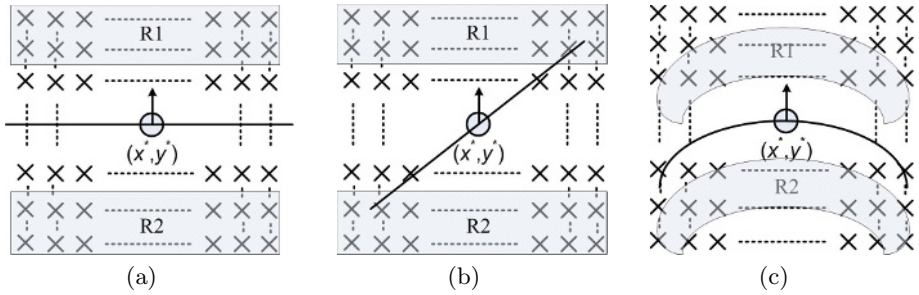


Fig. 1. The local regions for activity masking

luminance and activity, and then edge labelling is performed on the masked gradient image using another threshold. The keys of the improved method for detecting perceptible edges are to determine the mask regions and the local activities. It is very complex to determine the gradient direction and the edge shape for every pixel exactly. Therefore, we propose an improved method for precisely determining the activity masking regions, as described in Section 3.1.

3.1 Determine the Activity Masking Regions

The masking regions for every pixel in the prime edge image are determined based on the shape of the edge. For example, let the pixel (x_1, y_1) be an edge pixel in the prime edge image. In order to determine its masking regions, we first find out the pixels belong to the potential edges in the 5×5 sized region where the pixel (x_1, y_1) is the geometrical center. Every two of these edge pixels are linked to form a line, and for each line, compute the sum of the distances to the line of all the other edges pixels excluding the two pixels. One of the lines for the pixel (x_1, y_1) is shown in Fig. 2(a). The line achieves the minimum sum of the distances is determined to be the line to which the pixel (x_1, y_1) belongs, and the direction vertical to the line is determined to be the actual direction of the gradient at the pixel (x_1, y_1) , which is denoted by the arrowhead in Fig. 2(b).

Along the denoted actual direction, find the pixel (ex_1, ey_1) which is the first pixel with a gradient value smaller than $\lambda \cdot grad(x_1, y_1)$ and the pixel (ex'_1, ey'_1) with the distance of 3 pixels to the pixel (ex_1, ey_1) . Then for every edge pixel (x_i, y_i) in the 5×5 sized region where pixel (x_1, y_1) is the geometrical center, the pixels (ex_i, ey_i) and (ex'_i, ey'_i) are found out to form the activity masking region R_1 for the pixel (x_1, y_1) , as shown in Fig. 2(b). The masking region R_2 is determined using similar methods, where the direction against the arrowhead is used.

3.2 Luminance and Activity Masking

The gradient image $grad(x, y)$ is further masked with luminance and local activity using the following masking algorithm.

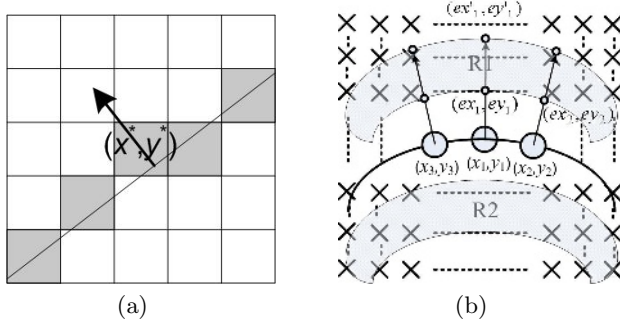


Fig. 2. Masking regions determination

If the pixel (x, y) is not a non-edge pixel in the prime edge image $pedge(x, y)$ its final masked gradient $grad'(x, y)$ is set to zero. Otherwise $grad(x, y)$ is masked with luminance and activity to get the perceptive gradient image $grad'(x, y)$ as Eq. 1 and Eq. 2. Take the pixel (x^*, y^*) for example.

$$grad_b(x^*, y^*) = \frac{grad(x^*, y^*)}{a_1 + \left(\frac{|b(x^*, y^*) - b_0|}{b_0}\right)^{r_1}} \tag{1}$$

where $grad_b(x^*, y^*)$ denotes the gradient masked with luminance, b_0 is the average luminance of the entire image, and $b(x^*, y^*)$ is the local average luminance [6], Parameters a_1 and r_1 are estimated experimentally. Eq. 1 explicates the contrast sensitivity characteristic and the luminance adaptation capability of HVS.

$$grad'(x^*, y^*) = \frac{grad_b(x^*, y^*)}{a_2 + m(x^*, y^*)^{r_2}} \tag{2}$$

Where parameters a_2 and r_2 are selected to match the variation of human visual sensitivity with the background activity, and the local activity $m(x^*, y^*)$ is computed as

$$m(x^*, y^*) = \min(m_1(x^*, y^*), m_2(x^*, y^*)) \tag{3}$$

where $m_1(x^*, y^*)$ and $m_2(x^*, y^*)$ are the activities of the regions R_1 and R_2 respectively. And the activities $m_i(x^*, y^*)$ of $R_i (i = 1, 2)$ can be expressed as follows.

$$m_i(x^*, y^*) = \frac{1}{M_i} \sum_{(x,y) \in R_i} grad(x, y) \tag{4}$$

where M_i is the number of pixels in region R_1 . Eq. 2 explicates the activity masking characteristic of HVS. Strongest masking occurs when the interacting stimuli have similar characteristics, i.e. similar frequencies, orientations, etc. Therefore,

the local activities are calculated by the spatial complexity with the shape of edges taking into account.

Then the edges can be labelled on the masked gradient image $grad'(x, y)$ using the threshold method.

4 Experimental Results

We give a simple implementation of the improved algorithm on the canny edge detector [3] in this paper. Canny detector uses Gaussian filter to smooth noise and then computes the gradient of each pixel in the smoothed image. Candidate edge pixels are identified as the pixels that survive a thinning process called nonmaximal suppression. Edge pixels are then labelled on the thinned edge magnitude image using the two thresholds method.

Our improved Canny algorithm is based on the *Matlab* function *edge*(image, 'canny'), in which the high threshold T_h is higher than the gradient values of 80% of all pixels and the low threshold T_l is set as $0.5 T_h$, and the standard deviation of the Gaussian filter σ is 1. We mask the gradient images with luminance and activity to get $grad'(x, y)$. Then we apply the nonmaximal suppression and edge labelling on $grad'(x, y)$ to get the edge images. In our implementation, the parameters are set as follows. $\lambda = 0.618$, $r_1 = 2$, $a_1 = 1.5$, $r_2 = 2$, $a_2 = 0.5$. They were determined by threshold experiments.

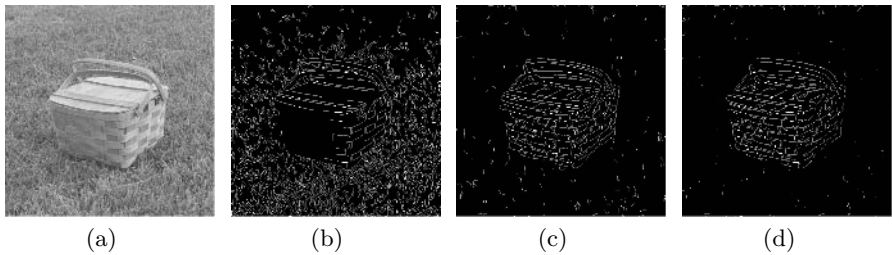


Fig. 3. Results for “Basket” image. (a) original image; (b) edge image obtained by Canny detector; (c) edge image obtained by the improved Canny detector in [6]; (d) edge image obtained by the proposed method.

Fig. 3 shows the original image and the edge images of “Basket”. From Fig. 3(b), we can observe that the edges detected by Canny detector is not consistent with our perception edges. For example, the obtained edge map preserves most of the lawn texture but losses some details of the grids on the basket. In Fig. 3(d), however, most of the potentially undesirable texture is eliminated and the details of interest are preserved well without a significant clutter on the lawn. The results of the image “Golf cart” are shown in Fig. 4, where similar results are obtained.

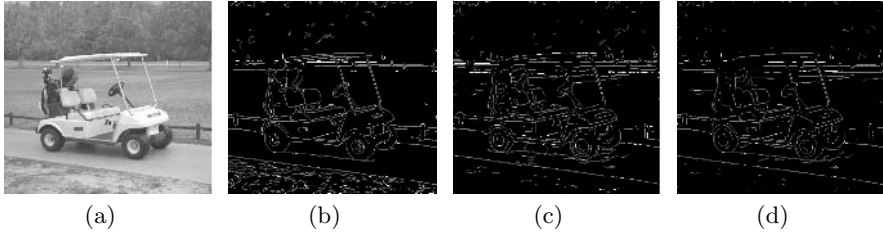


Fig. 4. Results for “Golf cart” image. (a) original image; (b) edge image obtained by Canny detector; (c) edge image obtained by the improved Canny detector in [6]; (d) edge image obtained by the proposed method.

5 Conclusion

The edges detected by traditional edge detectors may be inconsistent with the perceptive edges. In this paper, we propose an improved method based on HVS which is suitable for gradient-threshold edge detectors. In the proposed algorithm, the local masking regions for the edges with arbitrary shape are determined, and the gradient image is masked with the luminance and the activity of local image before the edges are labelled. The results show that the edge images obtained by our algorithm are more consistent with the perceptive edge images. The performance of the gradient-threshold edge detectors is much improved by this method, which is helpful to image partition, image analysis, and so on.

References

1. D. Ziou Djemel and S. Tabbone: Edge detection techniques-an overview. *Int. J. of Pattern Recognition and Image Analysis*, Vol. 8. (1998) 537 - 559
2. M. Basu: Gaussian-based edge-detection methods—a survey. *IEEE Trans. Systems, Man, and Cybernetics-Part C*, Vol. 32. (2002) 252 - 260
3. J. Canny: A computational approach to edge detection. *IEEE Trans. Pattern Anal. Machine Intell.* , Vol. 8. (1986) 679 - 698
4. M.D. Heath, S. Sarkar, T. Sanocki and K.W. Bowyer: A robust visual method for assessing the relative performance of edge-detection algorithms. *IEEE Trans. Pattern Anal. Machine Intell.* , Vol. 19. (1997) 1338 - 1358
5. A. Elmabrouk and A. Aggoun: Edge detection using local histogram analysis. *electronics letters*, Vol. 34. (1998) 1216 - 1217
6. F. Yang, Y. Chang and S. Wan: Gradient-threshold edge detection based on HVS. *Optical Engineering*, Vol. 44. (2005)

MUSC: Multigrid Shape Codes and Their Applications to Image Retrieval

Arindam Biswas¹, Partha Bhowmick¹, and Bhargab B. Bhattacharya²

¹ Computer Science and Technology Department,
Bengal Engineering and Science University, Shibpur, Howrah, India
{abiswas, partha}@becs.ac.in

² Center for Soft Computing Research,
Indian Statistical Institute, Kolkata, India
bhargab@isical.ac.in

Abstract. A novel technique for shape coding of a digital object based on its inner and/or outer isothetic polygonal shape, is proposed. The method uses a multigrid background for several applications including digital image visualization and retrieval with varying levels of accuracy. The elegance of the proposed shape code lies in capturing the shape of the objects present in an image from their gross appearances to finer details with a set of hierarchical isothetic polygons that tightly envelop (cover) the objects from outside (inside). Experimental results demonstrate the strength and efficiency of the proposed scheme.

1 Introduction

Encoding the shape information of an image is very fundamental to its visualization and retrieval. An efficient encoding both in terms of construction and storage is extremely important in the realm of object-based systems, especially in multimedia research. Encoded shape information play different roles depending upon the target application, vis-a-vis localization and retrieval of a specific shape.

Content-based arithmetic encoder (CAE) [5], run-length encoding [8], and polygon-based encoding [7] are some of the methods that deal with the problem of localization. However, the shape specification for localization, for example, shape coding based on rate-distortion [5], is not useful for retrieval as it involves complex decoding and contour extraction process. Similarly, shape descriptors for retrieval purpose using curvature scale space descriptors [6], geometric moments [4], etc. are not amenable to localization because the computation of these descriptors involves nonreversible transformation.

In this paper, a fast and efficient shape coding technique based on tight isothetic polygon(s) describing the object(s) on a multigrid background, is proposed. It stores the polygons for a given image imposed on coarse to finer background grids, with remarkably high storage efficiency. On a finer grid, it serves as a good encoder for visualization, whereas, for retrieval and matching purposes, at each finer level, one can use a smaller search space using this technique.

2 Proposed Work

A tight isothetic polygonal shape (TIPS) covering a 2D object can be determined using the algorithm TIPS, reported earlier [2, 3]. The algorithm TIPS has been modified here to obtain the complete set of isothetic polygons for multiple objects present in an image. Each polygon can be uniquely described, starting from a suitable vertex (start vertex), and the corresponding higher level description of the polygon has been coded properly to generate a shape code for the corresponding polygon. The set of polygons in a multigrid environment offers powerful features for the subsequent image retrieval process.

2.1 Tight Isothetic Polygons

Given a binary image \mathcal{I} containing one or more objects (connected components), and a set of uniformly spaced horizontal and vertical grid lines, $\mathcal{G} = (\mathcal{H}, \mathcal{V})$,¹ the corresponding set of tight isothetic polygonal envelopes, $\mathcal{P}(\mathcal{I}, \mathcal{G}) = \{P_i\}_{i=1}^n$, consists of n isothetic polygons, such that the following conditions are satisfied:

- (c1) $V_i \cap V_j = \emptyset, \forall i, \forall j, i \neq j$, where V_i and V_j represent the ordered set of vertices of P_i and P_j respectively;
- (c2) if P_i is the polygon that contains one connected component \mathcal{C} of \mathcal{I} , then no point $p \in \mathcal{C}$ should lie outside P_i ;
- (c3) each vertex of $P_i, i = 1, 2, \dots, n$, is an element of \mathcal{G} .
- (c4) $\sum_{i=1}^n \text{area}(P_i)$ is minimized.

It may be noted that, property (c1) ensures that no two polygons in $\mathcal{P}(\mathcal{I}, \mathcal{G})$ will intersect each other, although one polygon may contain another polygon. Further, the polygon may possess self-intersections.

2.2 Shape Code

Let V_{ri} be the ordered set of vertices for i th polygon P_{ri} in $\mathcal{P}(\mathcal{I}_r, \mathcal{G}) = \{P_{ri}\}_{i=1}^{n_r}$ corresponding to the r th digital image \mathcal{I}_r in a binary image database $\mathcal{D} = \{\mathcal{I}_r\}_{r=1}^N$ containing N images. Now, in $V_{ri} = \langle v_{ri}^{(1)}, v_{ri}^{(2)}, \dots, v_{ri}^{(m_{ri})} \rangle$, consisting of m_{ri} vertices, each vertex $v_{ri}^{(k)}$ will be either a 90° vertex or a 270° vertex, depending on the object containments in the four unit grid quadrants incident at $v_{ri}^{(k)}$. Further, in the set V_{ri} , there will be always at least one 90° vertex that will have object containment in its bottom-right unit-grid-quadrant and the other three quadrants free. One such vertex is selected as the ‘‘start vertex’’ to generate the shape code of P_{ri} . W.l.o.g., let $v_{ri}^{(1)}$ be the start vertex in V_{ri} .

It may be observed that, any two consecutive vertices in V_{ri} would form either an horizontal edge or a vertical edge of P_{ri} . Hence, if $(x[v_{ri}^{(j)}], y[v_{ri}^{(j)}])$ be the

¹ In this paper, we have also used \mathcal{G} to denote the set of grid points defined by \mathcal{H} and \mathcal{V} ; that is, the set of points of intersection of \mathcal{H} and \mathcal{V} , the meaning of the corresponding representation being evident from the context.

coordinates of the j th vertex in V_{ri} , then the distance (edge length) between two consecutive vertices $v_{ri}^{(k)}$ and $v_{ri}^{(k+1)}$ in V_{ri} is given by either $|x[v_{ri}^{(k)}] - x[v_{ri}^{(k+1)}]|$ or $|y[v_{ri}^{(k)}] - y[v_{ri}^{(k+1)}]|$, whichever is non-zero.

In the counter-clockwise enumeration of the vertices of P_{ri} , starting from the start vertex $v_{ri}^{(1)}$, the length of each edge and the type of the destination vertex of the edge are stored in eight-bit (one byte) representation, where the two most significant bits are reserved for the two types of vertices (90^0 and 270^0) and a false vertex (180^0) which acts as a continuity flag for a long edge whose length has to be represented in more than 6 bits. In general, if $e(v_{ri}^{(k)}, v_{ri}^{(k+1)})$ denotes the edge from the vertex $v_{ri}^{(k)}$ to the next vertex $v_{ri}^{(k+1)}$, the latter being the destination vertex, then the two most significant bits are “01” if $\text{type}(v_{ri}^{(k+1)}) = 90^0$, or “11” if $\text{type}(v_{ri}^{(k+1)}) = 270^0$. Further, if $|e(v_{ri}^{(k)}, v_{ri}^{(k+1)})|$ denotes the edge length between $v_{ri}^{(k)}$ and $v_{ri}^{(k+1)}$, and $|e(v_{ri}^{(k)}, v_{ri}^{(k+1)})| > 2^6 - 1 = 63$ grid units, then the problem of bit overflow is resolved by making the two most significant bits to be “10” (indicating 180^0) and the overflowing bits are put to the next byte until the entire length of the edge is stored in the corresponding bit representation.

As an example, if $|e(v_{ri}^{(k)}, v_{ri}^{(k+1)})| = 184$ and $\text{type}(v_{ri}^{(k+1)}) = 270^0$, then the code for $e(v_{ri}^{(k)}, v_{ri}^{(k+1)})$ needs 16 bits; and the 16-bit code for this edge would be “11000010 10111000”.

The shape code for the polygon P_{ri} , having the ordered set of vertices $V_{ri} = \langle v_{ri}^{(1)}, v_{ri}^{(2)}, \dots, v_{ri}^{(m_{ri})} \rangle$, therefore, can be obtained as follows:

$$SC(P_{ri}, g) = x[v_{ri}^{(1)}]y[v_{ri}^{(1)}] (b_8b_7)_{ri}^{(2)} |e(v_{ri}^{(1)}, v_{ri}^{(2)})| (b_8b_7)_{ri}^{(3)} |e(v_{ri}^{(2)}, v_{ri}^{(3)})| \dots (b_8b_7)_{ri}^{(m_{ri})} |e(v_{ri}^{(m_{ri}-1)}, v_{ri}^{(m_{ri})})|, \tag{1}$$

where, $(x[v_{ri}^{(1)}], y[v_{ri}^{(1)}])$ denotes the start vertex, $(b_8b_7)_{ri}^{(2)}$ denotes the two (most significant) bits representing the vertex type of the second vertex $v_{ri}^{(2)}$, $|e(v_{ri}^{(1)}, v_{ri}^{(2)})|$ the length of the edge from the first vertex to the second vertex; and so on.

2.3 Multigrid Shape Code

The shape code of a tight isothetic polygon for an object (connected component) in a given uniform (square) grid $\mathcal{G} = (\mathcal{H}, \mathcal{V})$ depends not only on the shape of the object but also on the spacing of the grid lines. Higher the separation between the grid lines is, lower will be the visual perception about the underlying object from its isothetic polygon. On the contrary, as the grid lines become denser, the corresponding isothetic polygon becomes more meaningful and expressive, thereby facilitating the process of recognizing the underlying object.

In order to achieve a fast preliminary result on resemblance of object shapes, therefore, sparsely separated grid lines with large unit grid squares are used to generate the shape codes of objects. Shape codes for denser grid lines with smaller

unit grid squares, on the other hand, are required for finer and accurate checking between objects with similar shapes as certified from grid boxes with higher sizes. It may be observed in Fig. 1 how the underlying objects become more and more revealing as the grid size (separation of grid lines) goes on decreasing from 16 in Fig. 1(a) to 4 in Fig. 1(c).

The **Multigrid Shape Code (MuSC)** for the image \mathcal{I}_r is given by the concatenated ordered sequence of the shape codes (SC) of the objects (connected components) in the image, where a byte with zero value (i.e. each of its eight bits is 0) marks the end of the shape code of a single polygon, and two successive zero-valued bytes indicate the end of the shape codes of all the polygons for a particular grid configuration. Hence, MuSC for the image \mathcal{I}_r is given by:

$$MuSC(\mathcal{I}_r) = SC(\mathcal{I}_r, g_1) 0^{16} SC(\mathcal{I}_r, g_2) 0^{16} \dots 0^{16} SC(\mathcal{I}_r, g_\alpha), \tag{2}$$

where, α is the number of different grid separations, ranging from g_1 to g_α , that have been used to generate the above MuSC. The shape code $SC(\mathcal{I}_r, g)$ of the image \mathcal{I}_r for a grid size g , $g_1 \leq g \leq g_\alpha$, which has been used in the above equation to obtain $MuSC(\mathcal{I}_r)$, is as follows:

$$SC(\mathcal{I}_r, g) = SC(P_{r1}, g) 0^8 SC(P_{r2}, g) 0^8 \dots 0^8 SC(P_{rn_r}, g). \tag{3}$$

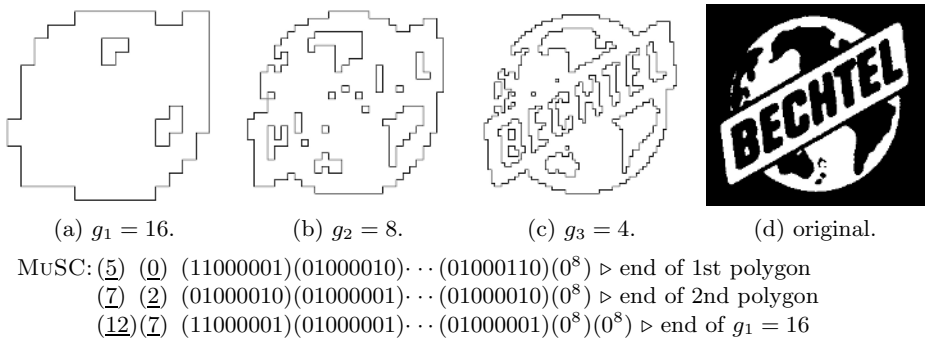


Fig. 1. A sample logo image and its tight isothetic polygonal envelopes representing the corresponding MuSC. Each byte (8 bits) has been shown in parentheses for clarity.

All images in \mathcal{D} are normalized to the size of $\mu \times \mu$ before finding their MuSCs. In our experiments, we have considered $\mu = 256$ pixels.

