Ying Tan
Yuhui Shi
Hongwei Mo (Eds.)

LNCS 7929

Advances in Swarm Intelligence

4th International Conference, ICSI 2013
Harbin, China, June 2013
Proceedings, Part II

2 Part II

ICSI

Springer

Lecture Notes in Computer Science 7929

Ying Tan Yuhui Shi Hongwei Mo (Eds.)

Advances in Swarm Intelligence

4th International Conference, ICSI 2013
Harbin, China, June 12-15, 2013
Proceedings, Part II

Springer

Volume Editors

Ying Tan
Peking University, Key Laboratory of Machine Perception (MOE)
School of lectronics Engineering and Computer Science
Department of Machine Intelligence
Beijing 100871, China
E-mail: ytan@pku.edu.cn

Yuhui Shi
Xi'an Jiaotong-Liverpool University
Department of Electrical and Electronic Engineering
Suzhou 215123, China
E-mail: yuhui.shi@xjtlu.edu.cn

Hongwei Mo
Harbin Engineering University, Automation College
Harbin 150001, China
E-mail: mhonwei@163.com

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-38714-2 e-ISBN 978-3-642-38715-9
DOI 10.1007/978-3-642-38715-9
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013939062

CR Subject Classification (1998): F.2, G.1, G.2, H.2.8, I.2.6, F.1, I.2, H.4, I.4, I.5

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

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

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This book and its companion volume, LNCS vols. 7928 and 7929, constitute the proceedings of the 4th International Conference on Swarm Intelligence (ICSI 2013) held during June 12–15, 2013, in Harbin, China. ICSI 2013 was the fourth international gathering in the world for researchers working on all aspects of swarm intelligence, following the successful and fruitful Shenzhen (ICSI 2012), Chongqing (ICSI 2011), and Beijing events (ICSI 2010), which provided a high-level academic forum for the participants to disseminate their new research findings and discuss emerging areas of research. It also created a stimulating environment for the participants to interact and exchange information on future challenges and opportunities in the field of swarm intelligence research.

ICSI 2013 received 268 submissions from about 613 authors in 35 countries and regions (Algeria, Australia, Austria, Bangladesh, Bonaire Saint Eustatius and Saba, Brazil, Canada, Chile, China, Czech Republic, France, Germany, Hong Kong, India, Islamic Republic of Iran, Italy, Japan, Republic of Korea, Malaysia, Mexico, Pakistan, Palestine, Romania, Russian Federation, Saudi Arabia, Singapore, South Africa, Spain, Sweden, Switzerland, Chinese Taiwan, Thailand, Tunisia, Turkey, UK, USA) across six continents (Asia, Europe, North America, South America, Africa, and Oceania). Each submission was reviewed by at least two reviewers, and on average 2.5 reviewers. Based on rigorous reviews by the Program Committee members and reviewers, 129 high-quality papers were selected for publication in this proceedings volume with an acceptance rate of 48.13%. The papers are organized in 22 cohesive sections covering all major topics of swarm intelligence research and development.

As organizers of ICSI 2013, we would like to express sincere thanks to Harbin Engineering University, Peking University, and Xian Jiaotong-Liverpool University for their sponsorship, as well as to the IEEE Computational Intelligence Society, World Federation on Soft Computing, and International Neural Network Society for their technical co-sponsorship. We appreciate the Natural Science Foundation of China for its financial and logistic support. We would also like to thank the members of the Advisory Committee for their guidance, the members of the International Program Committee and additional reviewers for reviewing the papers, and the members of the Publications Committee for checking the accepted papers in a short period of time. Particularly, we are grateful to the Springer for publishing the proceedings in the prestigious series of *Lecture Notes in Computer Science*. Moreover, we wish to express our heartfelt appreciation to the plenary speakers, session chairs, and student helpers. In addition, there are still many more colleagues, associates, friends, and supporters who helped us in immeasurable ways; we express our sincere gratitude to them all. Last but not the least, we would like to thank all the speakers, authors, and participants for their great contributions that made ICSI 2013 successful and all the hard work worthwhile.