2.4 Image Retrieval Using MuSC

Let \mathcal{Q} be the query image. At first, the multigrid shape code $MuSC(\mathcal{Q})$ (for grid sizes g from g_1 to g_α) is obtained as described in Eqn. 2. It may be noted that, for any image \mathcal{I} , $MuSC(\mathcal{I})$ represents the highest level description of all the tight isothetic polygons corresponding to the object(s) present in the image \mathcal{I}

for increasing grid resolution from g_1 to g_α . Hence, we resort to the next (lower) level description of each isothetic polygon in the process of retrieval as follows.

For each database image \mathcal{I}_r , an ordered list, namely $L(\mathcal{I}_r, g_1)$, containing (pointers to) the vertical edges of all its isothetic polygons corresponding to the grid size g_1 , is prepared using $SC(P_r, g_1)$ (see Eqn. 1), which, in turn, is extracted from its multigrid shape code, $MuSC(\mathcal{I}_r)$ (Eqn. 2). The list $L(\mathcal{I}_r, g_1)$ contains $H_1 = \mu/g_1$ pointers corresponding to $h = 1$ to $h = H_1$, since μ is the height of the normalized image. The salient points about $L(\mathcal{I}_r, g_1)$ are as follows:

- the pointer $L(\mathcal{I}_r, g_1)[h]$ points to a linked list containing those edges of $SC(P_r, g_1)$, each of whose upper vertex has y -coordinate = h .
- each node in the linked list pointed from $L(\mathcal{I}_r, g_1)[h]$ denotes a vertical edge $e(u, v)$ (with $h = y[u] < y[v]$ and $x[u] = x[v]$) of $SC(P_r, g_1)$, and contains $x[u]$, $y[u]$, and $y[v]$ required in the subsequent steps of the retrieval process.
- in the linked list that $L(\mathcal{I}_r, g_1)[h]$ points to, the edges (nodes) are sorted in ascending order of their x -coordinates (of upper vertices).

A similar list, $L(\mathcal{Q}, g_1)$, is also prepared for the query image \mathcal{Q} to find out the similarity between the isothetic polygon(s) of \mathcal{Q} and those of \mathcal{I}_r for grid size g_1 . In order to do this, we construct a binary matrix $M(\mathcal{I}_r, g_1)$, having size $H_1 \times H_1$, considering $g = g_1$ and $H = H_1$ in Eqn. 4. A similar matrix, namely $M(\mathcal{Q}, g_1)$, is also being constructed for \mathcal{Q} using Eqn. 4, and these two matrices, $M(\mathcal{I}_r, g_1)$ and $M(\mathcal{Q}, g_1)$, are used to find the Hamming distance $L(M(\mathcal{I}_r, g_1), M(\mathcal{Q}, g_1))$ between the isothetic polygons of database image \mathcal{I}_r and those of the query image \mathcal{Q} corresponding to grid size g_1 , considering $g = g_1$ and $H = H_1$ in Eqn. 5.

$$M(\mathcal{I}_r, g)[w][h] = \begin{cases} 1, & \text{if } |\{e(u, v) \in L(\mathcal{I}_r, g) : y[u] \leq k < y[v] \text{ AND} \\ & x[u] \leq w\}| \text{ is odd;} \\ 0, & \text{otherwise; for } 1 \leq w, h \leq H. \end{cases} \quad (4)$$

$$L(M(\mathcal{I}_r, g), M(\mathcal{Q}, g)) = \sum_{w=1}^H \sum_{h=1}^H |M(\mathcal{I}_r, g)[w][h] - M(\mathcal{Q}, g)[w][h]| \quad (5)$$

As evident from Eqns. 4 and 5, if the Hamming distance $L(M(\mathcal{I}_r, g_1), M(\mathcal{Q}, g_1))$ is small, then the image \mathcal{I}_r is a probable candidate for successful retrieval corresponding to the query image \mathcal{Q} . Hence, the images with lower values of Hamming distance are considered one by one for the next grid size g_2 .

3 Results and Discussions

The proposed method is implemented in C on a Sun_Ultra 5_10, Sparc, 233 MHz, the OS being the SunOS Release 5.7 Generic on databases (i) D1 of 1034 logo images, (ii) D2 of 110 logo images. The results of the experiments done on the above two sets of images are shown in Table 1. It shows the amount of storage (in bytes as well as in percentage) required for the shape codes for different grid sizes and the average CPU time (per image) required to construct

Table 1. Results for databases D1 and D2

grid size	D1 (1034 images)			D2 (110 images)		
	Storage required		CPU Time	Storage required		CPU Time
	bytes	%	millisecs	bytes	%	millisecs
16	51363	0.60	22.10	456	0.05	25.17
8	126113	1.49	31.04	12338	1.37	34.79
4	298943	3.53	52.40	32038	3.55	64.85
2	677208	8.00	105.97	71532	7.93	123.32
1	1422700	16.80	227.70	151296	16.78	265.96

the shape code. The storage requirement is lower for coarse grids compared to finer grids, as in finer grid the shape is captured in details. The shape codes for grid sizes 16, 8 and 4 are sufficient to produce good retrieval results and the total storage required is approximately 15 percent. On the other hand, for visualization, the shape codes of grid size 2 or 1, depending upon the quality requirement, will be sufficient, consuming only 8 or 16 percent of storage respectively. Also, the CPU time requirement increases with the decreasing grid size. Table 2 shows how efficiently the effective search database size is pruned at each grid level.

Table 2. Average effective search database size for D1 and D2

α	D1			D2		
	16	8	4	16	8	4
L_t	64	256	1024	64	256	1024
D_e	184	88	24	10	4	1
L_t = Threshold Hamming Distance						
D_e = Effective Search Database						

4 Conclusion and Future Work

A novel technique for determining the multigrid shape code of a binary image is proposed here using the characteristic properties of an isothetic polygon. An efficient retrieval scheme is designed and implemented to demonstrate the power and versatility of such shape codes, irrespective of the size and diversity of a database.

There lie further scopes for improvement of the proposed method, namely, devising an improved method for finding the degree of similarity between two input sets of isothetic polygons, and achieving rotation invariance of the retrieval scheme. For a gray scale image, proper adaptation of this technique based on Euler vector [1], may possibly yield good results, but feasibility of this aspect needs further investigation.

References

1. A. Bishnu, B.B. Bhattacharya, M.K. Kundu, C.A. Murthy, T. Acharya, *Euler Vector for Search and Retrieval of Gray-Tone Images*, IEEE TSMC-B, vol. 27, pp. 801–812, 2005.
2. A. Biswas, P. Bhowmick, and B. B. Bhattacharya, *TIPS: On Finding a Tight Isothetic Polygonal Shape Covering a 2D Object*, LNCS, vol. 3540, Proc. SCIA'05, pp. 930–939.
3. - *ibid*, *Isothetic Polygons of a 2D Object on Generalized Grid*, Proc. PReMI'05, LNCS, Dec. 2005 (to appear).
4. M.-K. Hu, *Visual Pattern Recognition by Moment Invariants*, IRE Transactions on Information Theory, vol. 8, pp.179–187, 1962.
5. A. K. Katsaggelos et al., *MPEG-4 and Rate-Distortion-Based Shape-Coding Techniques*, Proc. IEEE, vol. 86, no. 6, pp. 1126–1154, 1998.
6. F. Mokhtarian and A. K. Mackworth, *A Theory of Multiscale, Curvature-Based Shape Representation for Planar Curves*, IEEE TPAMI, vol. 14, no. 8, pp. 789–805, 1992.
7. K. J. O' Conell, *Object-Adaptive Vertex-Based Shape Coding Method*, IEEE TCSVT, vol. 7, no. 1, pp. 251–255, 1997.
8. G. M. Shuster, et al. *An Optimal Polygonal Boundary Encoding Scheme in the Rate Distortion Sense*, IEEE TIP, vol. 7, no. 1, pp. 13–26, 1998.

Adaptation of Intelligent Characters to Changes of Game Environments

Byeong Heon Cho¹, Sung Hoon Jung², Kwang-Hyun Shim¹,
Yeong Rak Seong³, and Ha Ryoung Oh³

¹ Electronics and Telecommunication Research Institute,
161 Gajeong-dong, Yuseong-gu, Daejeon, 305-350, Korea
{bhcho, shimkh}@etri.re.kr

² Dept. of Information and Communication Engineering, Hansung University,
389 Samseon-dong, Seongbuk-gu, Seoul, 136-792, Korea
shjung@hansung.ac.kr

³ School of Electrical Engineering, Kookmin University,
861-1 Jeongnung-dong, Seongbuk-gu, Seoul, 136-702, Korea
{yeong, hroh}@kookmin.ac.kr

Abstract. This paper addresses how intelligent characters, having learning capability based on the neural network technology, automatically adapt to environmental changes in computer games. Our adaptation solution includes an autonomous adaptation scheme and a cooperative adaptation scheme. With the autonomous adaptation scheme, each intelligent character steadily assesses changes of its game environment while taking into consideration recently earned scores, and initiates a new learning process when a change is detected. Intelligent characters may confront various opponents in many computer games. When each intelligent character has fought with just part of the opponents, the cooperative adaptation scheme, based on a genetic algorithm, creates new intelligent characters by composing their partial knowledge of the existing intelligent characters. The experimental results show that intelligent characters can properly accommodate to the changes with the proposed schemes.

1 Introduction

The computer game industry resides as one of the explosive growth areas of recent years. According to [1, 2], it is now bigger than the film industry. A well-known drawback of conventional computer games is that gamers' interests can be seriously diminished due to simple, predictable behavior of computer-controlled characters. Recently, therefore, incorporating artificial intelligence (AI) into computer games has become a major research issue [1, 3, 4, 5].

Games are one of the oldest and most widely studied topics of AI. However, most efforts have concentrated on board games, e.g. tic-tac-toe [6], five-in-a-row [7], Othello [8], Chess [9], and Go [10]. In those games, the whole situation has to be considered whenever movements of stones are decided, and each stone

has no intelligence. As opposed to the aforementioned, in contemporary action games, each character's intelligence to take proper action considering opponents' positions and actions is very important.

In [11], we proposed methods to implement intelligence characters (ICs) for fighting action games by using neural network technology. The neural network autonomously learns game rules and decides IC's action according to the opponent's action. However, since learning capability is just part of human intelligence, it is insufficient for implementing human-like ICs. For example, consider an IC confronts with a new opponent character (OC) having different behaviors with prior ones. For an effective response to this, the IC must have adaptation capability, which re-initiates its learning process, rather than insisting the previously learned knowledge.

In this paper, we address how ICs automatically adapt themselves to those environmental changes. Our adaptation solution includes two components: an autonomous adaptation scheme and a cooperative adaptation scheme. The autonomous adaptation scheme is used to accommodate a single IC to varying circumstances. With the scheme, an IC steadily checks its game score, assesses the environmental change, taking into consideration the recently earned scores, and initiates a new learning process when a change is detected. To speed up the learning process, the usefulness of the IC's existing knowledge is evaluated and used to determine learning parameters.

The cooperative adaptation scheme is based on the evolution of a population of ICs. A group of ICs confronts with a group of OCs in many computer games. In this case, an IC may fight with just part of the OCs during its lifetime. Thus each IC has partial knowledge of OCs. The cooperative adaptation scheme based on the genetic algorithm creates new ICs by composing the partial knowledge of the existing ICs. To illustrate the performance of the proposed schemes, we employ the simple fighting action game in [11] and experiment on it with changing game rules and OCs' action patterns. The experimental results show that the proposed schemes are able to make ICs adapt themselves to the change.

2 Previous Work

In [11], an IC for fighting action games is implemented by using neural network technology. The IC autonomously learns game rules and decides its action according to an opponent's action. Fig. 1 shows the structure of the neural network. The neural network receives the OC's action type, the OC's action progress, and the distance between the IC and the OC. Additionally, the neural network receives the OC's past actions for adapting to the opponent's action pattern. Each output node corresponds to an action, which can be performed by the IC. Thus, the number of output nodes is equal to the number of IC's available actions. The IC selects the winning output node, which produces the largest value, and then performs the action related to the node. The neural network is a general feed forward network [12]. After input and output values are decided, it is learned by using a reinforcement learning algorithm.

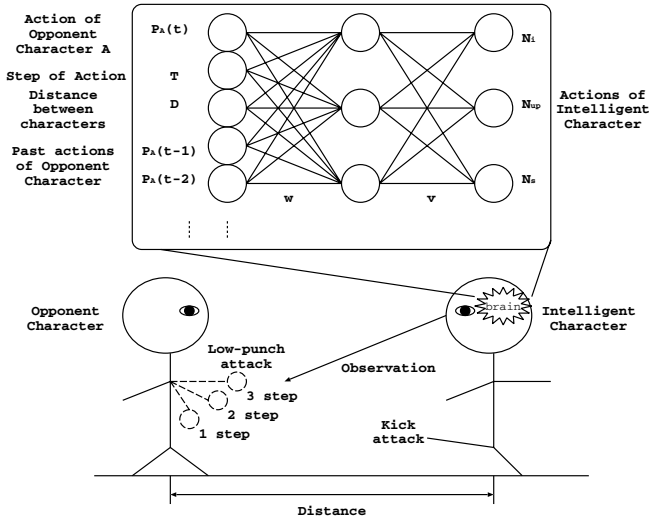


Fig. 1. The neural network structure

3 Autonomous Adaptation Scheme

Consider how a human being faces a new unknown enemy. Since he/she has no information about the enemy, he/she will try to confront the enemy by using his/her existing fighting skills. If the skills work, he/she will continue to use them. Otherwise, he/she should learn new skills for the opponent.

For simulating this process, an IC has to autonomously decide whether a new learning process is required or not. It can be determined by assessing the changes of score difference between the IC and the OC. Thus, if the change is greater than a threshold value, IC initiates a new learning process. Otherwise, IC sustains to use its existing skills.

A well-known drawback of neural networks is long learning time. When a neural network has no information about the underlying system, such a long learning time may be acceptable. However, if the neural network has some appropriate intelligence, it would be better to use the intelligence for reducing the learning time. In this paper, we infer that the score difference is involved with the usefulness of the IC's existing skills to fight with the present opponent. Thus, the difference is also used to determine the usefulness of the previous knowledge.

Fig. 2 shows the autonomous adaptation scheme. To implement the scheme, two variables, γ and T_l , are defined, where γ denotes the rejection rate with which an IC disbelieves its existing knowledge and attempts to try a random action, and T_l indicates the learning period. If γ is 1, an IC selects a random action, and if γ is 0, it acts with its existing skills. Initially, γ and T_l are set to 1 and 500,000, respectively. Two constants θ and T_a denote the threshold value of the relative deviation of the score difference, and the size of the observation

window in which the score difference is monitored. In this paper, they are set to 0.01 and 100, respectively. During the *Learning* step, an IC selects a random or a learned action by comparing a random value and γ . And it learns the result of the selected action. In the *Fighting* step, the IC behaves according to its learned knowledge but continually monitors the result. Thus, if Δ , the relative deviation of the score difference, is larger than θ , then the IC realizes an environmental change and re-initiates the learning process. At this time, γ and T_l are set relying on Δ . That is, the re-learning process is controlled by the degree of the change.

```

 $\gamma = 1.0;$  // rejection rate
 $T_l = 500000;$  // learning period

```

```

Learning: for  $i \leftarrow 1$  to  $T_l$  do
    if  $rand() > \gamma$  then
        act a random action;
    else
        act a learned action;
    learn the result;

```

```

Fighting:  $\delta_c = 0$ 
    for  $j \leftarrow 1$  to  $T_a$  do
        act a learned action;
         $\delta_c \leftarrow \delta_c + \text{IC's earned score} - \text{OC's earned score};$ 
         $\Delta \leftarrow |(\delta_c - \delta_p) / \delta_p|;$ 
         $\delta_p \leftarrow \delta_c;$ 
        if  $\Delta < \theta$  then
            goto Fighting;
        else
            set  $T_l$  and  $\gamma$  depending on  $\Delta$ ;
            goto Learning;

```

Fig. 2. The autonomous adaptation scheme

4 Cooperative Adaptation Scheme

Consider an engagement between a group of ICs and a group of OCs. With the autonomous adaptation scheme, if an IC needs to properly face with n types of OCs, it should initiate the re-learning process at least n times. This is a very time-consuming job. For this case, we propose a cooperative adaptation scheme. The scheme is used to accommodate a population of ICs to confront a group of enemies. The basic idea is to propagate superior fighting skills of each IC to others by means of evolution. For that, knowledge of each IC, i.e. weights of the associated neural network, is represented to a chromosome, a candidate for solutions of a problem, and then is inherited to the next generations of ICs by using a genetic algorithm.

For ease of exposition, we explain the scheme with an illustrative example in Fig. 3. Consider that (i) there is a group of ICs, (ii) enemies are composed of

three kinds of OCs (A, B, and C), and (iii) each IC has fought with only one kind of OCs. The goal is that every IC can properly confront with all three kinds of OCs as soon as possible. First of all, the fitness of each IC is evaluated. It is computed by means of the score difference resulting from IC's matches with every kind of OCs. This evaluation process takes a very short time in comparison with the autonomous adaptation process, since the ICs only evaluate their skills rather than searching an optimal action sequence. Then, parent ICs are selected in proportion to the fitness and generate their children by means of a crossover operation. Likewise, a mutation operation is applied to the children for avoiding a local optimum. These two operations will be explained next. By iteratively applying the above procedure, the next generation is produced. Also, through alternation of generations, the number of ICs that learn the optimal counter actions for all the OCs' action patterns increases.

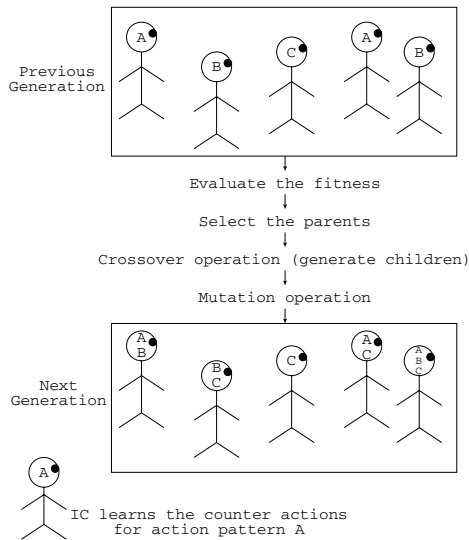


Fig. 3. Cooperative adaptation scheme

To implement the above scheme, the structure of chromosome and crossover and mutation operations should be defined. As mentioned earlier, since a neural network implements an IC, IC's knowledge is represented by the weights of the network. Thus, in this paper, the chromosome is constructed as a one-dimensional array containing the weight information. A crossover operation exchanges information of selected chromosomes and generates new chromosomes. We employ the well-known general n -point crossover scheme [13]. A mutation operation randomly modifies a selected part of a chromosome. It makes attributes absent in parents and extends search areas. In this paper, random numbers are added to genes of a chromosome with a mutation probability.

5 Performance Evaluation

In this section, we evaluate the performance of the proposed adaptation schemes. We change IC's game environment in two ways: (i) by changing OC's action patterns, and (ii) by changing game rules.

For these experiments, the fighting action game implemented in [11] is employed. In each match, two confronting characters move, attack, defend and obtain scores. A common clock synchronously triggers each action. Table 1 shows the available actions and their effective attack range, score and required time. Likewise, as real games, we presume the following three rules. First, if a character is attacked while doing an action, the action is stopped and invalidated. Second, if a character starts an action, it cannot be canceled or changed to another action. Third, an attack action would be blocked or weakened by defense actions as follows: (i) *low-punch* and *low-kick* attacks obtain no score if the opponent takes a *guard* or *jump* action, (ii) *high-punch* and *high-kick* attacks obtain no score if the opponent takes a *guard* or *down* action, and (iii) a *special* attack obtains a half of the attack score if the opponent takes a *guard* action. For simplicity, the underlined letters in Table 1 abbreviate each action from now on. For example, *lp* denotes *low-punch*.

Table 1. Available actions

Action	<u>i</u> idle, <u>f</u> orward, <u>b</u> ackward, <u>j</u> ump, <u>d</u> own, <u>g</u> uard	<u>l</u> ow- <u>p</u> unch	<u>h</u> igh- <u>p</u> unch	<u>l</u> ow- <u>k</u> ick	<u>h</u> igh- <u>k</u> ick	<u>s</u> pecial
Required time (clocks)	1	2	4	6	8	10
Attack score	-	1	2	3	4	5
Effective attack range	-	0-2		2-3		3-5

5.1 Evaluation of the Autonomous Adaptation Scheme

In the first experiment, we evaluated the performance of our autonomous adaptation scheme. To show the adaptation capability, three kinds of action patterns are employed. Table 2 shows the action patterns. Initially, an IC is trained with one of them. For each case, we use 20 random initial seeds and measure the average of the score ratio between competing characters.

First, the adaptation performance is evaluated by changing opponents' action patterns. Fig. 4 shows the results. In Fig. 4(a) and (b), the OCs initially act with

Table 2. Used action patterns and corresponding optimal counter-actions

Pattern	Action sequence
A	f, hp, b, lk, b, s, f, hk
B	f, hp, b, lk, f, lp, b, hk
C	b, s, f, lk, b, s, f, hk

Pattern A. Then, their action patterns are suddenly changed to Pattern B and C, respectively. For comparison, the learning result is also illustrated when the IC has not initially trained. In Fig. 4(a), when the IC has no initial knowledge, the average score ratio is very low until $t = 3,200,000$. However, when the IC has learned, although the opponent's changed action pattern differs with the previous one, the early score ratio is considerably high. Therefore, we can conclude that the IC can quickly learn changed action patterns by using the autonomous adaptation scheme. As opposed to the above case, in Fig. 4(b), the initial knowledge has relatively small effect to the result. This is because Pattern A and B are relatively similar, but Pattern C is quite different from others.

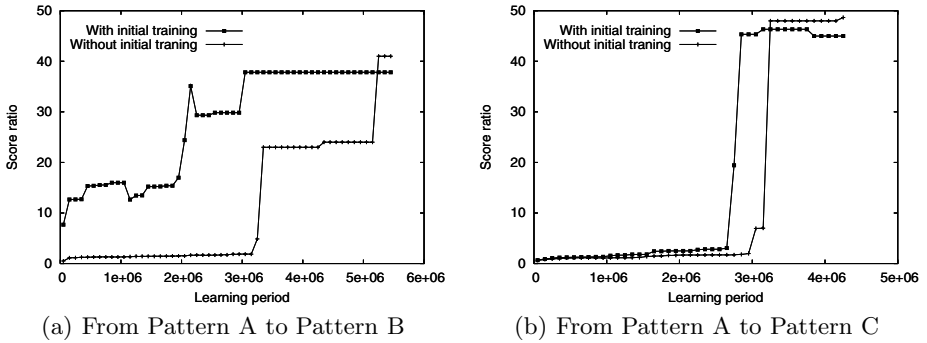


Fig. 4. Adaptation to changes of action patterns

Second, the adaptation performance is evaluated by changing game rules. Similarly to the previous experiment, we use 20 random initial seeds and investigate the average of the score ratio. Initially, each IC has trained with OCs, which acts with one of the action patterns in Table 2 by using Rule I in Table 3. Then, the game rule is changed to Rule II in Table 3, and the average score ratio is measured.

Table 3. Game rules

Action	Attack score				
	<i>lp</i>	<i>hp</i>	<i>lk</i>	<i>hk</i>	<i>s</i>
Rule I	1	2	3	4	5
Rule II	5	4	3	2	1

Fig. 5 shows the result. For comparison, the learning result is also illustrated when the IC has not initially trained as Fig. 4. In Fig. 5(a) and (c), when the IC has not initially trained, the average score ratio is very low during the early learning stage. However, when the IC has trained, although the present game rule differs greatly from the present one the early score ratios are very high. This is because the two game rules are identical in the sense of other aspects, i.e. attack range and elapse time, although they are quite different in sense of

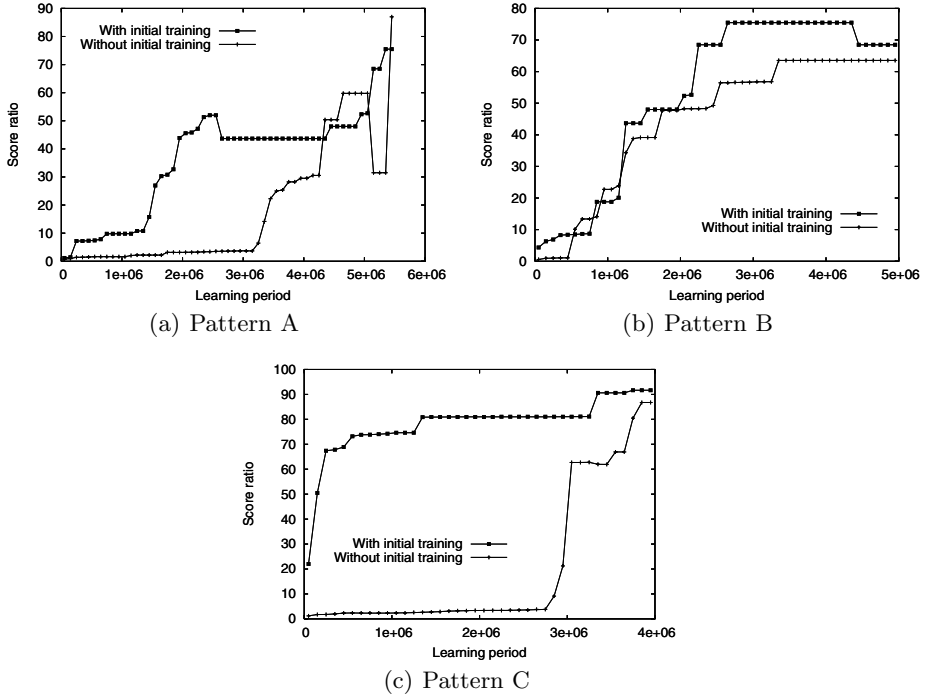


Fig. 5. Adaptation to changes of game rules

attack scores. Thus, IC's previous knowledge can be used to learn the changed rule. As opposed to these cases, in Fig. 5(b), the initial knowledge does not have influence on the results.

5.2 Evaluation of the Cooperative Adaptation Scheme

In the second experiment, we tested the performance of the cooperative adaptation scheme. The number of individuals is 30, the crossover probability is 0.7, and the mutation probability is 0.1. At every generation, the IC having the maximum fitness value is forced to be inherited to the next generation. We use 20 random initial seeds and show the average of the maximum fitness of each generation.

The performance is evaluated by changing opponents' action patterns. Before adaptation, ICs are partitioned into three groups. Each group confronts with an OC, which acts with only one of the three action patterns in Table 2, and there is no interaction between the groups. Then, the three groups are merged into one and evolved to adapt to the changed environment from the 0-th generation. Fig. 6(a) shows the result. As mentioned earlier, the fitness is computed by means of the score difference resulting from IC's matches with every kind of OCs. Before the 0-th generation, each IC had specialized knowledge of only one particular OC, but not all OCs. Thus, the fitness was not very high. The result shows that, although the fitness severely drops at the 0-th generation, it quickly

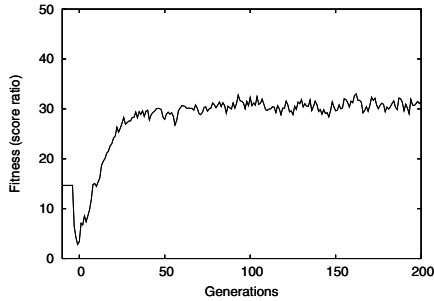


Fig. 6. Results of the cooperative adaptation

increases several tens of generations later. From this result, we can conclude that the cooperative adaptation scheme efficiently creates new ICs by collecting existing knowledge of ICs.

6 Conclusions

In this paper, we proposed two adaptation schemes to adapt ICs having learning capability using neural network technology to environmental changes. First, with the autonomous adaptation scheme, an IC steadily checks its game score, assesses the environmental change with taking into consideration of the recently earned scores, and initiates a new learning process when a change is detected. The usefulness of IC's previous knowledge is determined by means of the recently earned scores and is reflected to the learning period and the rejection rate of the existing knowledge. Second, the cooperative adaptation mechanism based on the genetic scheme creates new ICs by combining knowledge of the existing ICs. For this evolution, the link weights of the neural network of IC are represented to a chromosome, and crossover and mutation operators are defined. From the experimental results, we can conclude that the ICs can continually adapt to the game environment with the proposed adaptation schemes. Currently, we have been extending our study to other games including team sports games (e.g. soccer, baseball, etc).

Acknowledgements

This work was supported in part by research program 2005 of Kookmin University in Korea.

References

1. Laird, J.E., van Lent, M.: Human-level AI's killer application: Interactive computer games. *AI Magazine* **22** (2001) 15–26
2. Forbus, K.D., Laird, J.E.: Guest editors' introduction: AI and the entertainment industry. *IEEE Intelligent Systems* **17** (2002) 15–16

3. Fairclough, C., Fagan, M., Namee, B.M., Lippmann, R.P.: Research directions for AI in computer games. Technical report, Computer Science Department, Trinity College Dublin (2001)
4. Forbus, K.D., Mahoney, J.V., Dill, K.: How qualitative spatial reasoning can improve strategy game AIs. *IEEE Intelligent Systems* **17** (2002) 25–30
5. Bauckhage, C., Thureau, C., Sagerer, G.: Learning human-like opponent behavior for interactive computer games. *Pattern Recognition, Lecture Notes in Computer Science* **2781** (2003) 148–155
6. Fogel, D.: Using evolutionary programming to create neural networks that are capable of playing tic-tac-toe. In: *IEEE International Conference on Neural Networks*. (1993) 875–880
7. Freisleben, B.: A neural network that learns to play five-in-a-row. In: *2nd New Zealand Two-Stream International Conference on Artificial Neural Networks and Expert Systems*. (1995) 87–90
8. Moriarty, D.E., Miikkulainen, R.: Discovering complex othello strategies through evolutionary neural networks. *Connection Science* **7** (1995) 195–210
9. Fogel, D.B., Hays, T.J., Hahn, S.L., Quon, J.: A self-learning evolutionary chess program. *Proceedings of IEEE* **92** (2004) 1947–1954
10. Richards, N., Moriarty, D.E., Miikkulainen, R.: Evolving neural networks to play go. *Applied Intelligence* **8** (1998) 85–96
11. Cho, B.H., Jung, S.H., Seong, Y.R., Oh, H.R.: Exploiting intelligence in fighting action games using neural networks. (Submitted to *IEICE Transactions on Information and Systems*)
12. Lippmann, R.P.: An introduction to computing with neural nets. *IEEE ASSP Magazine* **4** (1987) 4–22
13. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Professional (1989)

An Knowledge Model for Self-regenerative Service Activations Adaptation Across Standards

Mengjie Yu, David Llewellyn Jones, and A. Taleb-Bendiab

School of Computing and Mathematical Science, Liverpool John Moores University,
Byrom St. Liverpool, L3 3AF, UK
{cmsmyu, d.llewellyn-jones, a.talebbendiab}@livjm.ac.uk

Abstract. One of the greatest challenges for dependable service-oriented software systems of next generation is coping with the complexity of required adaptation or reaction to the detected unforeseen vulnerability attacks. To this end, autonomic system[1] has been advocated as a way to design self-protective systems to defend against malicious attacks or cascading failures. However, other initiatives such as the self-regenerative system[2] adopt the biological-inspired [2, 3] notions such as natural diversity and self-immune as a main strategy to achieve the robust and adaptable self-protection. Based on an ongoing research into self-regenerative programming model, this paper presents a knowledge-centric approach for supporting the runtime automated generation of software adapters for cross-standard service activation; and argues the importance of application of a semantic knowledge to extract the notion of self-regenerative adaptation from the previous polyarchical middleware implementation. The benefit of this will be the production of a customizable self-regenerative adaptation service; and also, support for abstraction integration between domain of similar interests or others in a high-level management directed towards building autonomic systems in a large domain of interest.

1 Introduction

“Self” systems, inspired by the concept of autonomic computing introduced by IBM[1], has fuelled a growing interest in the construction of next generation distributed systems, which can efficiently self-manage themselves during the runtime processes without significant users’ interference. Self-healing, as a specific system reconfiguration of autonomic computing, denotes distributed systems abilities to observe failures resulting from faulty software and hardware, and consequently apply appropriate corrections to recover themselves from those unexpected failure. Most of these marvelous high-level decisions or behaviours require huge amounts of complex interaction and integration at the service component level. Moreover, research challenges raised from next generation software development architectures have also desires self-healing systems to build within service-oriented grid computing[3] environment. Components of distributed systems may deploy or implement with various distinct service standards and middleware technologies. Thus, robust self-adaptation becomes a vital capability for systems to recover themselves from failures by adopting any available grid resource. In these days, numbers of research projects have already addressed this

complex service activations adaptation problem. Much insight into use of common interoperation models has been adopted to accommodate adaptation among the diversity of grid components including: OPENWING[4], Model Oriented Architecture[5], and ICENI[6]. These approaches enable systems to interact with resources independently of direct knowledge of one another through a shared common interaction framework or a communication protocol layer. This work has shifted the burden of making interaction and adaptation consistent onto given technologies (e.g. static APIs) and data format (e.g. neutral meta-data such as Web service Description Language-WSDL). In order to apply these standard technologies, each component has to modify their original implementation if they intend to be integrated with or accessed (e.g. discovery and invocation) by other systems or components implemented with different mechanisms. Such approach is viable if suitable configuration and implementation has taken place during design time ahead of system deployment. However, rapidly changing technologies and unpredictable execution environments motivates a desire for more flexible runtime solutions allowing service adaptation across standards. In which case, systems don't need to halt processes for reconfiguring their adaptation setting in order to adopt available grid services bound with different approaches. Such runtime aspect is especially vital for self-immune systems to correct their security vulnerabilities whenever an unforeseen security attack occurs. As a result, numerous diverse versions of security systems can be generated by adopting various security components (such as access controls, encryption data formats, firewalls). Consequently, this approach avoids the generation of new vulnerability that might result from critical security components being halted during reconfiguration.

To solve the problems outlined above, this paper focuses on an alternate research solution: a runtime self-regenerative adaptation model based on a developed polyarchical middleware approach. A detailed description of this proposed polyarchical middleware and its self-regenerative model have been discussed in previous work[7] on supporting multi-standard service interoperation protocols and the adaptation abstract programming model[8]. However, this paper primarily focuses on the application of a knowledge model to extract the notions of self-regenerative adaptation activities from adaptation implementation. This knowledge can extend the polyarchical middleware to support software factory aspects in order to produce various adaptation services tailored to service consumers' own needs. It also supports the sharing of understanding of the self-regenerative adaptation between domains of same interests or others in a high-level management and configuration.

The rest of this paper is organized as follows: it begin with a short overview of the developed self-regenerative middleware service. It is followed by a discussion covering the reasons for use of a knowledge model. This is illustrated using a ongoing design and implementation of the knowledge model. Finally, the paper concludes with a discussion of the future development opportunities it affords.

2 Overview of Self-regenerative Adaptation Approach

To solve the problems outlined above, this paper will focus on a novel solution: a runtime self-regenerative model for service activation adaptation, based on a developed polyarchical middleware approach[7] that have been developed and implemented in Java. Considering that autonomic systems can spontaneously interact

with destination services because they all share the same interaction framework and communication layer, it can be supposed that in certain instances the only suitable service component arriving into the environment will be unexpectedly bound with different implementations. In this case, it is desirable that autonomic systems should still self-configure themselves by adopting the new component. As components have no prior knowledge of one another, a methodology is required to enable these discrete components to communicate with each other. The authors introduce a novel middleware framework that supports autonomic systems by dynamically generating the appropriate adaptation code (i.e. adapters) for the given or available services of choice. With such adapters, systems can discover and interact with remote components without revising individual components to implement special APIs at design time. The authors believe such an approach can well support autonomic systems to continuously maintain and adjust their operation through a high level of self-management while still achieving an alternative choice to interact with various components in unpredictable changing environments.

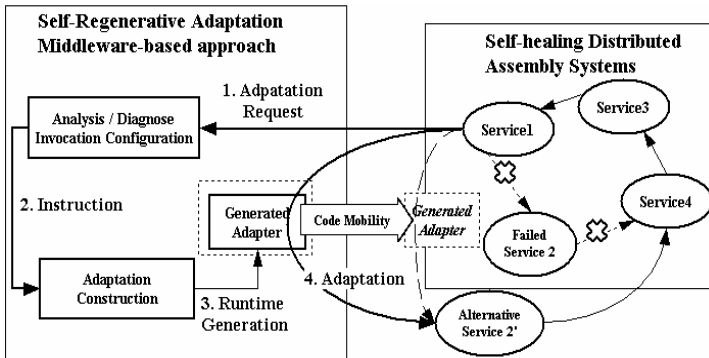


Fig. 1. Self-regenerative Service Adaptation Model

As shown in Figure 1 taken from [7], we considered a distributed service assembly (application), and at runtime we assumed that *service 2* fails. Suppose *service 2'* is the only available service matching the requested service (*service 2*). However, it implements a different interface (invocation mechanism) not supported by *service 1*. Thus, *adapter* between *service 1* and *service 2'* needs to be auto-generated and deployed. This will be achieved and supported by our polyarchical middleware architecture. In order for this to be achieved, a generic structural model [8] is designed to support our proposed self-regenerative adaptation service to accommodate dynamic loading and the binding of required middleware components. Accordingly the diversity of various service standards and middleware technologies can be achieved through the binding of those on-demand adapters.

3 Knowledge for Self-regenerative Adaptation

The previous section briefs the developed self-regenerative approach for supporting service adaptation across standards. The use of abstraction models [8] provide the

polyarchical middleware a generic structure to accommodate the diversity of service standards and middleware technologies adopted for the generation of on-demand adapters. To extend this further, our interest is in developing a semantic knowledge that can extract details of adaptation generation from the polyarchical middleware implementation. This knowledge can be used to support the design of *Service Adaptation Manager* in form of software factory.

3.1 Self-regenerative Adaptation Knowledge

Knowledge development can usually be viewed from two perspectives: 1) “Knowledge=Object” which specifies the semantic interpretation and implicit meanings of various terms and concepts across domains; 2) “Knowledge=Process” which primarily focuses on the service processes composition or software productions improvements. Here, the self-regenerative adaptation knowledge has been designed to describe processes involved with adaptation generation.

3.2 Reasons for Applying Adaptation Knowledge

In brief, the reasons for applying knowledge to support the self-regenerative adaptation service can be summarized as the following terms:-

Separation Domain Knowledge from Implementation is one of the more common goals in developing knowledge. In this case, details of static adaptation procedures will be extracted from the implementation and figured into this domain knowledge. Thus, *Service Adaptation Manager* can be mostly considered primarily as a configuration environment. It will follow process composition models provided by this domain knowledge to produce various adaptation services.

Reuse of Domain Knowledge is another driving force behind this approach. Domain with the same interest can simply adopt this knowledge by applying specific implementations. Additionally, it considers the building of a large domain knowledge, this existing knowledge can also be integrated and share portions of the large domain description. For example, considering more specific notions or concepts, this knowledge can be extended to integrate with domain interests of self-immune systems (like project PUCSec[9]). In this case after the dynamic evaluation of a system’s security through composition analysis, the security level may be automatically increased through the adoption of components providing various security algorithms or implementations. This may occur as a result of dynamic changes to the system, or whenever a security attack to the system is detected. The reuse of domain knowledge in this way helps to improve the effectiveness and robustness of the overall system.

Process Guidance and Resource Specification can support the design of *Service Adaptation Manager* into a software factory model, which considers the production of adaptation service families rather than a single approach. This software factory is guided by this knowledge to assemble the on-demand middleware resources for facilitating the adapters construction using the given process models either using default or customized settings.

Knowledge Integration plays an important role in the support of high-level information management or integration with domain of the same interests or others. The usages of this term can be viewed from two perspectives: 1) Enables service consumers

in other domains to integrate this self-regenerative adaptation service into their system processes. This knowledge can provide sufficient information about “what the self-regenerative adaptation does” and “how it works”. Thus, service consumers can customize adaptation services tailored to their own needs by defining individual adaptation process models. For example, this knowledge can be used in domains with self-healing interests to support distributed systems recovering from the component failures; 2) Improve adaptation knowledge by integrating with domains of same interests. In consideration of a large domain, various adaptation domains can extend their knowledge or share adaptation functions by integrating this high-level representation with each other. This can greatly leverage the existing adaptation resources in considering the large investment on the adaptation applications development.

3.3 Structure of Adaptation Knowledge

To complete the efforts mentioned in previous section, this adaptation knowledge has been process has been designed into a two-layer model as shown in the Fig2: 1) adaptation ontology; 2) middleware components description. Ontology[10], the term borrowed from philosophy, has widely been adopted to describe domain interests in various fields, and also for heterogeneous information integration[11]. The adaptation ontology here describes the self-regenerative adaptation service in term of a process model of the form “what adaptation functions it offers” and “how they work”. This information acts as a task list for software factory- service adaptation manager- on how to produce adaptation generation service, and also provides verification and justification for the adaptation processes. In order to do this, this ontology can further be divided into two layers: adaptation concepts and adaptation process models. The concept layer provides a high-level description for individual adaptation functionalities (e.g. processes) supported by polyarchical middleware. The details of the concept layer consist of various information including: processes description, control structure and data flow structure of processes in terms of the inputs, outputs, preconditions and their sub-processes if applies. The process model layer provides the details of how these processes can actually be composed to facilitate the generation of on-demand adapters. The separation of the adaptation ontology into two sub-layers leaves the configuration of adaptation process models more customizable. As a result, adaptation service consumer can compose process models tailored to their own needs based on the information from the concepts layer if they don’t want to stick with default process models provided by the polyarchical middleware. In addition, information from this adaptation ontology can also support the knowledge integration with domains of same interests or others in a high-level management or configuration.

The middleware components description declares the mappings from middleware components with specific implementation to processes declared in the adaptation ontology section. It also describes the accessing details for each middleware component into the technical terms including: protocol, message formats, serialization and addressing etc. In a quick summary, the software factory of service adaptation manager can get guidance from the adaptation ontology on “what to do” and “how to do”; furthermore, the component description provides information on “what components can do the job” and “how to access them”. In addition, the self-regenerative adaptation service can flexibly swap to another implementation approach by replacing component description and those components bound with specific standards and tech-

nologies. At the same time, the high-level abstraction of self-regenerative adaptation and the software factory structure can still remain the same without being infected.

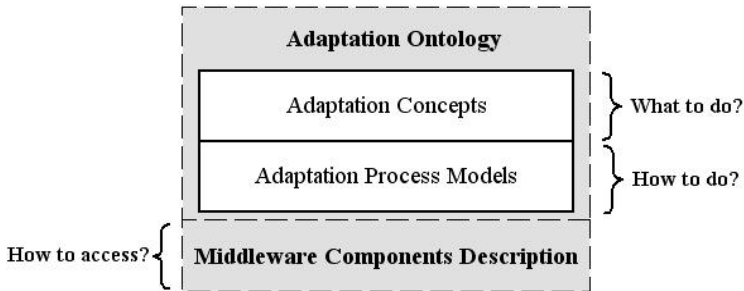


Fig. 2. Adaptation Knowledge Structure

4 Implementation of Self-regenerative Adaptation Knowledge

The three-layer adaptation knowledge has been implemented into separated meta-data documents (in format of XML). These machine-interpretable texts can flexibly integrate with the design of service adaptation manager on processes composition for providing self-regenerative adaptation service tailored to consumers' needs.

4.1 Adaptation Ontology Representation

As there is no "Correct" way or methodology for ontology development, this work is not trying to define a new ontology description language to address all the issues that an ontology development may need to grapple with. In order to quick test the research idea mentioned in this paper, it adopts a self-defined schema to describe adaptation process models instead of using standard technologies as DAML-S [12] or OWL-S.

Abstract process concepts defined in the adaptation ontology are mapped to the adaptation functionalities provided in the polyarchical middleware. This ontology describes the self-regenerative adaptation process as three developed integrated processes such as service discovery, service match making, and adapter construction. More specifically, a service lookup process consists of two separated sub-processes, namely both local and remote registry discovery. Also, an appropriate service match may consider several aspects such as the service name, and its programmatic signature (service signature). For the service parameters, it is also important to ascertain the matching of the number of parameters, their sequence and their data types are also important to ascertain. Based on this information, service consumers would then know the general characteristics for the self-regenerative adaptation service provided by the polyarchical middleware. They can then compose their own adaptation process models if they don't want to stick with the default one (offered by the polyarchical middleware). Indicated by the customized process models, the software factory of service adaptation manager

can produce an adaptation service tailored to the consumers' special requirements. The details of the specific middleware components bound with those abstract processes will be covered in the middleware component description file.