April 2013

Ying Tan
Yuhui Shi
Hongwei Mo

Organization

General Chairs

Russell C. Eberhart	Indiana University-Purdue University, USA
Guihua Xia	Harbin Engineering University, China
Ying Tan	Peking University, China

Program Committee Chair

Yuhui Shi	Xi'an Jiaotong-Liverpool University, China

Advisory Committee Chairs

Gary G. Yen	Oklahoma State University, USA
Xingui He	Peking University, China

Organizing Committee Chair

Hongwei Mo	Harbin Engineering University, China

Technical Committee Chairs

Carlos A. Coello Coello	CINVESTAV-IPN, Mexico
Xiaodong Li	RMIT University, Australia
Andries Engelbrecht	University of Pretoria, South Africa
Ram Akella	University of California, USA
M. Middendorf	University of Leipzig, Germany
Lin Zhao	Harbin Engineering University, China

Special Sessions Chairs

Fernando Buarque	University of Pernambuco, Brazil
Benlian Xu	Changsu Institute of Technology, China

Publications Chair

Radu-Emil Precup	Politehnica University of Timisoara, Romania

Publicity Chairs

Hideyuki Takagi	Kyushu University, Japan
Shan He	University of Birmingham, UK
Yew-Soon Ong	Nanyang Technological University, Singapore
Juan Luis Fernandez Martinez	University of Oviedo, Spain
Jose Alfredo F. Costa	Federal University, Brazil
Kejun Wang	Harbin Engineering University, China

Finance and Registration Chairs

Chao Deng	Peking University, China
Andreas Janecek	University of Vienna, Austria

Local Arrangements Chairs

Lifang Xu	Harbin Engineering University, China
Mo Tang	Harbin Engineering University, China

Program Committee

Payman Arabshahi	University of Washington, USA
Sabri Arik	Istanbul University, Turkey
Carmelo J. A. Bastos Filho	University of Pernambuco, Brazil
Walter Chen	National Taipei University of Technology, Chinese Taipei
Manuel Chica	European Centre for Soft Computing, Spain
Jose Alfredo Ferreira Costa	UFRN Universidade Federal do Rio Grande do Norte, Brazil
Arindam K. Das	University of Washington, USA
Prithviraj Dasgupta	University of Nebraska, USA
Mingcong Deng	Tokyo University of Agriculture and Technology, Japan
Yongsheng Ding	Donghua University, China
Haibin Duan	Beijing University of Aeronautics and Astronautics, China
Mark Embrechts	Rensselaer Institute, USA
Juan Luis Fernández Martínez	University of Oviedo, Spain
Wai-Keung Fung	University of Manitoba, Canada
Luca Gambardella	Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Switzerland

Dunwei Gong	China University of Mining and Technology, China
Maoguo Gong	Xidian University, China
Ping Guo	Beijing Normal University, China
Haibo He	University of Rhode Island, USA
Ran He	National Laboratory of Pattern Recognition, China
Shan He	University of Birmingham, UK
Lu Hongtao	Shanghai Jiao Tong University, China
Mo Hongwei	Harbin Engineering University, China
Jun Hu	Chinese Academy of Sciences, China
Guangbin Huang	Nanyang Technological University, Singapore
Yuancheng Huang	Wuhan University, China
Andreas Janecek	University of Vienna, Austria
Alan Jennings	University of Dayton, USA
Zhen Ji	Shenzhen University, China
Changan Jiang	RIKEN-TRI Collaboration Center for Human-Interactive Robot Research, Japan
Licheng Jiao	Xidian University, China
Colin Johnson	University of Kent, USA
Farrukh Khan	FAST-NUCES Islamabad, Pakistan
Thanatchai Kulworawanichpong	Suranaree University of Technology, Thailand
Germano Lambert-Torres	Itajuba Federal University, Brazil
Xia Li	Shenzhen University, China
Xuelong Li	University of London, UK
Andrei Lihu	Politehnica University of Timisoara, Romania
Fernando B. De Lima Neto	University of Pernambuco, Brazil
Guoping Liu	University of Glamorgan, UK
Jianhua Liu	Fujian University of Technology, China
Ju Liu	Shandong University, China
Wenlian Lu	Fudan University, China
Bernd Meyer	Monash University, Australia
Martin Middendorf	University of Leipzig, Germany
Bijaya Ketan Panigrahi	Indian Institute of Technology, Delhi, India
Thomas Potok	ORNL, USA
Radu-Emil Precup	Politehnica University of Timisoara, Romania
Yuhui Shi	Xi’an Jiaotong-Liverpool University, China
Zhongzhi Shi	Institute of Computing Technology, Chinese Academy of Sciences, China
Mohammad Taherdangkoo	Shiraz University, Iran
Hideyuki Takagi	Kyushu University, Japan
Ying Tan	Peking University, China
Ke Tang	University of Science and Technology of China, China