4.2 Middleware Component Description

As we known, WSDL (Web service description language) has been widely adopted to specify the technical details of accessing service components, which have mainly to do with protocol type, message formats, serialization, and transport etc. Currently, the syntax of WSDL has only been developed to bind with Web service relevant technologies. The middleware component description here has to adopt a self-defined XML schema to specify information raised by various other service invocation protocols besides Web services. With the similar goal as WSDL does, this middleware component description provides the details of accessing those middleware components bound with specific implementation. It also provides the mapping from (abstract) processes defined in the adaptation ontology representation to those middleware components. In future, this work can be easily swapped with any particular specification language if it is designed to describe information about various service bindings with strong industry backing.

5 Conclusions and Future Work

This paper first briefs a description of an alternate novel adaptation approach via runtime self-regenerative adapters provided by the polyarchical middleware. The rest of the paper highlights the motivations and requirements for the application of a knowledge-centric model on supporting and reasoning the runtime generation of self-regenerative adaptation. Future evaluation of this work is underway including a number of further developments including: the design of an evaluation tested for this ontology model based on the developed polyarchical middleware; supporting the semantic interpretation on the adaptation field across domains of same interests; another ontology model to describe the generation of adaptation template class, which might be transformed to deployable code (template class) encapsulated with specific information of service bindings and technologies in case the polyarchical middleware temporarily hasn't got one.

References

1. J. Kephart, D. Chess.: The Vision of Autonomic Computing, in Computer Magazine, IEEE, (2003).
2. Badger, Lee.: Self-Regenerative System (SRS) Program Abstract. DARPA, (2004).
3. Liang-Jie Zhang, Jen-Yao Chung, Researcher, IBM T.J. Watson Research Centre.: Developing Grid Computing Application: Part1, Introduction of a Grid architecture and toolkit for building Grid solutions, (2003).
4. Wassenberg, Wade, Software Engineer, and Motorola ISD.: Protocol Independent Programming Using Openwings Connector Services, (2003).

5. Heuvel, Willem-Jan van den. Matching and Adaptation: Core Techniques for MDA-(ADM)-driven Integration of new Business Applications with Wrapped Legacy Systems.
6. Furmento, Nathalie, Jeffrey Hau, William Lee, Steven Newhouse, and John Darlington.: Implementations of a Service-Oriented Architecture on top of Jini, JXTA and OGSA, In Proceedings of UK e-Science All Hands Meeting, (2003).
7. M.Yu, A. Taleb-Bendiab, D. Reilly.: A Polyarchical Middleware for Self-Regenerative Invocation of Multi-Standard Ubiquitous Services, In Proceedings of the IEEE International Conference on Web Services, (2004).
8. M.Yu, A. Taleb-Bendiab.: Generic Programming for Self-Regenerative Invocation cross-Standards, In Proceedings of 5th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking & Broadcasting, (2004).
9. David Llewellyn-Jones, Madjid Merabti, Qi Shi, Bob Askwith.: A Security Framework for Executables in a Ubiquitous Computing Environment, In Proceedings of Globecom, (2004).
10. Noy, Natalya F. and Deborah L. McGuinne.: Ontology Development 101: A Guide to Creating Your First Ontology, (2001).
11. Andreas Maier, Hans-Peter Schnurr, York Sure.: Ontology-based Information Integration in the Automotive Industry, In Proceedings of the 2nd International Semantic Web Conference, (2003).
12. Anupriya Ankolekar, Mark Burstein, Jerry R. Hobbs, Ora Lassila, David Martin, DrewMcDermott, Sheila A. McIlraith, and Massimo Paolucci Srini Narayanan, Terry Payne, and Katia Sycara.: DAML-S: Web Service Description for the Semantic Web.

An Agent for the HCARD Model in the Distributed Environment

Bobby D. Gerardo, Jae-Wan Lee, Jae-jeong Hwang, and Jung-Eun Kim

Kunsan National University,
68 Miryong-dong, Kunsan, Chonbuk 573-701,
South Korea
{bgerardo, jwlee, hwang, jekim}@kunsan.ac.kr

Abstract. In this study, we will employ a multi-agent for searching and extraction of data. We will use Integrator Agent based on CORBA architecture for the proposed model on hierarchical Clustering and Association Rule Discovery (HCARD). The model will address the inadequacy of other data mining tools in processing performance and efficiency when use for knowledge discovery. The result revealed faster searching using the agents. Our experiment also shows that the HCARD generated isolated but imperative association rules which in return could be practically explained for decision making purposes. Shorter processing time had been noted in computing for smaller clusters implying ideal processing period than dealing with the entire dataset.

1 Introduction

The problem of finding information on the Web or distributed networks is well known and the Web is the largest and most used distributed hypermedia systems known to date. The information systems that are distributed in nature continue to become available on the Web would complicate the problem even more [1]. As a result, attempting to locate, integrate and organize related information has become a major challenge.

On another perspective, data mining had been viewed or identified as valuable and essential steps towards knowledge discovery because it could be applied to extraction or mining of knowledge from vast amount of data. Some constraints that most researchers observed in the data mining tasks were computing speed, reliability of the approach for computation, heterogeneity of database, and vast amount of data to compute [2], [3]. These are restraints that defeat typical and popular data mining tools.

In this study, we investigate the formulation of the cluster analysis technique as integrated component of the proposed HCARD model to partition the original data prior to implementation of data mining tools. The model that we proposed uses the hierarchical nearest neighbor method and apriori algorithm for knowledge discovery purposes to be used on business or transactional databases. In addition, we will develop a multi-agent system based on CORBA architecture for data searching and extraction in the distributed environment. Two agents, such as the user interface and the facilitator agents will be developed for interface, searching, extraction, and management.

2 Related Studies

2.1 The CORBA Architecture

The Common Object Request Broker Architecture (CORBA) was developed by an industry consortium known as the Object Management Group (OMG), an architecture that enables pieces of programs, called objects, to communicate with one another regardless of what programming language they were written in or what operating system they are running on [4], [5].

Because of the easy way that CORBA integrates machines from many vendors, with sizes ranging from mainframes through hand-held and embedded systems, it is the middleware of choice for large enterprises. One of its most important, as well most frequent, uses is in servers that must handle large number of clients, at high hit rates, with high reliability [5]. Due to CORBA's popularity, it has several implementations. The most widely used are being IBM's SOM (System Object Model) and DSOM (Distributed System Object Model) architectures. CORBA has also been adopted by Netscape as part of its ONE (Open Network Environment) platform. Two competing models are Microsoft's COM (Common Object Model) and DCOM (Distributed Common Object Model) and Sun Microsystems' RMI (Remote Method Invocation).

Similarly, the multi-agent system can be used for the construction of open, distributed, and flexible architectures, capable of offering services for collective works. A multi-agent system is composed of a group of agents that are autonomous or semiautonomous which interact or work together, to perform some tasks or goals. In addition, the agents in such systems may either be homogeneous or heterogeneous and they may have common goals that are distinct with each other [1].

2.2 Cluster Analysis

The goal of cluster analysis is categorization of attributes like consumer products, objects or events into groups, so that the degree of correlation is strong between members of the same cluster and weak between members of different clusters [6]. It may show structure and associations in data, although not previously evident, but are sensible and useful once discovered [2]. Each group describes the class in terms of the data collected to which its members belong.

2.3 Data Mining

Several data mining algorithms have been introduced that can perform summarization, classification, deviation detection, data characterization and interpretation. Also, there are varieties of data mining algorithms that have been recently developed to facilitate the processing and interpretation of large databases [7], [8]. One example is the association rule algorithm, which discovers correlations between items in databases [9]. Apriori algorithm is used to find candidate patterns and those candidates that receive sufficient support from the database are considered for transformation into a rule [2].

3 Architecture of the Proposed System

The proposed architecture of the HCARD model is illustrated in Figure 1. This shows the implementation of the cluster analysis using the nearest neighbor clustering and association rule discovery method. The architecture of the interface, facilitator, and user interface agents are shown in Figure 2.

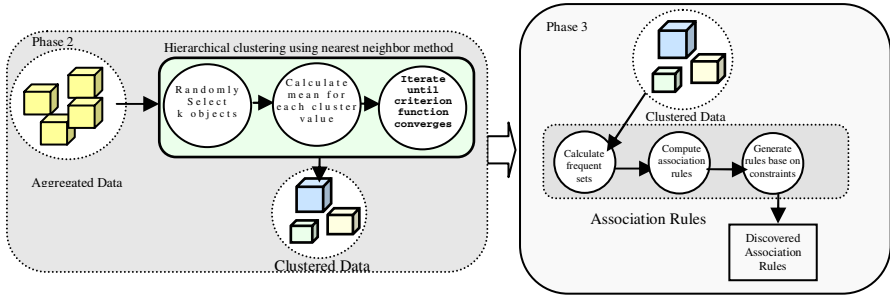


Fig 1. Refinements of the HCARD at Phase 2 and 3. Phase 2 shows the implementation of the hierarchical clustering algorithm while Phase 3 is the implementation association rule mining.

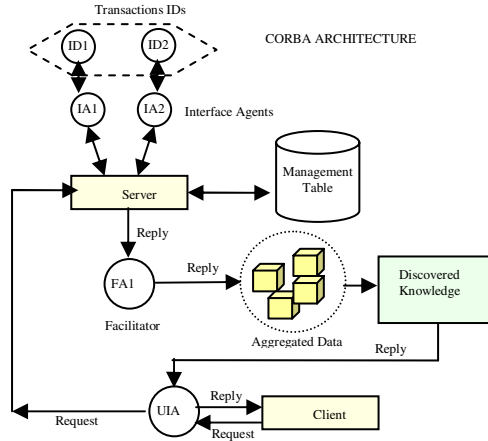


Fig. 2. Multi-agent system architecture. The interface (IA), facilitator (FA), and user interface agents (UIA) are illustrated in the figure and shows how they communicate to each other.

The Facilitator Agent (FA) will attempt to integrate some of the information that can assist users in locating information they need either within their local information systems or on the Web. In this study, the Facilitator Agent also refers to Integrator Agent (IA). It supports routing of information, discovery of information, delayed and persistent notification of messages and communication management. In addition, we will develop User Interface Agents (UIA) to provide the user with an interface whereby he/she can make use of the data or services available in the system through transparent communication with agents.

4 The Clustering Algorithm and the HCARD Model

4.1 Nearest-Neighbor Clustering

One of the simplest agglomerative hierarchical clustering methods is single linkage which is also known as the nearest neighbor technique. The defining feature of the method is that distance between groups is defined as the distance between the closest pair of objects, where only pairs consisting of one object from each group are considered. If we have two points given by, $X = (x_1, x_2, x_3, \dots, x_n)$ and $Y = (y_1, y_2, y_3, \dots, y_n)$, then the distance between the two points can be computed using equation 1 below.

$$d(X, Y) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2} \quad (1)$$

This nearest neighbor clustering algorithm is presented in Figure 3 below.

Given:
A set X of objects $\{x_1, \dots, x_n\}$, A distance function $dis(c_1, c_2)$

1. **for** $i = 1$ to n
 $c_i = \{x_i\}$
 end for
2. $C = \{c_1, \dots, c_n\}$
3. $l = n+1$
4. **while** $C.size > 1$ **do**
 - a) $(c_{min1}, c_{min2}) = \text{minimum } dis(c_i, c_j)$ for all c_i, c_j in C
 - b) remove c_{min1} and c_{min2} from C
 - c) add $\{c_{min1}, c_{min2}\}$ to C
 - d) $l = l + 1$

end while

Fig. 3. The clustering procedure produces series of partitions of the data, C_n, C_{n-1} until C_1 . The first C_n consists of n single object cluster, while the last C_1 consists of single group containing all n cases. At each stage, the method joins the two clusters which are most similar.

5 Experimental Evaluations

The experiment was performed on the database containing 30 attributes comprising of six (6) major dimensions and a total of 1,000 tuples of e-commerce and transactional types of data. The evaluation platforms used in the study were IBM compatible computer, Windows XP, Sun Solaris 9, Visibroker, Java, and Python. The smart agent console indicates four host computers that were used for experiment. Searching and extraction of data were done through the active agents running on port 14000. The Sun Solaris Server is indicated by the PID 620 while the PID 4324, 3560, and 4016, respectively, are hosts running on Windows XP operating systems. In CORBA implementation, it indicates that the server (host) is active and is ready for communications with clients. The data obtained are in the form of online transactions. It corresponds to the items (attributes) that had been extracted from the host computer.

Table 1. Comparison of the discovered rules showing the two clusters and the first 5 rules

Models	Discovered Rules	Sup.	Conf.
Without HCARD (1,758 rules)	A6=Buy -> A2=Buy F4=Buy	0.935	0.942
	A6=Buy -> A2=Buy A3=Buy	0.927	0.934
	A6=Buy -> A2=Buy C4=Buy F4=Buy	0.916	0.922
	A6=Buy -> A2=Buy D2=Buy F4=Buy	0.915	0.921
	A6=Buy -> A2=Buy A3=Buy C4=Buy	0.910	0.916
HCARD Clusters 1(1154rules)	A6=Buy -> A2=Buy F4=Buy	0.924	0.939
	A6=Buy -> A2=Buy A3=Buy	0.919	0.934
	A6=Buy -> A2=Buy D2=Buy F4=Buy	0.905	0.920
	A6=Buy -> A2=Buy C4=Buy F4=Buy	0.903	0.918
	A6=Buy -> A3=Buy F4=Buy	0.901	0.915
2 (708 rules)	A6=Buy -> A2=Buy F4=Buy	0.912	0.923
	A6=Buy -> A2=Buy A3=Buy	0.910	0.921
	A6=Buy -> A2=Buy	0.963	0.974
	A6=Buy -> A2=Buy C4=Buy	0.942	0.953
	A6=Buy -> A2=Buy D2=Buy	0.940	0.951

The result in Table 1 only shows the two clusters and the first five rules generated. The support threshold that we set prior to the experiment was 0.90. In the original dataset, those who buy A6 (books on religion) will most likely buy A2 (social science) and F4 (Health) with support of 0.935 and confidence of 0.942 (94.2% probability of buying). The same fashion of analysis could be done to other rules. In cluster 1, those who buy A6 (religion) will most likely buy A2 (social science) and F4 (Health) with support of 0.924 and confidence of 0.939 (93.9%). Similar approach of analysis could be made for other rules in this cluster. And a similar fashion of explanation could also be done for other rules discovered such as in clusters 2, 3 and 4, respectively.

One good implication for our findings is that we reduced the task of making generalization on the rules obtained for the entire dataset. If we will increase the number of dataset entries, it will follow that the number of rules will be exponentially increasing and it is impractical to that effect to handle such output. We had noted convenience in analyzing the rules generated since we based it on the rules obtain per cluster.

The difference of computing performance is shown by the graph below, showing the comparison between HCARD (clustered) and without HCARD. Shorter processing time had been observed to compute for smaller clusters of attributes implying faster and ideal processing period than processing the entire dataset. The result of the time comparison is shown in Figure 4.

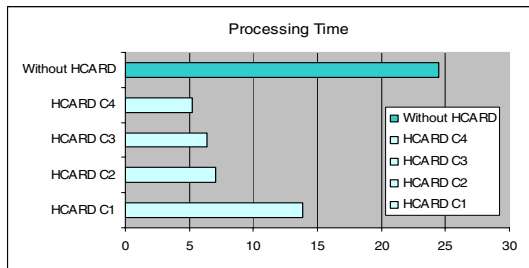


Fig. 4. Comparison of the Processing time. Shorter processing time had been noted when using the HCARD model as indicated by HCARD C1-C4.

The blending of cluster analysis and association rule generation in HCARD model specifically isolate groups of correlated cases using the hierarchical nearest neighbor clustering and then using of the extended data mining steps like the algorithm for association rule generation. The HCARD identify relatively homogeneous groups of cases based on selected characteristics and then employed the apriori algorithm to calculate for association rules. This resulted to some partitions where we could conveniently analyze specific associations among clusters of attributes. This further explains that the generated rules were discovered on clusters indicating highly correlated cases which eventually imply simplification of analyses for the data mining results.

6 Conclusions

Although our implementation uses heuristic approach, the experiment shows that efficiency in terms of convenience and practicality in analyzing the results based on the discovered rules had been observed. The Integrator Agent was developed based on CORBA architecture for search and extraction of data in the distributed environment. Shorter processing time had been noted in computing for smaller clusters implying faster and ideal processing period than dealing with the entire dataset. We have provided examples, performed experiments and generated rules but more rigorous treatment maybe needed if dealing with more complex databases. The HCARD model reveals clusters that have high correlation according to predetermined characteristics and generated isolated but imperative association rules based on clustered data which in return could be practically explained for decision making purposes. The discovery of such information often yields important insights into business and its client may lead to unlocking hidden potentials by devising innovative strategies.

References

1. El-Beltagy, S., DeRoure, D. C. and Hall, W. A. Multiagent system for Navigation Assistance & Information Finding. In Proceedings of the Fourth International Conference on the Practical Application of Intelligent Agents & Multi-Agent Technology, (1999) 281-295
2. Han J. & Kamber M.: Data Mining Concepts & Tech. USA, Morgan Kaufmann (2001)
3. Margaritis D., Faloutsos, C., & Thrun S.: NetCube: A Scalable Tool for Fast Data Mining & Compression. Conference on Very Large Databases (VLDB), Italy (2001) 311-320
4. Coulouris, G. Dollimore, J. and Kinderberg, T.: Distributed Systems: Concepts and Design. USA, Addison-Wesley (2001)
5. Object Management Group (OMG). Common Object Resource Broker Architecture (CORBA). Available at <http://www.omg.org> (2005)
6. Siourbas, C.: Determining the Number of Clusters. Available at: <http://cgm.cs.mcgill.ca/~soss/cs644/projects/siourbas> (2005)
7. Hellerstein, J.L., Ma, S. and Perng, C. S.: Discovering actionable patterns in event data. IBM Systems Journal, Vol. 41, No. 3 (2002) 475-493
8. Lam, J.M.W.: Multi-Dimensional Constrained Gradient Mining. Published Master's Thesis, Simon Fraser University (2001)
9. Chen, B., Haas, P., and Scheuermann, P.: A new two-phase sampling based algorithm for discovering association rules. Proceedings of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (2002) 462-468

A New Class of Filled Functions for Global Minimization*

Xiaoliang He, Chengxian Xu, and Chuanchao Zhu

School of Science, Xi'an Jiaotong University, Xi'an 710049, China
{hexl, mxu}@mail.xjtu.edu.cn

Abstract. Filled function method is a type of efficient methods to solve global optimization problems arisen in non-convex programming. In this paper, a new class of filled functions is proposed. This class of filled functions has only one adjustable parameter a . Several examples of this class of filled functions with specified parameter values are given, which contain the filled functions proposed in [3] and [4]. These examples show this class of filled functions contains more simple functions, therefore this class of filled functions have better computability. An algorithm employing the proposed filled function is presented, and numerical experiments show that the proposed filled functions are efficient.

1 Introduction

Global optimization problems arise in many fields of science and technology. Filled function method is a type of efficient methods to obtain the global solution of a non-convex multivariable function. This method was initially introduced by Ge([1]) and the following studies can be found in Ge and Qin([2, 6]), Xu et al ([5]), Liu([3]), Kong and Zhuang([7]), Zhang et al([8]).

The global optimization problem has the following form

$$\min_{x \in \Omega} f(x) \quad (1)$$

where $\Omega \in R^n$ is closed and bounded region, $f(x)$ is continuously differentiable on Ω , non-convex and has a finite number of minimizers on Ω .

The basic definitions(see [1]) about the filled function are as follows.

Definition 1. A function $P(x)$ is called a filled function of $f(x)$ at its local minimizer x_1 if

- 1) x_1 is a local maximizer of $P(x)$ and the whole basin(see [1]) B_1 correspond to x_1 becomes a part of a hill of $P(x)$ at x_1 ;
- 2) $P(x)$ has no stationary in any higher basin B_h than B_1 ;
- 3) There is a point x' in lower basin B_l (if such a basin exists) that minimizes $P(x)$ on the line through x' and x_1 .

* Key project supported by National Nature Science Foundation of China:10231060.

In this paper, we propose a modified definition for filled function:

Definition 2. A function $P(x)$ is called a filled function of $f(x)$ at its local minimizer x_1 if

- 1) $\lim_{x \rightarrow x_1} P(x) = +\infty$, and for any $x (\neq x_1)$ in basin B_1 correspond to x_1 , $d = x - x_1$ is a descent direction of $P(x)$ at x ;
- 2) $P(x)$ has no stationary in any higher basin B_h than B_1 ;
- 3) There is a point x' in lower basin B_l (if such a basin exists) that minimizes $P(x)$ on the line through x' and x_1 .

The difference between these two definitions is that the former is continuous and the later is infinite discontinuous at x_1 .

Through this paper, the filled function is given by definition 2.

The filled function method consists of two phases as follows:

Local minimization: Any available local minimization method such as the quasi-Newton method can be used to find a minimizer, x_1 say of $f(x)$.

Filling: A filled function $P(x)$ is constructed at the point x_1 . $P(x)$ has no stationary points in any higher basins B_h and does have a minimizer point in a lower basin B_l (if it exists). Then minimization of $P(x)$ starts from a point near x_1 . This Phase ends when a point x' is found that x' is in a lower basin B_l . Then, the process reenters **Local minimization** with x' as the starting point.

2 A New Class of Filled Functions

In this section we propose a new class of filled functions, it is

$$v(x) = \eta[f(x) - f(x_1)] - aw[(\|x - x_1\|)^p] \quad a > 0, p \geq 1 \tag{2}$$

where a, p are parameters, $w(t)$ and $\eta(t)$ satisfy the following properties:

(i) $\eta(t)$ is continuously differentiable for any $t \in (0, +\infty)$, and

$$\lim_{t \rightarrow 0^+} \eta(t) = +\infty, \lim_{t \rightarrow 0^+} \eta'(t) \rightarrow -\infty$$

(ii) $w(t)$ is continuously differentiable for any $t \in (0, +\infty)$, and $w(0) = 0, w(t) > 0$ for $t > 0$.

(iii) for $t > 0, w'(t) \geq c > 0, \eta'(t) < 0$ and $\eta'(t)$ is monotonically increasing.

The following theorems show the properties of the function $v(x)$.

Theorem 1. Suppose x_1 is a local minimizer of $f(x)$ and B_1 is the basin of $f(x)$ at $x_1, d \in R^n, d \neq 0$. If x satisfies $f(x) > f(x_1)$, then

- 1) For any $a > 0, \lim_{x \rightarrow x_1} v(x) = +\infty$.
- 2) If $d^T \nabla f(x) \geq 0, d^T(x - x_1) > 0$ or $d^T \nabla f(x) > 0, d^T(x - x_1) \geq 0$, then for any $a > 0, d$ is a descent direction of $v(x)$ at point x . Especially for $x \in B_1, x \neq x_1, d = x - x_1$ is a descent direction of $v(x)$ at point x .