Ba-Ngu Vo	Curtin University, Australia
Bing Wang	University of Hull, UK
Jiahai Wang	Sun Yat-sen University, China
Lei Wang	Tongji University, China
Ling Wang	Tsinghua University, China
Lipo Wang	Nanyang Technological University, Singapore
Qi Wang	Xi'an Institute of Optics and Precision Mechanics Of CAS, China
Shunren Xia	Zhejiang University, China
Benlian Xu	Changshu Institute of Technology, China
Yingjie Yang	De Montfort University, UK
Peng-Yeng Yin	National Chi Nan University, China
Zhuhong You	Shenzhen University, China
Jie Zhang	Newcastle University, UK
Jun Zhang	Waseda University, Japan
Junqi Zhang	Tongji University, China
Lifeng Zhang	Renmin University of China, China
Qieshi Zhang	Waseda University, Japan
Qingfu Zhang	University of Essex, UK
Dongbin Zhao	Institute of Automation, Chinese Academy of Science, China
Zhi-Hua Zhou	Nanjing University, China
Zexuan Zhu	Shenzhen University, China
Xingquan Zuo	Beijing University of Posts and Telecommunications, China

Additional Reviewers

Ali, Aftab
Bo, Xing
Bova, Nicola
Dai, Wang-Zhou
Ding, Ke
Ding, Ming
Fang, Jianwu
Gambardella, Luca Maria
Hao, Pengyi
Ho, Tze-Yee
Jiesheng, Wang
Mi, Guyue
Pei, Yan
Pérez Pancho, David
Qian, Chao
Sun, Minghui
Tanoto, Andry
Wan, Wenbo
Wang, Jaidong
Wang, Li
Xing, Bo
Yeh, Ming-Feng
Yu, Chao
Yu, James
Yu, Jian
Zhang, Pengtao
Zheng, Shaoqiu
Zheng, Zhongyang
Zhou, Wei
Zhu, Guokang

Table of Contents – Part II

Hybrid Algorithms

Swarm-Robot and Multi-agent Systems

Support Vector Machines

Data Mining Methods

System and Information Security

Intelligent Control

Wireless Sensor Network

Scheduling and Path Planning

Image and Video Processing

Other Applications

Table of Contents – Part I

Analysis of Swarm Intelligence Based Algorithms

Particle Swarm Optimization

Applications of PSO Algorithms

Ant Colony Optimization Algorithms

Biogeography-Based Optimization Algorithms

Novel Swarm-Based Search Methods

Bee Colony Algorithms

Differential Evolution

Parameter Optimization

Neural Networks

Fuzzy Methods

Evolutionary Programming and Evolutionary Games

Hybrid Gravitational Search and Clonal Selection Algorithm for Global Optimization

Shangce Gao,[1] Hongjian Chai,[1] Beibei Chen[1], and Gang Yang[2]

[1] College of Information Science and Technology, Donghua University, China
gaosc@dhu.edu.cn
[2] School of Information, Renmin University of China

Abstract. In recent years, there has been a growing interest in algorithms inspired by the behaviors of natural phenomena. However, the performance of any single pure algorithm is limited by the size and complexity of the problem. To further improve the search effectiveness and solution robustness, hybridization of different algorithms is a promising research direction. In this paper, we propose a hybrid iteration algorithm by combing the gravitational search algorithm with the clonal selection. The gravitational search performs exploration in the search space, while the clonal selection is implemented to carry out exploitation within the neighborhood of the solutio found by gravitational search. The emerged hybrid algorithm, called GSCSA, thus reasonably combines the characteristics of both base algorithms. Experimental results based on several benchmark functions demonstrate the superiority of the proposed algorithm in terms of solution quality and convergence speed.

Keywords: gravitational search, clonal selection, hybridization.