Proof. 1) From the properties of $\eta(t)$, $w(t)$, It follows that for any $a > 0$, we have

$$\lim_{x \rightarrow x_1} v(x) = +\infty$$

2) It follows from definition of $v(x)$ that

$$d^T \nabla v(x) = \eta'[f(x) - f(x_1)]d^T \nabla f(x) - apw'(\|x - x_1\|^p) \|x - x_1\|^{p-2} d^T(x - x_1) \tag{3}$$

From the properties of $\eta(t)$ and $w(t)$, we have $\eta'[f(x) - f(x_1)] < 0$, $w'(\|x - x_1\|^p) \geq c > 0$. Under the conditions of theorem, it follows that for $a > 0$, the value of (3) is negative, that is

$$d^T \nabla v(x) < 0$$

hence, d is a descent direction of $v(x)$ at point x for any $a > 0$.

Especially, if $x \in B_1$, $x \neq x_1$, then $d = x - x_1$ satisfies the conditions of Theorem, and hence $d = x - x_1$ is a descent direction of $v(x)$ at point x .

Theorem 2. *Suppose x_1 is a local minimizer of $f(x)$ and B_1 is the basin of $f(x)$ at x_1 , $d \in R^n$, $d \neq 0$, and x satisfies $f(x) > f(x_1)$, $d^T \nabla f(x) < 0$, $d^T(x - x_1) > 0$. Let*

$$A(x) = \frac{\eta'[f(x) - f(x_1)]d^T \nabla f(x)}{pw'(\|x - x_1\|^p) \|x - x_1\|^{p-2} d^T(x - x_1)}. \tag{4}$$

If $a > A(x)$, then d is a descent direction of $v(x)$ at point x .

Proof. From the properties of $\eta(t)$ and $w(t)$, and $a > A(x)$, it follows that the value of (3) is negative, that is

$$d^T \nabla v(x) < 0$$

hence, d is a descent direction of $v(x)$ at point x .

Corollary 1. *When the value of the parameter a is selected properly, $v(x)$ has no stationary point in any higher basin than B_1 .*

Theorem 3. *Suppose x_1 is a local minimizer of $f(x)$ and B_1 is the basin of $f(x)$ at x_1 , $d \in R^n$, $d \neq 0$. x satisfies $f(x) > f(x_1)$,*

- 1) *if $d^T \nabla f(x) < 0$, $d^T(x - x_1) > 0$ and $a < A(x)$, where $A(x)$ is defined by (4), then d is an ascent direction of $v(x)$ at point x .*
- 2) *For any given $a > 0$, the condition $a < A(x)$ can be satisfied when $f(x) > f(x_1)$ and $f(x)$ close to $f(x_1)$.*

Proof. 1) From the properties of $\eta(t)$ and $w(t)$, and the condition $a < A(x)$, it follows that the value of (3) is positive, that is

$$d^T \nabla v(x) > 0$$

hence, d is an ascent direction of $v(x)$ at point x .

2) From the properties of $\eta(t)$, $w(t)$, it follows that $A(x) \rightarrow +\infty$ when $f(x) > f(x_1)$ and $f(x)$ close to $f(x_1)$. Hence if $f(x)$ close sufficiently to $f(x_1)$, then $a < A(x)$, and the proof is completed.

Corollary 2. *There exists a point x' in any basin B_l lower than the current basin B_1 that minimizes $P(x)$ on the line through x' and x_1 .*

From the analysis above, we know $v(x)$ is a filled function of $f(x)$.

In the rest part of this section, We give some examples of $v(x)$

Example 1. If we take $p = 2$, $\eta(t) = 1/\ln(1 + t)$, $w(t) = t$, then we can get the filled function $H_1(x)$, which was proposed in ([3]):

$$H_1(x) = \frac{1}{\ln[1 + f(x) - f(x_1)]} - a \|x - x_1\|^2 \tag{5}$$

Example 2. If we take $p = 2$, $\eta(t) = 1/t^\beta$, $w(t) = t$, where $\beta > 0$, then we can get the filled function $H_2(x)$:

$$H_2(x) = \frac{1}{[f(x) - f(x_1)]^\beta} - a \|x - x_1\|^2 \tag{6}$$

Example 3. If we take $p = 2$, $\eta(t) = 1/(\exp(t) - 1)$, $w(t) = t$, then we can get the filled function $H_3(x)$:

$$H_3(x) = \frac{1}{\exp[f(x) - f(x_1)] - 1} - a \|x - x_1\|^2 \tag{7}$$

Example 4. If we take $p = 2$, $\eta(t) = -\ln(t)$, $w(t) = t$, then we can get the filled function $H_4(x)$:

$$H_4(x) = -\ln(f(x) - f(x_1)) - a \|x - x_1\|^2 \tag{8}$$

$H_2(x)$, $H_3(x)$ and $H_4(x)$ are new filled functions proposed in this paper. The simple function $w(t) = t$ is employed in the above examples, other functions that satisfy the desired properties can also be chosen for $w(t)$.

Above examples show that the new filled function class contains more simple functions, which can be computed easily, so have better computability.

3 Algorithm

In this section, we describe the algorithm that employs the one-parameter filled functions. The algorithm is expressed as follows:

Algorithm

Step 0. Initialize the parameters $\varepsilon > 0$, $0 < \delta < 1$, $a > 0$, $\Omega \subset R^n$ (where, ε is the tolerance of error, δ is the coefficient of movement of point, a is the parameter in the filled function. Ω is the domain of $f(x)$ contains all minimizer);

Step 1. Minimize $f(x)$ by starting with an arbitrary point $x' \in \Omega$ and obtain the first local minimizer x_1 of $f(x)$;

Step 2. Construct the filled function $v(x)$ of $f(x)$ at point x_1 . For $i = 1, 2, \dots, 2n$, minimize $v(x)$ starting from $x_{i0} = x_1 + (-1)^i \delta e_{\lfloor \frac{i+1}{2} \rfloor}$, where $e_i (i = 1, 2, \dots, n)$ is

the i th unit vector. If the minimizing process from an initial point goes out of Ω without getting a desired point, then change to the next initial point. If no desired point is found starting from all the given initial points, then stop, x_1 is the global minimizer of $f(x)$. else go step 3.

Step 3. Let x_k be the point obtained in Step 2. If one of the following criteria holds at x_k , then the point x_k has been in a lower basin B_l and take x_k as x' :

- (a) $d_{k-1}^\top \nabla v(x_k) \geq 0$, where d_{k-1}^\top is the search direction at x_{k-1} ;
- (b) $(x_k - x_1)^\top \nabla v(x_k) \geq 0$;
- (c) $\|\nabla v(x_k)\| < \varepsilon$;
- (d) $f(x_k) < f(x_1)$.

Step 4. Minimizing $f(x)$ by starting with x' , one obtain another local minimizer x_2 of $f(x)$. If $f(x_2) \leq f(x_1)$, set $x_1 = x_2$ and go back to *Step 2*; else, increase a , set $a = \text{Max}\{10a, A(x_1)\}$. where $A(x)$ is defined by (4), then go back to *Step 2*.

4 Numerical Experiments

In this section, the new filled functions are used to solve some global test problems. These testing functions are usually used to evaluate the numerical performance of a new method((see [1, 3, 6])etc). We do experiment for more than ten test problems, Because of the page limitation,we list only two typical test problems

(1). Six-hump camel-back function($n = 2$)

$$f(x) = 4x_1^2 - 2.1x_1^4 + x_1^6/3 + x_1x_2 - 4x_2^2 + 4x_2^4, \quad (-3 \leq x_1, x_2 \leq 3)$$

It's global minima are (0.08983, -0.7126) and (-0.08983, 0.7126).

(2). Shubert-II function ($n = 2$)

$$f(x) = \left\{ \sum_{i=1}^5 i \cos[(i+1)x_1 + i] \right\} \left\{ \sum_{i=1}^5 i \cos[(i+1)x_2 + i] \right\} + 0.5[(x_1 + 1.42513)^2 + (x_2 + 0.80032)^2], \quad (-10 \leq x_1, x_2 \leq 10)$$

This function has about 760 local minima, and it's global minimum is (-1.42513, -0.80032). Because of the large number of local minima and the steep slope around the global minimum, this function has widely been recognized as an important testing function for global optimization.

The algorithm in section 3 is employed to find the global minimum of the test functions above. The local minimization of the function $f(x)$ and the filled function $v(x)$ is implemented using the quasi-Newton BFGS method, where Armijo and Goldstein principle is used to do line search, The stop rule used in BFGS method is $|g(x)| < \varepsilon$, where $g(x)$ is the gradient of $f(x)$ or $v(x)$.

The functions $H_1(x), H_2(x), H_3(x), H_4(x)$ are used in the algorithm, respectively. parameters used in the algorithm are $\varepsilon = 10^{-5}, \delta = 0.01, a = 1000$, the initial point is taken randomly. The experiments show that the algorithm finds the global minimizer for the test problems. we will not list the results and just give two graphs to describe the iteration process for problems (1) and (2).

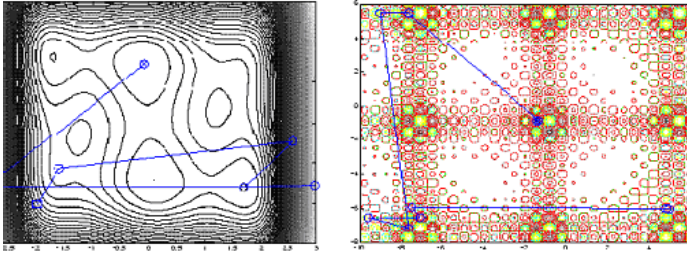


Fig. 1. The iterative points for problem(1), problem(2)

5 Conclusion

Filled function methods are a type of efficient methods for global optimization. In this paper, a new class of filled functions with one parameter is proposed, which contain the filled function proposed in [3] and [4]. Several examples of this class of filled function are given, and show the new class contains more simple functions, hence have better computability. Numerical experiments show that the proposed filled functions are efficient.

References

1. Ge Renu.: A filled function method for finding global minimizer of a function of several variables[J]. mathematical programming. **46**(1990) 191–204.
2. Ge R.P.and Qin Y.F.: A class of filled functions for finding a global minimizer of a function of several variables[J]. Journal of optimization theory and applications. **54**(1987) 241–252.
3. Xian Liu.: Finding global minima with a computable filled function. Journal of global optimization. **19**(2001) 151–161.
4. Xian Liu, Wilsun Xu.: A new filled function applied to global optimization. Computer and Operation Research. **31**(2004) 61–80.
5. Zheng Xu,Hongxuan Huang,Panos M.P.,Chengxian Xu.: Filled functions for unconstrained global optimization. Journal of global optimization. **20**(2001) 49–65.
6. Ge R.P.and Qin Y.F.: The globally convexized filled functions for globally optimization. Applied mathematics and computations. **35**(1990) 131–158.
7. Kong M. and Zhuang J.N. : A modified filled function method for finding a global minimizer of a non-smooth function of several variables. Numerical mathematics-A journal of chinese universities. **18**(2)(1996) 165–174.
8. Lian-Sheng Zhang, Chi-Kong Ng, Duan Li and Wei-wen Tian. : A new filled function method for global optimization. Journal of global optimization. **28**(2004) 17–43

Modified PSO Algorithm for Deadlock Control in FMS

Hesuan Hu, Zhiwu Li, and Weidong Wang

School of Eletro-Mechanical Engineering, Xidian University, Xi'an 710071, China
{hshu, wangwd}@mail.xidian.edu.cn, zhqli@xidian.edu.cn

Abstract. Both a concept of the optimal set of elementary siphons and a deadlock prevention policy based on integer programming are presented to solve deadlock problems arising in flexible manufacturing systems(FMS). Furthermore, an algorithm based on modified particle swarm optimization(PSO) is illustrated to show its efficiency to deal with such problems. Numerical simulation shows that this policy can minimize the number of newly additional control places and arcs while improving the dynamic performance of the resultant system.

1 Introduction

Flexible manufacturing systems(FMS) are prone to deadlock if the resources in them are not managed properly. Analysis and control upon deadlock problems arising in these systems are indispensable. Petri nets are suitable tools to deal with such problems for their strong abilities to describe discrete event systems with properties such as concurrency and stochastic[1 – 4].This paper proposes the concept of the optimal set of elementary siphons based on such an idea, a novel deadlock prevention policy derived from integer programming is also presented. According to this policy, the flow relations and the number of initial tokens can be generated through an optimization model[4]. In other words, many theories of the nonlinear integer programming can be utilized to characterize deadlock problems in FMS. Due to the speciality of Petri nets, the objective function of such a problem is discrete and linear, while the constrained ones are discrete and nonlinear. Many traditional optimization models cannot solve such problems precisely. Fortunately, an efficient evolutionary algorithm named by particle swarm optimization(PSO) suitable to deal with them is developed by Kennedy and Eberhart in 1995[5, 6]. PSO derived from the model of a swarm of bird to find the highest density of food in limited times without any knowledge of the field a priori. Unfortunately, traditional PSO algorithm can only be used to solve problems in real space. The modified PSO presented in [2] and modified in this paper makes the solution of integer programming feasible[6]. Finally, an example is illustrated to show the feasibility of our policy.

2 Traditional And Modified PSO Algorithm[5, 6]

PSO is a 5-tuple $PSO=(N_{popu}, K_{iter}, V, P, F_{fit})$, where N_{popu} means the size of the swarm, and K_{iter} the iterative times. V and P mean the location

and velocities of all the particles respectively. F_{fit} means the fitness function. Supposing there are m particles in D -dimension space, then the i_{th} particle can be denoted as $x_i=(x_{i_1}, x_{i_2}, \dots, x_{i_d}), i=1, 2, \dots, m$. The best location where x_i encountered is denoted as $p_i=(p_{i_1}, p_{i_2}, \dots, p_{i_d}), i=1, 2, \dots, m$. The best location where all the particles encountered is denoted as $p_g=(g_1, g_2, \dots, g_d)$. Given the k -generation of the i_{th} particle, location and velocities of its $k + 1$ -generation can be generated by the following equations[5]. $v_i^{k+1} = wv_i^k + c_1r_1(p_i - x_i^k) + c_2r_2(p_g - x_i^k), x_i^{k+1} = x_i^k + v_i^k, i=1, 2, \dots, m$, where r_1 and r_2 are random numbers between 0 and 1. w is known as the inertial weight. c_1 and c_2 are scaling factors. v_i is bounded by the maximum velocity v_{max} , which means $v_i = v_{max}$ if $v_i > v_{max}$. In optimization model, objective function is usually taken as fitness function. The feasible space of traditional particle swarm optimization is the real space. However, the optimization problems arising in Petri nets always develop in the integer space. So, it is necessary to modify the traditional algorithm so as to solve the integer programming problem. Through equation $x_i^{k+1} = x_i^k + v_i^k$, we can see that fitness function and location of every particle will evolve in the integer space strictly if and only if location and velocity are integer in every iteration. So the integer programming problem can be solved by the following equations[6]. $v_i^{k+1} = int(wv_i^k + c_1r_1(p_i - x_i^k) + c_2r_2(p_g - x_i^k)), x_i^{k+1} = x_i^k + v_i^k$, where $i=1, 2, \dots, m$.

3 Deadlock Prevention Based on Modified PSO

Since Petri nets are used to model flexible manufacturing systems(FMS), some basic notions about them should be presented to make this paper self-contained. Let S_1, S_2, \dots, S_n be the siphons in net N . T-Vectors $\eta_1, \eta_2, \dots, \eta_m$ form a vector space. The base of the vector space is denoted by $\eta_B=\{\eta_{B_1}, \eta_{B_2}, \dots, \eta_{B_k}\}$, where k is the rank of the vector space. Then $S_{B_1}, S_{B_2}, \dots, S_{B_k}$ are called elementary siphons in net N [3]. The optimal set of elementary siphons are those siphons whose being controlled will introduce the most number of reachable states[1]. Let $N=(P, T, F)$ be a Petri net and w be the number of elementary siphons of N . We have $w \leq \min(|P|, |T|)$. This is trivial due to $w=Rank(\eta_S^T)=Rank(\lambda_S^T \bullet [N]) \leq \min(Rank(\eta_S))$, thus $Rank([N]) \leq \min(|P|, |T|)$ holds. Let (N_0, M_0) be a net system, and $v_p=1\{p \notin S\}, z_t=1\{t \notin S^\bullet\}$, then $S=\{p_i | i=1, 2, \dots, n\}$ is the maximal siphon empty at M if and only if $G(M_0) < |P|$ holds. where

$$G(M)=\min \sum v_p$$

s.t.

$$z_t \geq \sum_{p \in \bullet t} v_p - |\bullet t|, \forall t \in T, \tag{1}$$

$$v_p \geq z_t, \forall (t, p) \in F, \tag{2}$$

$$v_p \geq M(p)/SB(p), \forall p \in P, \tag{3}$$

$$v_p, z_t \in 0, 1 \tag{4}$$

$$M = M_0 + [N_0] \bullet Y, M \geq 0, Y \geq 0, \tag{5}$$

Here, $[N_0]$ means the incidence matrix of N_0 and Y means the firing sequence[4], $SB(p) = \max\{M(p) \mid M = M_0 + [N_0] \bullet Y, M \geq 0, Y \geq 0\}$. For detail, please refer to [4].

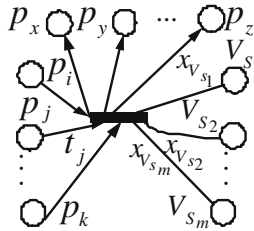


Fig. 1. Controlling places and their flow relations with other transitions

Let (N_0, M_0) be a net, it is deadlock free if and only if $G(M) = |P|$ holds. Let (N_0, M_0) be a net with n places p_1, p_2, \dots, p_n , m elementary siphons S_1, S_2, \dots, S_m . Select S_1, S_2, \dots, S_m as elementary siphons and add m controlling places $V_{S_1}, V_{S_2}, \dots, V_{S_m}$ on it, Let (N_1, M_1) be the newly generated net system, N_1 is deadlock free if and only if $G(M_1) = n + m$. Fig. 1 shows the flow relations between $V_{S_i} (i = 1, 2, \dots, m)$ and $t_j (j = 1, 2, \dots, |T|)$ in (N_1, M_1) . Let $x_{V_{S_i} t_j} = 1$ when $V_{S_i} \in t_i^\bullet$, $x_{V_{S_i} t_j} = -1$ when $V_{S_i} \in {}^\bullet t_i$, and $V_{S_i} \notin t_i^\bullet \cup {}^\bullet t_i$ when $x_{V_{S_i} t_j} = 0$, constraint(2) in (N_1, M_1) can be written as

$$z_{t_j} \geq \sum v_p - |p_{ij\dots k}| + \sum_{i=1}^m \frac{x_{V_{S_i} t_j}^2 - x_{V_{S_i} t_j}}{2} V_{S_i} - \sum_{i=1}^m \frac{x_{V_{S_i} t_j}^2 - x_{v_{S_i} t_j}}{2} + 1 \tag{6}$$

For t_j , m rows should be added as follows.

$$v_{p_i} \geq \frac{x_{V_{S_i} t_j}^2 + x_{V_{S_i} t_j}}{2} z_{t_j}, i = 1, 2, \dots, m \tag{7}$$

m row vectors should be added to the corresponding incidence matrix and the new matrix should be named as N_1 . Let every elements in N_1 be e_{ij} , then

$$e_{ij} = x_{V_{S_i} t_j} \tag{8}$$

The resultant system with m newly added monitors will be deadlock free if $G(M_1) = \min \sum v_p = n + m$. Let d be a medium variable, the objective function in the integer programming model can be changed to be constraint $\sum v_p + d = m + n$. Thus the objective function $G(M_1) = \min \sum v_p$ can be replaced by $G'(M_1) = \min \{d\}$. Many experimental results show that more reachable states

will be generated in the resultant net system if more initial tokens are contained in the controlling places. So $\max \sum M(V_{S_i})$ should taken as another objective function in the whole system control. Thus, the whole optimization algorithm can be described by the following integer programming model.

$$G'(M_1) = \{ \min\{d\} \cap \max\{\sum M(V_{S_i})\} \}$$

$$\text{s.t. (2)(3)(4)(6)(7)(8)}$$

$$M' = M'_0 + N'_0 \bullet Y, M \geq 0, Y \geq 0, \quad (9)$$

$$SB'(p) = \max\{M'(p) \mid M' = M'_0 + N'_0 \bullet Y, M', Y \geq 0\}, \quad (10)$$

$$d \geq 0, x_{V_{S_i}t_j} \in \{-1, 0, 1\} \quad (11)$$

The whole deadlock prevention policy based on the optimal set of elementary siphons and integer programming can be summarized as follows. Step1. Model the FMS with a Petri net (N_0, M_0) system. Step2. Generate the optimal set of elementary siphons $\Pi_E = S_1, S_2, \dots, S_m$. Step3. Add m controlling places $V_{S_1}, V_{S_2}, \dots, V_{S_m}$. Let the weight of arcs between V_{S_i} ($i = 1, 2, \dots, m$) and t_j ($j = 1, 2, \dots, |T|$) be $x_{V_{S_i}t_j}$, and the resultant net system be (N_1, M_1) . Step4. Generate the integer programming model $G'(M_1)$ of (N_1, M_1) . Step5. Solve $G'(M_1)$, solutions of $x_{V_{S_i}t_j}$ represent the flow relations between V_{S_i} and every transition while $M(V_{S_i})$ means the number of initial tokens in V_{S_i} . Step6. End the whole algorithm and the resultant net system will be deadlock free. In step5, modified particle swarm optimization will be utilized to solve the integer programming model. Its detail procedures can be described as follows. Step1. Initiate the places and velocities of the particles. Step2. Generate the fitness function for every particles and compare it to the current optimal one, renew the current optimal fitness if the newly generated one is better than it, and record the places of all the particles. Step3. Generate the global fitness function value for every particles and compare it to the current global optimal one, renew the current optimal global fitness function value if the newly generated one is better than it, and record the places of all the particles. Step4. Renew the place and velocity of every particle. Step5. Repeat Step2 to Step4 until the condition is satisfied.

4 Experimental Results

In Fig. 2, an FMS cell with 3 robots and 3 machines is illustrated, where each robot can hold one part and each machine can produce parts during one time. There are also 3 input buffer denoted by I1, I2, I3 and 3 output buffer denoted by O1, O2, O3. The product type characterizes the process to be made in the cell as follow: 1) the raw product P1 is taken from I1 and once it has been manufactured, is moved to O1. This part can be manufactured by either M1, M2 or M3. 2) the raw product P2 is taken from I2, manufactured by M3 followed by treatment in M1, and, finally, placed in O2; 3) the raw product P3 is taken

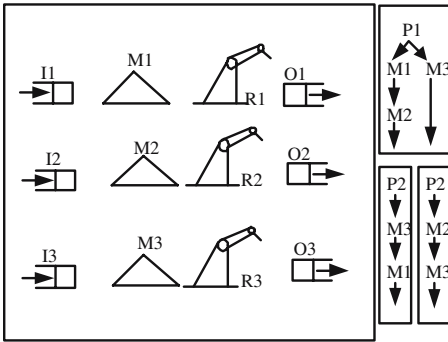


Fig. 2. A simple production cell

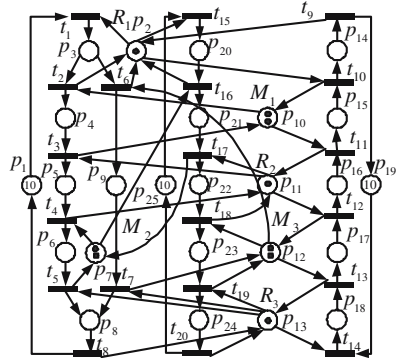


Fig. 3. Petri net modelling system in Fig. 2

from I3, manufactured by M2 followed by treatment in M3, and , finally, placed in O3. Fig. 3 shows the Petri net model for the system in Fig. 2.

There are 17 siphons in such a net system, $S_1=\{p_8, p_{12}, p_{13}, p_{17}, p_{24}\}$, $S_2=\{p_6, p_7, p_{11}, p_{16}, p_{22}\}$, $S_3=\{p_5, p_9, p_{11}, p_{12}, p_{16}, p_{23}\}$, $S_4=\{p_5, p_{10}, p_{11}, p_{15}, p_{22}\}$, $S_5=\{p_5, p_8, p_{11}, p_{12}, p_{13}, p_{16}, p_{24}\}$, $S_6=\{p_6, p_7, p_9, p_{11}, p_{12}, p_{16}, p_{23}\}$, $S_7=\{p_6, p_7, p_{10}, p_{11}, p_{15}, p_{22}\}$, $S_8=\{p_5, p_9, p_{10}, p_{11}, p_{12}, p_{15}, p_{23}\}$, $S_9=\{p_7, p_8, p_{11}, p_{12}, p_{13}, p_{16}, p_{24}\}$, $S_{10}=\{p_5, p_8, p_{10}, p_{11}, p_{12}, p_{13}, p_{15}, p_{24}\}$, $S_{11}=\{p_2, p_5, p_9, p_{10}, p_{11}, p_{12}, p_{14}, p_{20}, p_{23}\}$, $S_{12}=\{p_2, p_3, p_6, p_7, p_{10}, p_{11}, p_{14}, p_{22}\}$, $S_{13}=\{p_2, p_5, p_8, p_{10}, p_{11}, p_{12}, p_{13}, p_{14}, p_{20}, p_{24}\}$, $S_{14}=\{p_6, p_7, p_9, p_{10}, p_{11}, p_{12}, p_{15}, p_{23}\}$, $S_{15}=\{p_7, p_8, p_{10}, p_{11}, p_{12}, p_{13}, p_{15}, p_{24}\}$, $S_{16}=\{p_2, p_6, p_7, p_9, p_{10}, p_{11}, p_{12}, p_{14}, p_{23}\}$, $S_{17}=\{p_2, p_7, p_8, p_{10}, p_{11}, p_{12}, p_{13}, p_{14}, p_{24}\}$. The number of tokens contained in them can be illustrated as follows $M(S_1) = 3, M(S_2) = 3, M(S_3) = 3, M(S_4) = 3, M(S_5) = 4, M(S_6) = 5, M(S_7) = 5, M(S_8) = 5, M(S_9) = 6, M(S_{10}) = 6, M(S_{11}) = 6, M(S_{12}) = 6, M(S_{13}) = 7, M(S_{14}) = 7, M(S_{15}) = 8, M(S_{16}) = 8, M(S_{17}) = 9$. Utilizing the algorithm for the elementary siphons, the optimal set of elementary siphons can be $S_1, S_2, S_3, S_4, S_9, S_{11}, S_{12}, S_{16}$. Finally, the policy based on particle swarm optimization is exploited to generated

Table 1. Arcs and initial tokens of controlling places

V_{S_i}	$\bullet V_{S_i}$	$V_{S_i}^\bullet$	$M(V_{S_i})$
1	t_2, t_7, t_{13}, t_{19}	t_1, t_{14}, t_{15}	2
2	t_4, t_6, t_{17}	t_1, t_{15}	2
3	t_{12}, t_{18}	t_{14}, t_{15}	2
4	t_3, t_6, t_{11}	t_1, t_{14}	2
9	t_5, t_7, t_{12}, t_{19}	t_1, t_{14}, t_{15}	5
11	t_3, t_6, t_{10}, t_{18}	t_1, t_{14}, t_{15}	5
12	t_4, t_6, t_{10}, t_{17}	t_1, t_{14}, t_{15}	5
16	t_4, t_6, t_{10}, t_{18}	t_1, t_{14}, t_{15}	3

the input and output arcs of V_{S_i} ($i=1, 2, 3, 4, 9, 11, 12, 16$) and also their corresponding initial tokens shown by Table 1. Simulation results show that the newly generated system with such 8 controlling places is live and 4358 reachable states can be produced in it. When the policy proposed in[2] is used, 17 controlling places is needed and the dynamic behavior of the resultant system will also be bounded seriously. It is not difficult to predict that the the advantage of the policy based on particle swarm optimization will be more obvious if the size of the net is large.

5 Conclusion

A deadlock prevention policy based on modified particle swarm optimization and optimal set of elementary siphons is illustrated. Experimental results show that the adoption of such an optimization algorithm can minimize the number of newly added controlling places and arcs while generating more reachable states. Further research will move in the following directions. First, development of modelling the whole net system with swarms in the algorithm directly is needed. Second, analysis on the computational complexity should be investigated.

References

1. Li Z. W., Hu H. S., Zhou M. C.:An Algorithm for an Optimal Set of Elementary Siphons in Petri Nets for Deadlock Control.in Proceedings of the IEEE International Conference on Systems, Man and Cybernetics. **5**(2004)4849-4854.
2. Ezpeleta J.,Colom J. M., Martinez J.:A Petri net based deadlock prevention policy for flexible manufacturing systems. IEEE Transactions on Robotics and Automation. **11**(1995)173-184.
3. Li Z. W., Zhou M. C.:Elementary siphons of Petri nets and their application to deadlock prevention for flexible manufacturing systems. IEEE Transactions on Systems, Man, and Cybernetics. **34**(2004)38-51.
4. Feng C., Xie X. L.:Deadlock analysis of Petri nets using siphons and mathematical programming. IEEE Transactions on Robotics and Automation. **13**(1997)793-804.
5. Kennedy J., Eberhart R. C.:Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks. **4**(1995)1942-1948.
6. Tan Y., Gao H. M., Zeng J. C.:Particle Swarm Optimization for Integer Programming. System Engineering Theory and Practice(in Chinese), **24**(2004)126-129.

Optimization Design of Controller Periods Using Evolution Strategy*

Hong Jin, Hui Wang, Hongan Wang, and Guozhong Dai

Institute of Software, Chinese Academy of Sciences, Beijing 100080, China
{hjin, hui.wang, wha, dgz}@iel.iscas.ac.cn

Abstract. For real-time computer-controlled systems, a control task does not have a fixed period but a range of periods in which control performance varies. Hence for multiple tasks scheduled on a single processor, to consider the optimization design of sampling periods in the co-design of control and scheduling is necessary to improve the control performance and use limited computing resource efficiently. In this paper, the mathematic description of the optimization problem of designing periods is presented, and the optimization solution using evolution strategy is proposed. The performances of proposed solution are revealed via simulation studies. Simulation shows that the optimization design of sampling periods can be implemented by using the evolution strategy method.

1 Introduction

In the scheduling design of a real-time control system, a digital controller is normally designed assuming a fixed sampling period and a fixed computation delay, these simplistic design assumptions are seldom met in the target system^{[1][2]}. Some characteristics of a controller task, e.g., the sampling period, may be uncertainty usually and can change randomly in an interval. In reality, the sampling period of controller is also a design parameter and typically chosen according to rules of thumb^[3].

Unlike other real-time tasks, a control task has a range of sampling period in which it can be scheduled^[4], and the sampling period is usually larger than its worst-case execution time. For multiple tasks, the lower bound of each task's sampling period can be determined by the schedulability constraint. The upper bound of each task's sampling period can be determined by the stability constraint of its plant. The given range of sampling period needs be acceptable to its control loop. The smaller sampling period will need more computing resource and may violate the schedulability; however, the larger sampling period may violate the robustness of system. So the optimization design of sampling periods needs be considered in codesign of control and scheduling.

Seto D. et al. studied task period selection for RM scheduling algorithm, assumed all tasks are schedulable for a known upper bound of period, and use a branch and bound approach to search the feasible periods for given task set^[5]. Lee H. S. et al.

* This work is supported by China NSF under Grant No. 60374058, 60373055 and 60473039; the National High-Tech Research and Development Program of China under Grant No. 2004AA412040 and 2004AA1Z2450.

proposed a performance index that involves control performance and energy consumption based on the assumption that a control task does not have a fixed period but a range of period in which control performance varies, and gave a static solution to obtain optimal processor speed and a set of periods ^[4].

Evolution strategies are algorithms which imitate the principles of natural evolution as a method to solve parameter optimization problems ^[6], which is applied to optimize the period design here. The paper is arranged as follows: in Section 2, we describe and formulate the optimization designing problem of sampling periods, and give its control performance index and search space; the evolution strategy is introduced in Section 3; Section 4 gives simulation results; finally, the conclusion is given.

2 Mathematic Models

2.1 Controller Task

For a given plant, an appropriate control law can be designed to control the plant to satisfy a given constraint. In a control loop with feedback, the dispersion of the reference input $r(t)$ with the system output $y(t)$ is used as the input of the controller which recalculates a new control signal $u(t)$ used as the input of plant for every P seconds. The control aim is to make $y(t)$ approach to $r(t)$ as quickly as possible.

The controller (T_i) can be derived and implemented as an example of a periodic, hard real-time control task having a fixed sampling period P_i , a known worst-cast execute time C_i and a hard deadline d_i , which is usually composed of A/D sampling of input data, the computation of control algorithm, and D/A transform of output signal. For simplicity, assuming that A/D and D/A conversions take negligible amount of time, the computation of control output takes up most of the execution time of the control task. Without loss of generation, A/D and D/A transforms are neglected.

2.2 Problem Statement

Given a set of n periodic control tasks $\{T_1, \dots, T_n\}$ with execution time C_1, \dots, C_n and period ranges $[P_{i,min}, P_{i,max}]$, find all the possible optimal periods P_1, \dots, P_n such that all the tasks are schedulable and an integrated control performance index gets optimized. For a chosen scheduling policy, a static optimal solution to the problem can be found offline. However since all tasks have independent ranges of allowable periods, a full search using computer simulation may take indefinite amount of time as n is increased. A practical solution with evolution strategy technology is proposed.

2.3 Objective Function

Two principles were presented for choosing of periods ^[5]: (1) the period of each task will be bounded above by some value corresponding to the maximal permissible latency requirement associated with the task, and (2) the performance of a task is often inversely related to the task's period, i.e., the shorter the period, the better the performance. In simulation interval $[0, TS]$ where TS is the simulation time, let $n_i = \lfloor TS/P_i \rfloor$, then T_i has n_i instances in the interval. Let $e_i(t) = y_i(t) - r_i(t)$, $Q = \text{diag}(w_1, \dots, w_n)$, $E(t) = (e_1(t), \dots, e_n(t))$, $t_j = jP_i$, where, y_i and r_i are its actual output and reference input respectively. Consider the following control performance according to above principles:

$$f(P_1, \dots, P_n) = \int_{t=0}^{TS} E(t)QE(t)' dt = \int_{t=0}^{TS} \sum_{i=1}^n w_i e_i^2(t) dt = \sum_{i=1}^n w_i \int_{t=0}^{TS} e_i^2(t) dt = \sum_{i=1}^n w_i f_i(P_i)$$

$$f_i(P_i) = \int_0^{TS} e_i^2(t) dt = \int_0^{TS} (y_i(t) - r_i(t))^2 dt = P_i \sum_{j=1}^{n_i} (y_i(t_{ij}) - r_i(t_{ij}))^2$$

The quadratic term emphasizes minimizing the magnitude of weighted square error between the control output and reference input, where, f_i and w_i are the control performance and weighed coefficient of the i th subsystem respectively. Here, f can be considered as the integrated control performance and used as the fitness function in the mutation of evolution strategy. The performance function can express with a suitable performance index of control system, which is often a target for optimization from which a control algorithm is derived. The periods which optimize the performance are called optimal periods. The objective value is often referred to as fitness. Let $X=(P_1, \dots, P_n)$, the above optimization problem can be described as follows:

$$\min_{X \in R^n} f(X) = \min_{X \in R^n} CX = \min_{X \in R^n} \{c_1 P_1 + \dots + c_n P_n\}, \text{ where } c_i = w_i \sum_{j=1}^{n_i} (y_i(t_{ij}) - r_i(t_{ij}))^2$$

2.4 Search Space

Obviously, the above performance index is a linear function of all sampling periods. Moreover, the objective function directly depends on the scheduling policy chosen in scheduling also. So the search space of $\{P_1, \dots, P_n\}$ is constrained by:

- 1) The lower and upper bound limits, i.e., $P_{i,min} \leq P_i \leq P_{i,max}, i=1, 2, \dots, n;$
- 2) Schedulability constraint, i.e., all tasks are schedulable. Let Thr be the upper bound of utilization, then, $Thr=1$ for EDF policy or $Thr=n(2^{1/n}-1)$ for RM policy.

3 Evolution Strategy

3.1 Mutations

Let $X=(P_1, P_2, \dots, P_n)$ be the vector representing a point in the search space; σ_i be the standard deviation of P_i , $\Sigma=diag(\sigma_1^2, \dots, \sigma_n^2)$ be a matrix with diagonal elements of variances, k scale generation. Mutations are realized by replacing $X(k)$ by $X(k+1)=X(k)+N(0, \Sigma)$, where $N(0, \Sigma)$ is a vector of independent random Gaussian number with a mean of zero vector and variance matrix of Σ . The offspring (the mutated individual) is accepted as a new member of the population (it replaces its parent) iff it has better fitness and all constraints (if any) are satisfied. For example, if f is the objective function, an offspring $(X(k+1), \Sigma)$ replaces its parent $(X(k), \Sigma)$ iff $f(X(k+1)) < f(X(k))$. Otherwise, the offspring is eliminated and the population remains unchanged.

3.2 Convergence

If all standard deviations are identical, the optimization problem is regular, and it is possible to prove that the global optimum is found with probability one for sufficiently long search time; however, the conclusion does not provide any clues for the convergence rate (quotient of the distance covered towards the optimum and the number of elapsed generations needed to cover this distance).

3.3 1/5 Success Rule

Usually the standard deviation of normal distribution with which periods are mutated determines the search neighborhood size. In order to control the mutation strength, a measure for the realized neighborhood size also, we adopt the “1/5 success rule”.

Let ϕ be the ratio of successful mutations that an offspring has a better objective value than its parent to all mutations. The mutation strength, i.e., the variance of mutation operator, is increased after a certain number of generations if ϕ is greatly than 1/5, and the search would continue in “larger” steps; otherwise, decreased if $\phi < 1/5$, and the steps would be shorter, where ϕ is the success ratio of the mutation operator.

4 Simulation

Consider the optimization design of four servo systems, every which is described by using the transfer function of $G(s)=1000/[s(s+1)]$, and controlled by a PD controller which discrete control algorithm form is $u(t)=Prop(t)+Der(t)$, where $Prop(t)=K(r(t)-y(t))$, and $Der(t)=\alpha Der(t-P)+\beta[y(t-P)-y(t)]$, $\alpha=M/(NP+M)$, $\beta=NKM/(NP+M)$, K and M are control parameters, N is constant, P is the sampling period^[3]. The reference input of each subsystem is a step-function all, e.g., $r(t)=1$ when $t \leq 0.5s$ or $1s < t \leq 1.5s$; or $r(t)=-1$ in other time interval. And the simulation time is 2s.

For the four controller tasks (T_1, T_2, T_3 and T_4), their K parameter values are 1, 1.2, 0.9 and 1.1; M parameter values are 0.04, 0.042, 0.05 and 0.03; N parameter values are 30, 3, 100 and 50; their execute time are 1, 1, 2 and 2. The lower and upper bounds of sampling period are 6 and 32 respectively.

4.1 Optimization Design

In following simulation, all standard deviations are the same, and the initial feasible solution is $X(0)=(32,32,32,32)$. The increase and decrease rates for the variance of mutation are $coe_i=1/0.8$, $coe_d=0.8$ respectively. Figure 1 gives the control

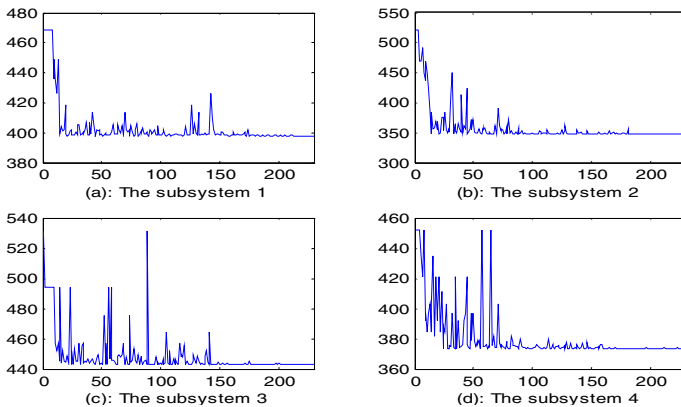


Fig. 1. Control performance indexes (f_i) of 4 subsystems

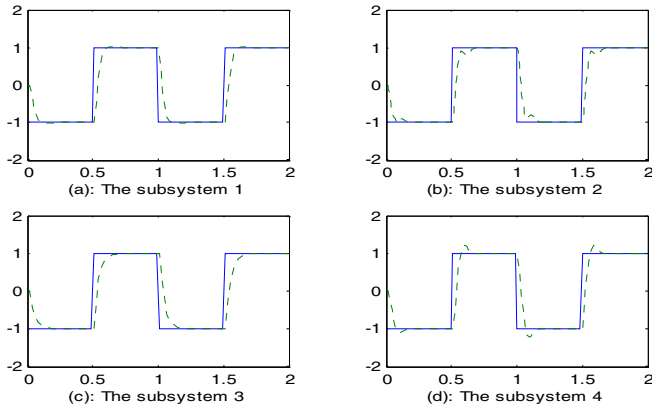


Fig. 2. Control performance comparisons for optimization solution

performance (f_i) of four subsystems; The X-axis scales the iterative number. The mutation will be stable soon, and the final optimization solution is $X_{opt}=(17,6,21,15)$.

4.2 Control Performance Comparison

Figure 2 gives comparisons between the reference signal r (the real line) and the actual measurement y (the dotted line) of four subsystems for the optimization solution X_{opt} . Compared with the initial sampling period, for the optimization solution, the mean control performance value decreases from 450.4 to 390.366, and the control performances of the 2nd and 3rd subsystems get improved obviously.

5 Conclusion

In this paper, we studied the issue of designing the sampling periods for a given set of real-time periodic control tasks. The optimization design is implemented by using the evolution strategy such that an integrated control performance index gets minimized and all tasks are schedulable for a chosen scheduling policy. The objective function is a linear weighed summation of sampling periods such that the optimization is similar to the nonlinear programming problem.

In the use of evolution strategy, the 1/5 success rule is adopted to control the mutation strength. Simulation shows that the integrated control performance of whole system gets greatly improved under the optimization solution of sampling periods, and any violation of all deadline constraints can get avoided.

References

1. Lin, Q., Chen, P.C.Y., Neow, P.A.: Dynamical scheduling of digital control systems. *Proc. of IEEE International Conference on Systems, Man and Cybernetics*. Washington, D.C., USA. Piscataway, NY: IEEE Computer Society. Vol.5 (2003) 4098-4103
2. Cervin, A., Henriksson, D., Lincoln, B., Eker, J., Arzen, K. E.: How does control timing affect performance? Analysis and simulation of timing using Jitterbug and TrueTime. *IEEE Control Systems Magazine*. Vol.23, No.3 (2003) 16-30

3. Jin, H., Wang, H. A., Wang, H., Wang, D. L.: Scheduling design of controllers with fuzzy deadline. In: Wang, L., Jin, Y. (eds.): *Fuzzy Systems and Knowledge Discovery*. Lecture Notes in Computer Science. Springer-Verlag Berlin Heidelberg. Vol.3613 (2005) 861-864
4. Lee, H. S., Kim, B. K.: Design of digital control systems with dynamic voltage scaling. *Proc. of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium*. Toronto, Canada. Piscataway, NJ: IEEE Computer Society. (2004) 94-101
5. Seto, D., Lehoczky, J. P., Sha, L.: Task period selection and schedulability in real-time systems. *Proc. of the 19th IEEE Real-Time Systems Symposium*. Madrid, Spain. Piscataway, NJ: IEEE Computer Society. (1998) 188-198
6. Michalewicz, Z.: *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996)

Application of Multi-objective Evolutionary Algorithm in Coordinated Design of PSS and SVC Controllers

Zhenyu Zou, Quanyuan Jiang, Pengxiang Zhang, and Yijia Cao

Department of Electrical Engineering, Zhejiang University, Hangzhou 310027, China
jameszou@zj165.com

Abstract. A multi-objective evolutionary algorithm (MOEA) based approach to Power System Stabilizer (PSS) and Static Var Compensators (SVC) tuning has been investigated in this paper. The coordinated design problem of PSS and SVC is formulated as a multi-objective optimization problem, in which the system response is optimized by minimizing several system-behavior measure criterions. MOEA is employed to search optimal controller parameters. Design of the multi-objective optimization aims to find out the Pareto optimal solution which is a set of possible optimal solutions for controller parameters. And effectiveness of the proposed control scheme has been demonstrated in a multiple power system.

1 Introduction

In the past two decades, the conventional power system stabilizer (PSS), i.e. a fixed parameters lead-lag compensator, is widely used by power system utilities. The recent advances in power electronics have led to the development of the flexible alternating current transmission systems (FACTS). FACTS are designed to overcome the limitations of the present mechanically controlled power systems and enhance power system stability by using reliable and high-speed electronic devices. One of the promising FACTS devices is the Static Var Compensators (SVC). SVC is a shunt compensation component and can quickly regulate its susceptance to provide dynamic reactive compensation and maintain the bus voltage in power system [1].

However, in the literature, a little work has been done on the coordination problem investigation of excitation and FACTS controllers. Mahran et al. [2] presented a coordinated PSS and SVC control for a synchronous generator. However, the proposed approach uses recursive least squares identification, which reduces its effectiveness for on-line applications. Moreover, optimum feedback strategies for both SVC and exciter controls presented in [3] require some or all states to be measurable or estimated. It leads to a centralized controller for multi-machine power systems, which reduces its applicability and reliability.

In this paper, a systematic method for coordinated design of PSS and SVC is investigated, aiming to improve the power angle stability and maintain the voltage quality of the power system simultaneously. The coordinated design problem of multiple power system controllers is formulated as a nonlinear constrained multi-objective optimization problem, where the multi-objective evolutionary algorithm (MOEA) is

employed to search for the Pare-optimal solutions of controller parameters. Simulation results in multi-machine power system show the effectiveness of the proposed design method of coordinated controllers.

2 Mathematic Model of Power System with PSS and SVC

A multi-machine power system as shown in Fig. 1 is considered. Generator 2 and 3 are equipped with the PSS and one SVC is installed at the bus 101. The generator is represented by the four-order model comprising of the electromechanical swing equation and the generator internal voltage equation. For the k -th generator, the differential equations are given as follows.

$$\begin{aligned}\rho\delta_k &= \omega_0(\omega_k - 1) \\ \rho\omega_k &= (P_{m_k} - P_{e_k} - D_k(\omega_k - 1)) / M_k \\ \rho E'_{dk} &= (E_{fdk} - (x_{dk} - x'_{dk})I_{dk} - E'_{qk}) / T'_{d0k}\end{aligned}\quad (1)$$

where ρ is the derivative operator d/dt . The state variables and parameter in the above equations have the usually physical meaning in Ref [4].

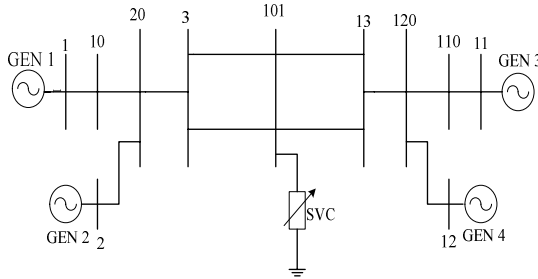


Fig.1. Single-line diagram of the power system.

The IEEE Type-ST1 excitation system is considered. It can be described as

$$\rho E_{fdk} = (-E_{fdk} + K_A(V_{refk} - v + u_{PSSk})) / T_A \quad (2)$$

where K_A , T_A are the gain and time constant of the excitation system, respectively. And, a conventional lead-lag PSS is installed in the feedback loop to generate a stabilizing signal u_{PSS} .

SVC lead-lag controller is employed. The susceptance of the SVC, B_{SVC} , can be expressed as

$$\rho B_{SVC_i} = (K_{S_i}(V_{SVCref_i} - V_{SVC_i}) - B_{SVC_i}) / T_{S_i} \quad (3)$$

where K_S , T_S are the gain and time constant of the SVC.

3 Multi-objective Evolutionary Algorithm

The GA approach is a population-based optimization technique in which a group of individuals, called a *population*, undergoes some evolutionary operations to find the best-fit individual and eventually the best-fit population. An *individual* (also called a chromosome or string) is used to represent the decision variables of the given design problem and to evaluate the objective function of the design problem. Then, a *fitness* value of an individual is calculated in terms of the objective function. More-fit individuals transmit their genetic properties to subsequent generations through their offspring. To create new child individuals, selection, crossover, and mutation operators are generally applied. Finally, over many generations, a population gradually evolves into a group of best-fit individuals.

3.1 Multi-objective Optimization Problem

MOEAs have been applied in various fields of science and engineering. In the electrical engineering fields [6,7]. And, these applications can be described as multiobjective optimization problems (MOPs), formulated as shown below.

$$\begin{aligned} \min_{\mathbf{x} \in \Omega} \mathbf{z} &= f_i(\mathbf{x}) \quad i = 1, \dots, N_{obj} \\ \text{s.t.} \quad \mathbf{x} &= (x_1 \quad x_2 \quad \dots \quad x_m) \in X \end{aligned} \tag{4}$$

where \mathbf{z} is called the objective vector, \mathbf{x} the decision vector and X the parameter space. A decision vector $\mathbf{a} \in X$ is said to dominate a decision vector $\mathbf{b} \in X$ if and only if

$$\begin{aligned} \forall i \in \{1 \quad 2 \quad \dots \quad k\}; \quad f_i(\mathbf{a}) \leq f_i(\mathbf{b}) \\ \text{and} \quad \exists j \in \{1 \quad 2 \quad \dots \quad k\}; \quad f_j(\mathbf{a}) < f_j(\mathbf{b}) \end{aligned} \tag{5}$$

If there is no vector in X that dominates the decision vector \mathbf{a} , then \mathbf{a} is said to be nondominated or Pareto optimal. The entire set of nondominated decision vectors is called the Pareto-optimal set and the corresponding solutions in the objective space are called the Pareto front. The goal of MOEA algorithms is to accurately represent the true Pareto front. A detail review on MOEA can be obtained from [5].

3.2 Pareto ranking and Fitness Sharing

To search Pareto-optimal solutions, the ranking selection method can be extended to identify the near Pareto-optimal set within the population of GA. Non-dominated solutions within the feasible region in the objective function space give the Pareto-optimal set. Consider an individual x^i at generation i which is dominated by p_i individuals in the current population. Its current position in the individuals' rank can be given by

$$\text{Rank}(x_i, t) = I + p_i \tag{6}$$

Then, all non-dominated individuals are assigned rank 1, and the smaller rank the higher probability to survive. It is commonly recognized that to maintain genetic diversity in the evolution process is important. However, random genetic drift could

happen sometimes. To avoid this, a practically effective scheme is to divide the raw fitness through by the accumulated value of shares as shown in (9).

$$f_s(x_i) = f(x_i) / \sum_j sh(d(x_i, x_j)) \tag{7}$$

where, $sh(d)$ is a sharing function that determines the neighborhood and the degree of sharing. It measures the distance between the design objective function values. The sharing function is given by the following equation.

$$Sh(d_{ij}) = \begin{cases} 1 - (d_{ij} / \sigma_{share})^\alpha, & \text{if } d_{ij} < \sigma_{share}; \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

Where, σ_{share} is the niche size and α is the exponent value. Then, the pseudocode of our MOEA adopted shows at Table 1.

Table 1. Pseudocode of MOEA

Step	operation
1	Generate population
2	Evaluate population
3	Do nondominated sorting
4	Do fitness sharing
5	while (* stopping criteria)
6	{Intentional blank} _a
7	Do selection/crossover/mutation
8	Evaluate population
9	Do nondominated sorting
10	Do fitness sharing
11	Update population
12	end while

4 Controller Design Method of PSS and SVC Via MOEA

The coordination design between PSS and SVC controllers is formulated as a nonlinear constrained multi-objective optimization problem. Input variables of the optimized function are controller parameters of PSS and SVC. The output values of the function are composed of the values of the performance measures. In MOEA, all parameters needed to optimize in the parameter set for the optimization problem is coded as finite-length string. For the PSS controller, the washout time constant T_w and T_2 are usually prespecified. The controller gain K_{PSS} , T_{IPSS} are to be determined. Moreover, the controller parameter K_s is also treated as the optimal parameters of SVC. The parameters take their values the following ranges.

$$0 \leq K_{pss_{1,2}} \leq 200 \quad 0 \leq T_{1pss_{1,2}} \leq 2 \quad 0 \leq K_s \leq 150 \tag{9}$$

A multiple performance index of integral square-error criterion is selected. Generator rotor angular speed differences are chosen as the feedback signals for the PSSs. In addition, the errors between the references value of SVC voltage and measured SVC voltage. Then, a multiple performance index of integral square-error criterion is pro-

posed here to indicate the dynamic performance of controllers. The fitness function for the multi-objective optimization is described as follows.

$$\text{Min} \begin{cases} F_{\omega_{12}}(K_{pss_1}, T_{1pss_1}, K_{pss_2}, T_{1pss_2}, K_S) = \int_0^{t_{sim}} |\Delta\omega_{12}| dt \\ F_{\omega_{13}}(K_{pss_1}, T_{1pss_1}, K_{pss_2}, T_{1pss_2}, K_S) = \int_0^{t_{sim}} |\Delta\omega_{13}| dt \\ F_{SVC}(K_{pss_1}, T_{1pss_1}, K_{pss_2}, T_{1pss_2}, K_S) = \int_0^{t_{sim}} |\Delta V_{SVC}| dt \end{cases} \quad (10)$$

where, t_{sim} denotes the time duration of the simulation.

5 Case Studies

The values of MOEA parameters adopted in the study are defined as: Number of generations is 50, Population size is 50, and Crossover probability and Mutation

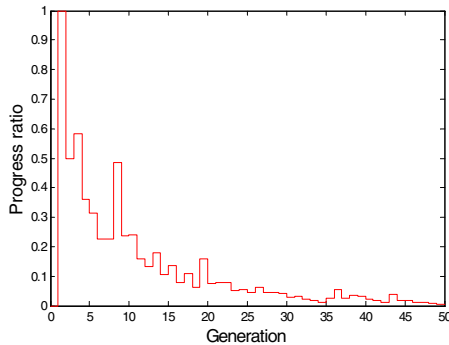


Fig. 2. Progress ratio of multi-objective evolution progress

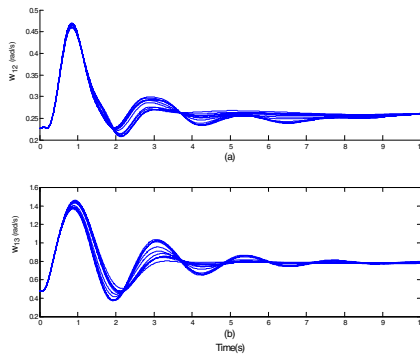


Fig. 3. (a) Relative rotor speed response between Gen 1 and Gen 2. (b) Relative rotor speed response between Gen 1 and Gen 3.

probability are 0.4 and 0.1 respectively. The controllers are designed under the three-phase short circuit occurring at bus 3 of line 3-101. The fault is cleared after 20ms. The controller parameters of PSS and SVC have been adjusted after fifty generations of optimization. Fig. 2 indicates the convergence property of the algorithm in the senses of progress ratio. The system response of rotor angular speed differences of all the PSSs optimally tuned by the Pareto-optimal solutions are shown in Fig. 3. It can be seen from the figure that the system responses in time domain are quite satisfied.

6 Conclusions

An optimization method Based MOEA for coordinated design of PSS and SVC is developed. With the method proposed, several design objectives can be simultaneously optimized through searching in the parameter space. Pareto optimal solutions are obtained from the genetic algorithm based on the ranking and the fitness sharing techniques. Simulation results demonstrate the effectiveness of the proposed design method and good joint operation performances of PSS and SVC are obtained. The proposed method is general and may be extended to include other power devices.

References

1. Hammad AE.: Analysis of Power System Stability Enhancement by Static VAR Compensators. *IEEE Trans PWRS*.1(1986) 222–7
2. Mahran AR, Hogg BW.: Coordinated Control of Synchronous Generator Excitation and Static VAR Compensator. *IEEE Trans Energy Conv.* 7(1992) 615–22
3. Rahim A, Nassimi S.: Synchronous Generator Damping Enhancement Through Cordinated Control of Exciter and SVC. *IEE Proc Gener Transm Distrib.*143(1996) 211–8
4. Yu YN.: *Electric Power System Dynamics*. New York Academic Press (1983)
5. Tamaki H., Kita H., Kobayashi S.: Multi-objective optimization by genetic algorithms: a review. *IEEE International Conference on Evolutionary Computation*. (1996) 517–522
6. Abido M.A.: Environmental/Economic Power Dispatch Using Multiobjective Evolutionary Algorithms. *IEEE Trans. on Power Systems*. 18(2003) 1529-1537
7. Milosevic B., Begovic M.: Nondominated Sorting Genetic Algorithm for Optimal Phasor Measurement Placement. *IEEE Trans. on Power Systems*. 18(2003) 69-75

Author Index

- Ahn, Dosung II-635
Aine, Sandip I-57
Akhtar, Saqlain I-528
Alagar, Vasu S. I-303
Allende, Héctor I-49
An, Bo I-375
Araki, Kenji I-434
Årnes, André II-388
Aritsugi, Masayoshi II-548
Austin, Francis R. I-902, II-649
- Bae, Duhyun II-439
Baek, Joong-Hwan I-729, II-899
Baek, Joon-sik I-729
Baek, Kyunghwan II-285
Bagis, Aytekin II-1042
Bagula, Antoine B. I-224
Bai, Ji II-826
Baik, Doo-Kwon II-725
Bao, Guanjun II-977
Bao, Shiqiang II-989
Bau, Y.T. I-657
Berg, Henrik I-17
Bhattacharya, Bhargab B. I-1057
Bhowmick, Partha I-1057
Bian, Shun I-809
Bin, Liu I-616
Bin, Shen I-704
Biswas, Arindam I-1057
Bo, Liefeng I-909
Bo, Yuan I-704
Bougaev, Anton I-991
Brekne, Tønnes II-388
Brown, Christopher L. II-80
Brown, Warick II-1074
Burkhardt, Hans II-820
Byun, Jin Wook II-143
- Cai, Wandong II-457
Cai, Yuanli I-410
Cai, Zhiping II-703
Cao, Chunhong I-145
Cao, Jian I-121, I-267, II-679
Cao, Lei II-679
- Cao, Tianjie II-1036
Cao, Wenming II-1068, II-1110
Cao, Yijia I-1106
Cao, Zhenfu II-104, II-167, II-445
Cao, Zhenyu I-267
Castiello, Ciro I-321
Chae, Kyu-Yeol II-947
Chai, Wanfang II-1068
Chakrabarti, P.P. I-57
Chau, K.W. I-645
Chang, Chinchun I-464, II-538, II-554, II-567
Chang, Elizabeth II-55, II-273
Chang, Hye-Young II-260
Chang, Nam Su II-1
Chang, Yilin I-864, I-1051, II-597
Chen, Chun I-941
Chen, Chunlin I-393
Chen, Enhong I-73
Chen, Gen-Cai I-941
Chen, Guimin I-663
Chen, Haibo I-758
Chen, Haixia I-145
Chen, Hua I-101
Chen, Jing II-838
Chen, Junying I-957
Chen, Licong II-629
Chen, Ling I-261, I-941
Chen, Liping II-629
Chen, Mingjun I-638
Chen, Mingyu II-322
Chen, Qingjiang II-1134
Chen, Qingliang I-349
Chen, SiBao I-1003
Chen, Tengbo I-651
Chen, Weihong II-74
Chen, Wensheng I-933
Chen, Xiaofeng II-117
Chen, Xuyang II-851
Chen, Yongqiang II-1017
Chen, Zhiguo II-433
Chen, Ziyi I-200
Chen, Zonghai I-393
Cheng, ChunTian II-421

- Cheng, Daijie I-375
 Cheng, Xiangguo II-49
 Cheng, Xiaochun I-381, I-422
 Cheung, Yiu-Ming I-184, I-965
 Chi, Yue II-415
 Cho, A-Young II-623
 Cho, Byeong Heon I-1064
 Cho, Ik-Hwan II-623
 Cho, SeongJe II-125, II-260
 Cho, Tae Ho II-340
 Cho, Yookun II-125
 Cho, Youn-Ho II-1025
 Choi, Jae Ho II-869
 Choi, Jongmoo II-125
 Choi, Jun I-313
 Choi, Jun Yong II-303, II-451
 Choi, Kyunghee II-297
 Choi, Soo-Mi II-1140
 Choi, Sung Jin II-494
 Choi, Won-Hyuck II-669
 Choi, WoongChul II-260
 Choi, Yoo-Joo II-1140
 Choudary Gorantla, M. II-110
 Chu, Xiuqin I-1009
 Chung, Kyoil II-1104
 Chung, Sang-Kyoon II-1104
 Chung, Yongwha II-635
 Coello Coello, Carlos A. I-208
 Cruz-Cortés, Nareli I-208
 Cui, Can I-864
 Cui, Shuning II-731
- Dagdelen, Ulvi II-1042
 Dai, Guozhong I-355, I-1100
 Dai, Qionghai II-863
 Dai, Xianhua I-927
 Dan, Hongwei I-752
 Dash, P.K. I-163
 de la Iglesia, B. II-1011
 Deng, Naiyang I-580, I-586
 Deng, Shengchun I-157, I-735
 Deng, Yafeng II-772
 Deng, Zhihong I-458
 Dhurandhar, Amit I-1021
 Ding, Guiguang II-863
 Ding, Jianli I-230
 Doh, Kyu B. II-780
 Doi, Hiroshi II-33
 Dong, Daoyi I-393
 Dong, Jinxiang II-965
- Dong, Lihua II-68
 Dong, Liyan I-145
 Dong, Ping II-27
 Dong, Won-Pyo II-947
 Dongliang, Wang I-616
 Du, Haifeng II-923
 Du, Ruizhong II-244
 Du, Tao I-151
 Duan, Haixin II-482, II-518
 Duan, Peijun I-381
 Duan, Shanshan II-445
 Duc, Duong Anh II-605
 Duthen, Yves I-329
- Ekinci, Murat I-973
 Engelbrecht, Andries P. I-192
 Eom, Young Ik II-1149
 Eun, Jongmin I-574
 Ewe, Hong Tat I-622, I-657
- Fan, Liangzhong II-935
 Fan, Wenbing II-838
 Fanelli, Anna Maria I-321
 Fang, Bin I-833, II-1060
 Fang, Binxing II-212, II-415
 Fang, Wei I-1015
 Fang, Yong I-1015
 Feng, Boqin I-747, II-731
 Feng, Dengguo II-995
 Feng, Huamin I-1015
 Feng, Xiang-chu II-1122
 Feng, Zhilin II-965
 Feng, Zhiyong I-878
 Fu, Guojiang I-638
 Fu, Peng II-1017
 Fu, Zetian I-598
 Fung, Chun Che II-1074
- Gan, John Q. I-495
 Gan, Xiaobing I-902
 Gao, Dengpan I-915
 Gao, Xiaojuan II-398
 Gao, Xinbo I-696, I-997
 Gao, Yuelin I-675
 Ge, Fei I-915
 Gedikli, Eyup I-973
 Georgiadis, Panagiotis II-589
 Gerardo, Bobby D. I-337, I-1082
 Ghose, Supratip I-470
 Gong, Maoguo I-793, I-846, I-858

- Gonzalez, Jesus A. I-41
 Gopalan, Srividya II-236
 Gu, Chunxiang II-9
 Gu, Lixu I-884, I-890
 Gu, Ming I-476
 Guan, Chun I-387
 Guo, Jun I-979
 Guo, Ping II-43
 Gustafsson, Lennart I-81
- Hahn, Hernsoo II-285
 Han, Bing I-997
 Han, Changcai II-1116
 Han, Dongfeng I-483
 Han, Jiuqiang I-133
 Han, Lixia I-297
 Han, Qi I-663
 Han, Song II-55
 Han, Sunyoung II-737
 Han, Youngjun II-285
 Hao, Yue II-796
 Haslum, Kjetil II-388
 He, Caisheng I-927
 He, Hui II-560
 He, Xiaoliang I-1088
 He, Xin I-381
 He, Yan II-463
 He, Zengyou I-157, I-735
 He, Zhaoyang II-965
 He, Zhenyu II-1060
 Hippisley, Andrew I-489
 Ho, Chang-Hoi I-503
 Ho, Anthony T.S. II-661
 Ho, Chin Kuan I-622, I-657
 Hong, Sungjune II-737
 Hong, Wei-Chiang I-512
 House, Terry C. II-719
 Hou, Yunshu II-875
 Hsu, Che-Chang I-550
 Hu, Fangming I-1009
 Hu, Heping I-442
 Hu, Hesuan I-1094
 Hu, Jun I-387
 Hu, Mingzeng II-560
 Hu, Ruijuan II-989
 Hu, S.X. II-917
 Hu, Xiaofeng II-804
 Hu, Yupu II-68
 Huang, Dezhi II-812
 Huang, Fenggang I-1039
- Huang, Houkuan I-9, I-33, I-741
 Huang, Huifeng II-554
 Huang, Jian I-933
 Huang, Jing I-604
 Huang, Jiwu II-573
 Huang, Mei-juan II-532
 Huang, Peiwei II-161
 Huang, T.Z. II-917
 Huh, Sung-Hoe I-503
 Huh, Sung-Hoe I-544
 Hui, Li II-180
 Huo, Hongwei I-821
 Hwang, Jae-jeong I-1082
- Imai, Sayaka II-548
- Jang, Heejun II-285
 Jang, Insook II-1149
 Jang, Jong Whan II-869
 Jang, MinSeok I-95
 Jawale, Rakesh I-1021
 Jeong, Chang-Sung II-845, II-947
 Jeong, Dong-Seok II-623
 Jeong, Dongwon II-725
 Ji, Dongmin I-127
 Ji, Hongbing I-556, I-997
 Ji, Ping I-405
 Ji, Shiming II-977
 Jia, Jianyuan I-663
 Jiang, Gangyi II-935
 Jiang, Quanyuan I-1106
 Jiang, Zhengtao II-1080
 Jiao, Licheng I-89, I-238, I-273,
 I-696, I-793, I-839, I-846, I-858,
 I-909
 Jiao, Yong-Chang I-247, I-651
 Jin, Andrew Teoh Beng II-788
 Jin, Hong I-1100
 Jin, Jae-Do II-1098
 Jin, Long I-721
 Jin, Yi-hui I-630
 Jing, Wenfeng I-139
 Jing, Yinan II-1017
 Jo, Geun-Sik I-470
 Jones, David Llewellyn I-1074
 Joo, Moon G. I-127
 Juárez-Morales, Raúl I-208
 Juneja, Dimple I-367
 Jung, Gihyun II-297
 Jung, In-Sung I-107

- Jung, Je Kyo I-216
 Jung, SeungHwan II-635
 Jung, Seung Wook II-86
 Jung, Sung Hoon I-1064
 Jung, Young-Chul II-743

 Kajava, Jorma II-508
 Kanamori, Yoshinari II-548
 Kang, Daesung I-65
 Kang, Daniel II-1127
 Kang, Geuntaek I-127
 Kang, Hyun-Ho II-581
 Kang, Hyun-Soo II-857
 Kang, Lishan I-200
 Kang, Sin-Jae I-361
 Kang, Yong-hyeog II-1149
 Karaboga, Dervis II-1042
 Karim, Asim I-170
 Kato, Nei II-252
 Kellinis, Emmanouel II-589
 Kim, Joo-Hong I-503
 Kim, Byung Ki II-303, II-451
 Kim, Chang Han II-1
 Kim, Cheol-Ki I-985
 Kim, Dong-Kyue II-1104
 Kim, Doo-Hyun II-669
 Kim, Gwanyeon II-439
 Kim, HongGeun II-260
 Kim, HoWon II-469, II-1104
 Kim, Hyun-Mi II-623
 Kim, Jangbok II-297
 Kim, Jeong Hyun II-749
 Kim, Jeong-Sik II-1140
 Kim, Jiho II-439
 Kim, Jin-Geol I-176
 Kim, Jinhyung II-725
 Kim, Jongho I-65
 Kim, Jong-Wan I-361
 Kim, Jung-Eun I-1082
 Kim, Jung-Sun II-669
 Kim, Kwang-Baek I-785, I-985
 Kim, Kyoung-Ho II-1098
 Kim, Min-Jeong II-1140
 Kim, Myoung-Hee II-1140
 Kim, Nam Chul I-799
 Kim, Sang Hyun I-799
 Kim, Sang-Kyoon II-635
 Kim, Seong-Whan II-643, II-1086
 Kim, Seong-Woo II-743
 Kim, Shin Hyoung II-869

 Kim, Sosun II-1
 Kim, Sungshin I-785
 Kim, Taehae II-635
 Kim, Tae-Yong II-857
 Kim, Weon-Goo I-95
 Kim, Wookhyun I-25
 Kim, Yong-Deak II-935
 Kim, Young Dae II-303, II-451
 Kim, Young-Tak II-743
 Knapskog, Svein Johan II-388
 Ko, Kwangsun II-1149
 Ko, Sung-Jea II-1098
 Kong, Jung-Shik I-176
 Kong, Min I-682, I-1003
 Koo, Han-Suh II-845
 Kraipeerapun, Pawalai II-1074
 Kudo, Daisuke II-252
 Kumar, Rajeev I-57
 Kwak, Jong In I-799
 Kwon, Goo-Rak II-1098
 Kwon, Ki-Ryong II-581
 Kwon, Taeck-Geun II-220
 Kwon, Taekyoung II-427

 Lai, Jianhuang I-933
 Lee, Bo-Hee I-176
 Lee, Byungil II-469
 Lee, Chong Ho I-216
 Lee, Deok-Gyu I-313
 Lee, Dong Hoon II-143
 Lee, Dong Wook II-749
 Lee, Gun-Woo II-911
 Lee, Jae-Wan I-337, I-1082
 Lee, Jeongjun I-127
 Lee, Jinseok II-1149
 Lee, JungSan II-538
 Lee, Kang-Woong II-899
 Lee, Kuhn-Il II-911
 Lee, Kwnag-Jae II-669
 Lee, Kyoung-Mi II-832
 Lee, Mun-Kyu II-1104
 Lee, Sang-Gul I-985
 Lee, Sang-Jin I-313
 Lee, Sang Jun II-303, II-451
 Lee, Seon-Gu I-544
 Lee, Seung Phil II-869
 Lee, Seungwon II-125
 Lee, Songwook I-574
 Lee, Soon-tak I-729
 Lee, Tai Sik II-749

- Lee, Tea-Young II-1098
 Lee, Wonchang I-127
 Lee, Woobeom I-25
 Lee, Yong-min I-216
 Lee, Yoon Cheol II-33
 Lee, Youngseok II-220
 Lei, Qiang II-1080
 Leung, Ho-fung II-1030
 Li, Bicheng II-764
 Li, Bo II-291
 Li, Changjun II-655
 Li, Chunping I-713
 Li, Chunzhong I-592
 Li, Chuyu I-721
 Li, Congcong II-772
 Li, Fengxiang I-1045
 Li, Fuhai II-812
 Li, Gang I-405
 Li, Haijun I-145
 Li, Haiquan I-405
 Li, Hong I-247
 Li, Hongwei II-941
 Li, HuiXian II-421
 Li, Jianhua II-526
 Li, Jianqing II-1110
 Li, Jie I-696
 Li, Jingtao II-1017
 Li, Kenli II-463
 Li, Lingjuan II-334
 Li, Liping II-381
 Li, Ming II-267
 Li, Minglu II-679
 Li, Ning I-638
 Li, Qianmu II-542
 Li, Qunxia I-979
 Li, Renfa II-463
 Li, Rui II-941
 Li, Wei II-291
 Li, Wenhui I-483
 Li, Wenjie I-878
 Li, Xiang II-173
 Li, Xiangxue II-104
 Li, Xing II-518
 Li, Xinghua II-356
 Li, Yajing II-74
 Li, Yao II-309
 Li, Yushan I-1009
 Li, Zhanchun II-309
 Li, Zhitang II-309, II-611
 Li, Zhiwu I-1094
 Liang, Xiaoping II-573
 Liao, Jian II-161
 Liao, Z.W. II-917
 Lim, Chan II-857
 Lim, Jongin II-1, II-143
 Lim, Sukhyun I-827
 Lin, Yongmin I-741
 Lin, Chiachen I-464, II-567
 Lin, Chih-Sheng I-512
 Lin, Dongdai II-131, II-375, II-1036
 Lin, Ling I-489
 Lin, Mengquan II-526
 Lin, Shukuan I-520
 Lin, Xuegang II-407
 Ling, David Ngo Chek II-788
 Linshu, He I-528
 Liotta, Antonio I-489
 Liu, Bin II-309, II-611
 Liu, Bo I-630
 Liu, Chi II-703
 Liu, Dalian I-184
 Liu, Fang I-793, I-852
 Liu, Fenglei I-410
 Liu, Fengyu II-542
 Liu, Gang I-416, I-979
 Liu, Jinfeng I-688
 Liu, Jing I-238
 Liu, Jingmei II-49
 Liu, Keqin II-711
 Liu, Liangjiang I-609
 Liu, Pengfei I-133
 Liu, Qin II-695
 Liu, Sen I-1015
 Liu, Shengli II-117
 Liu, Shizhu I-442
 Liu, Weiping II-457
 Liu, Wu II-482, II-518
 Liu, Ying I-902
 Liu, Yu I-957
 Liu, Zaiqiang II-995
 Lu, Bin I-846
 Lu, Jiang I-957
 Lu, Jiangang I-921
 Lu, Zhe-Ming II-820
 Luan, Shangmin I-355
 Luo, Bin I-1003
 Luo, Xiaohua II-687
 Luo, Yunlun II-43
 Lv, Kewei II-96
 Lv, Lintao I-747

- Lv, Shaohe II-703
 Lv, Tianyang I-771
- Ma, Jianfeng II-228, II-356, II-476,
 II-488
 Ma, Jinwen I-915, II-812
 Ma, Jun I-101
 Ma, Linru II-328
 Ma, Lizhuang II-826
 Ma, Long II-655
 Ma, Rui I-1033
 Ma, Wenping I-793
 Ma, Xiaoqi I-381, I-422
 Maik, Vivek II-929
 Marias, Giannis F. II-589
 Mateo, Romeo Mark A. I-337
 Meng, Deyu I-139
 Meng, Hongyun I-839
 Meng, Kai II-772
 Meng, Zhiqing I-267, I-568
 Miao, Chunyan I-375
 Miao, Yuan I-375
 Min, Yao I-704
 Ming, Hua I-821
 Mitchell, Chris J. II-149, II-192
 Moe, Marie Elisabeth Gaup II-388
 Mogami, Yoshio I-115
 Moon, Daesung II-635
 Moon, Kiyong II-635
 Moon, SangJae II-356, II-488
 Moraga, Claudio I-49
 Morii, Masakatu II-17
 Mu, Chengpo I-9
- Nah, Won II-899
 Ñanculef, Ricardo I-49
 Naqvi, Syed II-348
 Nemoto, Yoshiaki II-252
 Ni, Zhonghua I-870
 Nin, Jordi I-1
 Ning, Yufu I-230
 Nishiura, Yasumasa I-669
- Oh, Sang-Hun II-1025
 Oh, Ha Ryoung I-1064
 Ohigashi, Toshihiro II-17
 Ohn, Jungho II-780
 Okamoto, Eiji II-198
 Okamoto, Takeshi II-198
 Olmos, Ivan I-41
- Olsson, Roland I-17
 Omran, Mahamed G.H. I-192
 Osorio, Mauricio I-41
- Pai, Ping-Feng I-512
 Paik, Joonki II-929
 Pan, Rihong II-629
 Pan, Xiaoying I-852
 Pan, Yun I-285
 Pan, Yunhe I-752, II-687
 Pancake, Cherri M. II-548
 Panda, G. I-163
 Pang, LiaoJun II-421
 Pang, Sulin I-1027
 Pang, Ying-Han II-788
 Panigrahi, B.K. I-163
 Papapanagiotou, Konstantinos II-589
 Papiński, Andrew P. I-81
 Park, Jong-Hyuk I-313
 Park, Seon-Ki I-503
 Park, Chang Won II-494
 Park, Hye-Ung I-313
 Park, Ji-Hyung I-450, II-204
 Park, Jooyoung I-65
 Park, Kyu-Sik II-1025
 Park, Sang-ho II-427
 Park, Sehyun II-439
 Park, Young-Ho II-1
 Park, Young-Ran II-581
 Patnaik, Lalit M. I-562
 Pei, Hui II-820
 Peng, Changgen II-173
 Peng, Jing I-669
 Peng, Lifang I-568
 Peng, Qinke II-315
 Peters, Terry I-884
 Poon, Ting-C II-780
 Potdar, Vidyasagar II-273
 Puhan, Niladri B. II-661
 Purevji, Bat-Odon II-548
- Qi, Yaxuan I-1033
 Qi, Yinghao II-161
 Qi, Zhiquan I-580
 Qian, Haifeng II-104
 Qian, Yuntao II-757
 Qin, Zheng I-957
 Qing, Sihan II-381
 Qu, Youli I-741

- Rayward-Smith, V.J. II-1011
 Riguidel, Michel II-348
 Rodríguez-Henríquez, Francisco I-208
 Rong, Gang I-688
 Rong, Mentian II-161
 Rong, Xiaofeng II-398
 Ruland, Christoph II-86
 Ryu, Jeha I-949
 Ryu, Keun Ho I-721
- Sachidananda, Saraswathi II-236
 Sallhammar, Karin II-388
 Salman, Ayed I-192
 Samantaray, S.R. I-163
 Sanza, Cédric I-329
 Savola, Reijo II-508
 Saxena, Ashutosh II-110
 Sbert, Mateu II-989
 Schmucker, Martin II-80
 Seo, Hee Suk II-340
 Seo, Jungyun I-574
 Seo, Kang-hee I-827
 Seo, Seong Chae II-303, II-451
 Seo, Sungbo I-721
 Seol, Jae-Min II-1086
 Seong, Yeong Rak I-1064
 Shang, Ronghua I-846
 Shang, Wenqian I-741
 Shankarnarayanan, Kartik I-1021
 Sharath, S. I-562
 Sharma, A.K. I-367
 She, Yanjie I-483
 Shen, Haifeng I-979
 Shen, Jianjun II-381
 Shen, Lincheng I-399
 Shen, Qingni II-381
 Shen, Zhiqi I-375
 Shen, Zhong I-864
 Shi, Feng I-604
 Shi, Guangming II-851
 Shi, Xiangdong II-27
 Shi, Yi II-364
 Shim, Jaehong II-297
 Shim, Kwang-Hyun I-1064
 Shin, Byeong-Seok I-827, II-1127
 Shin, Jeongho II-929
 Shin, Sang-Uk II-581
 Shiraishi, Yoshiaki II-17
 Sim, Kwee-Bo I-428
 Skinner, Geoff II-55
- Song, Hee-Jun I-503, I-544
 Song, Il-Seop II-220
 Song, Li II-983
 Song, Ohyoung II-439
 Song, Shaoxu I-713
 Song, Shuni II-655
 Song, Wenchao I-663
 Srinivasa, K.G. I-562
 Su, Guangda II-772
 Su, Han I-1039
 Su, Kaile I-349
 Su, Ruidan II-398
 Su, Yong I-416
 Suk, Jung-Youp II-911
 Sun, Baolin I-101
 Sun, Fan I-536
 Sun, Jiaguang I-476, II-959
 Sun, Linyan I-405
 Sun, Maosong I-536
 Sun, Ninghui II-322
 Sun, Suqin I-965
 Sun, Xi II-500
 Sun, Xiaojuan II-322
 Sun, Youxian I-921
 Sun, Yu II-1048
 Sun, Yuxiang II-155
 Sun, Zengqi I-255
 Sung, Hyun-Sung II-643
- Takagi, Tsuyoshi II-198
 Taleb-Bendiab, A. I-1074
 Tang, Qiang II-149, II-192
 Tang, Shaohua II-186
 Tang, Shugang I-878
 Tang, Wansheng I-230
 Tang, Wenyu II-334
 Tang, Yuan Yan I-833, II-1060
 Tao, Xiaoyan I-556
 Thapa, Devinder I-107
 Tian, Fengzhan I-33
 Tian, Haibo II-500
 Tian, Jing I-399
 Tian, Junfeng II-244
 Tian, Peng I-682
 Tian, Shengfeng I-9, I-33
 Tian, Yingjie I-580, I-586
 Tikkanen, Antti II-1054
 Ting, Shen II-180
 Torra, Vicenç I-1
 Tran, Trung Hau I-329

- Triet, Tran Minh II-605
 Tu, Li I-261

 Urmanov, Aleksey I-991

 Valle, Carlos I-49
 Varadarajan, Sridhar II-236
 Varonen, Rauno II-508
 Venugopal, K.R. I-562
 Virtanen, Teemupekka II-1054

 Waizumi, Yuji II-252
 Wan, Shuai I-1051, II-597
 Wan, Yuehua II-977
 Wang, Zhihai I-741
 Wang, BaoBao I-343
 Wang, Ben I-495
 Wang, Changguang II-476
 Wang, Changjie II-1030
 Wang, Dayin II-375
 Wang, Fangwei II-476
 Wang, Fasong II-941
 Wang, Feng I-73
 Wang, Gi-Nam I-107
 Wang, Guoyuo II-893
 Wang, Hai I-765
 Wang, Hongan I-1100
 Wang, Hong F. I-224
 Wang, Hui I-1100
 Wang, Jianxin II-328
 Wang, Jiaxin I-1033
 Wang, Jin I-216
 Wang, Jinlong I-752
 Wang, Junping II-796
 Wang, Lei II-959
 Wang, Licheng I-285, II-104
 Wang, Lihua II-198
 Wang, Limin I-145
 Wang, Ling I-630, I-909
 Wang, Long I-422
 Wang, Ping II-212
 Wang, Qiangmin II-526
 Wang, Qin II-167
 Wang, Rangding II-935
 Wang, Ruchuan II-334
 Wang, Shaomei I-638
 Wang, Shoujue II-1068, II-1110
 Wang, Sun'an II-923
 Wang, Wei I-765
 Wang, Weidong I-1094

 Wang, Weixing II-887
 Wang, Wenjia I-809
 Wang, Wentao II-893
 Wang, Xiangyang II-617
 Wang, Xiaomo I-483
 Wang, Xinmei II-49
 Wang, Yaonan I-609
 Wang, Ye I-343
 Wang, Yingluo I-291
 Wang, Youqing II-1003
 Wang, Yuexuan I-870
 Wang, Yumin II-500, II-1080
 Wang, Yuping I-184, I-247, I-297
 Wang, Zhanquan I-758
 Wang, Zhengxuan I-771
 Wang, Zhiling II-137
 Wang, Zhiquan II-881
 Wang, Zhurong I-747
 Wang, Zongjiang I-151
 Wei, Jinwu II-971
 Wei, Ping I-815
 Wei, Xiaopeng I-896
 Wei, Zhiqiang I-279
 Wen, Fengtong II-62
 Wen, Qiaoyan II-62
 Wenjun, Zhang II-983
 Woo, Woontack I-949
 Wu, Chanle II-695
 Wu, Chen II-273
 Wu, Jiangxing II-971
 Wu, Jianping II-482, II-518
 Wu, Lijun I-349
 Wu, Mingni I-464
 Wu, Wenling II-62, II-375
 Wu, Xiaoyun II-573
 Wu, Yanling I-921
 Wu, Zhaohui II-687

 Xiang, Youjun II-953
 Xie, Bo I-941
 Xie, Shengli II-953
 Xie, Shu-cui II-532
 Xie, Xuemei II-851
 Xie, Yi II-977
 Xin, Zhiyun I-476
 Xiong, Hongkai II-983
 Xixiang, Lv II-1092
 Xu, Benlian II-881
 Xu, Chen I-139, II-649
 Xu, Chengxian I-675, I-815, I-1088

- Xu, Congfu I-752
 Xu, Jianfeng I-884, I-890
 Xu, Jin An I-434
 Xu, Manwu II-542
 Xu, Qijian II-1048
 Xu, Qing II-989
 Xu, Rongsheng II-407
 Xu, Wenli II-863
 Xu, Xiaofei I-157, I-735
 Xu, Yong II-155
 Xu, Zhiliang II-953
 Xu, Zongben I-777
 Xue, Rui II-1036

 Yan, Wei II-279
 Yang, Bo II-137, II-1092
 Yang, Chan-Yun I-550
 Yang, Fuzheng I-1051, II-597
 Yang, Jiehui II-27
 Yang, Jr-Syu I-550
 Yang, Lin II-328
 Yang, Ping I-598
 Yang, Qiu I-101
 Yang, Tao II-463
 Yang, Xinyu II-364
 Yang, Yongdong II-1122
 Yang, Yueting I-675
 Yang, Zhi II-560
 Yao, Li II-826
 Yao, Tingting II-826
 Yao, Zhiqiang II-629
 Ye, Feng I-121
 Ye, Qingtai II-804
 Ye, Xien II-935
 Yeh, Junbin II-567
 Yeom, Ki-Won I-450, II-204
 Yi, Hong I-870
 Yi, Hong Yan II-1011
 Yin, Bo I-279
 Yin, Hujun II-764
 Yin, Jianping II-703
 Yin, Jianwei II-965
 Yin, Qian II-43
 Yoon, Han-Ul I-428
 You, Jin Ho II-303, II-451
 You, Xinge I-833, II-1060
 You, Xinhua II-1060
 Youn, Hee Yong II-494
 Yu, Hang I-858
 Yu, Jian I-33

 Yu, Mei II-935
 Yu, Mengjie I-1074
 Yu, Peifei I-921
 Yu, Shaoquan II-941
 Yu, Shengsheng I-416
 Yu, Xinbo I-1045
 Yu, Zhenhua I-410
 Yu, Zhenwei I-285
 Yuan, Dongfeng II-1116
 Yuan, Qiaoling II-977
 Yuan, Xiaohui I-669
 Yuan, Yubo I-592
 Yuen, Pong C. I-933
 Yun, Jaeseok I-949
 Yun, Xiaochun II-212, II-415

 Zeng, Yong II-68, II-228
 Zengke, Zhang I-616
 Zhan, Daqi I-965
 Zhanchun, Li II-611
 Zhang, Bin I-858
 Zhang, Changshui I-1033
 Zhang, Chenbin I-393
 Zhang, Cheng II-315
 Zhang, Dan II-1060
 Zhang, Dongdong II-983
 Zhang, Fan II-433, II-488
 Zhang, Fangguo II-117
 Zhang, Fushun I-651
 Zhang, Gendu II-1017
 Zhang, Hong II-482, II-542
 Zhang, Hongli II-560
 Zhang, Ji I-765
 Zhang, Jian I-713
 Zhang, Jian-zhong II-532
 Zhang, Jie II-1048
 Zhang, Jinhua II-923
 Zhang, Li II-977
 Zhang, Ling II-905
 Zhang, Ming I-458
 Zhang, Naitong II-1048
 Zhang, Pengxiang I-1106
 Zhang, Qiang I-896
 Zhang, Rui I-777, II-989
 Zhang, Shaomin I-520
 Zhang, S.S. I-151
 Zhang, Tongqi II-1134
 Zhang, Weiguo I-291
 Zhang, Weiqiang II-649
 Zhang, Weizhe II-560

- Zhang, Xian-Da II-905
Zhang, Xiaohua I-839
Zhang, Xiaoshuan I-598
Zhang, Xin I-864
Zhang, Xinhong II-433
Zhang, Xizhe I-771
Zhang, Yajuan II-9
Zhang, Yanning II-875
Zhang, Ying II-826
Zhang, Yongzheng II-415
Zhang, Yunkai II-476
Zhang, Yuqing II-137
Zhao, Dexin I-878
Zhao, Fengji I-815
Zhao, Hong II-617
Zhao, Jizhong I-476
Zhao, Jun I-273
Zhao, Liyuan II-711
Zhao, Rongchun II-875
Zhao, Shuguang I-273
Zhao, Wei II-267
Zhao, Weidong II-244
Zhao, Wencang I-1045
Zhen, Jina II-838
Zheng, Jin II-291
Zheng, Jun II-711
Zheng, Kougen II-687
Zheng, Mao I-303
Zheng, Qiuhua II-757
Zheng, Yanxing I-399
Zhi, Lijia I-520
Zhong, Hua I-89
Zhong, Shaochun I-381, I-422
Zhong, Weicai I-238
Zhou, Bai-Tao II-899
Zhou, Bing II-291
Zhou, Dongdai I-381
Zhou, Donghua II-1003
Zhou, Dongsheng I-896
Zhou, Gengui I-121, I-267, I-568
Zhou, Hongfang I-747
Zhou, Jingli I-416
Zhou, Jun II-772
Zhou, Lihua II-398
Zhou, Shude I-255
Zhou, Sujing II-131
Zhou, Xinliang II-322
Zhou, Xuebing II-80
Zhou, Yong II-804
Zhou, Yuan II-445
Zhu, Chuanchao I-1088
Zhu, Haibin I-741
Zhu, Huayong I-399
Zhu, Miaoliang II-407
Zhu, Yihua I-568
Zhu, Yuefei II-9
Zhuang, Chuanli I-598
Zhuang, Jian II-923
Zhuang, Xiahai I-884, I-890
Zhuang, Xiaodong I-279
Zou, Beiji II-959
Zou, Boxian II-629
Zou, Zhenyu I-1106
Zuo, Wanli I-771
Zuo, Wuheng II-965