1 Introduction

Considering a general global numerical optimization problem formulated as minimizing $f(X)$, where $X = (x^1, x^2, ..., x^D) \in S$ is the vector of decision variables to be optimized, each variable x^i is bounded by $[l^i, u^i]$, $S \subseteq \prod_{i=1}^{D}[l^i, u^i]$ is an D-dimensional rectangular space in $\mathcal{R}^D$. Numerous global optimization problems are arising from almost every field of the scientific communities, such as industrial engineering, mechanical engineering, business, biological pharmacy, etc. However, many difficulties involving multimodality, dimensionality and non-differentiability are often associated with the above optimization problems. Classical techniques like steepest decent, linear programming, dynamic programming generally fail to solve such problems since most of them require gradient information and easily get trapped into local optima. So there remains a need for efficient and effective optimization techniques.

Over the last few decades, nature-inspired meta-heuristic optimization techniques are proving to be better than the classic techniques and thus are widely used. The more notable developments have been the evolutionary algorithm (EA) [1,2] inspired by neo-Darwinian theory of evolution, the artificial immune

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 1–10, 2013.

system (AIS) [3] inspired by biological immune principles, the swarm intelligence (SI) [4–6] inspired by social behavior of gregarious insects, and the gravitational search algorithm (GSA) inspired by the Newtonian gravity law [7,8], etc. These algorithms have been applied to many engineering problems and demonstrated promising performance on solving some specific kind of problems. However, as noticed in [9] that there is no specific algorithm to achieve the best solution for all optimization problems, all optimization techniques have some limitations in one or the other aspect. Thus, more research is required to enhance the existing algorithms to suit particular applications. Modification of the search mechanisms or hybridization of the existing algorithms are two alternative means to enhance the performance of the algorithm. Enhancement can be done by modifying the algorithm from many aspects: maintaining the population diversity [10], reducing or self-adapting the user-specific parameters [11,12], incorporating quantum or chaotic encoding strategies [13,14], or utilizing hierarchal cooperative learning mechanisms [15,16], etc. Hybridization of different search mechanisms from two algorithms has recently received much attention. The emerged algorithm is expected to utilize beneficial solutions found by each base algorithm, therefore resulting in a faster convergence speed or more precise solutions for the optimization problem [17].

In this paper, an attempt of combining clonal selection into the traditional GSA is presented. Inspired by one of the physical phenomena, i.e. Newtonian gravity law, GSA manipulates a set of agents in the universe, in which every particle attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them. From an optimization perspective, GSA is a memory-less iterative algorithm. The movement direction of each agent is calculated based on the overall force obtained by other agents, rather than the agent itself or global (local) optimal agent. This distinct characteristic makes GSA more powerful than traditional particle swarm and central force optimization algorithms [7]. In GSA, the force is proportional to the fitness value while reversely proportional to the distance between solutions, in such a way heavy masses have large effective attraction radius and hence great intensities of attraction, revealing that the agents tend to move toward the best agent. Previous works [18–23] on solving many practical problems have demonstrated the superiority of the algorithm. However, few literatures paid attention on the improvement for GSA on solving global numerical optimization. Thus, the presented work in this paper is devoted to proposed an effective gravitational search algorithm by incorporating clonal selection mechanism, and gives a rather comprehensive simulation results to verify the performance of the proposed hybrid algorithm.

2 Hybridization of Gravitational Search and Clonal Selection Algorithm

Hybridization of different algorithms is nowadays considered to be an effective method to improve the performance of algorithms. Nevertheless, the key issue of the hybridization progress is how to organize the different search mechanisms effectively from the base algorithms. By analyzing the characteristics of GSA, we can find that GSA has a very fast convergence speed, and the population diversity quickly declines, which indicates that the algorithm can not improve the quality of solutions in the latter search phases, especially for multimodal optimization functions. Regard as the clonal selection algorithm (CSA), the cloning operator carries out a proliferation of B cells in the population, mapping the solutions in the current search space into a temporal higher dimension. Together with the hypermutation operator, the CSA can exploit the search space very well [24, 25]. Based on the above consideration, in the present work, we propose a hybrid algorithm called GSCSA, where the gravitational search mainly performs exploration within the search space, while clonal selection is executed to exploit the neighborhood of the solution found by the gravitational search procedure.

The proposed GSCSA is still an iteration-based optimization algorithm, i.e., the global best solution is acquired iteratively. Initially, a number of randomly generated agents constitute the candidate solutions for the optimization problem. Along with the iteration number goes, these solutions are gradually improved by using gravitational search operators. Once a solution is acquired by gravitational search, it will be undergone the clonal selection procedure. The solution is firstly cloned and proliferated into a set of multiple solutions which are identical with the solution itself. Then the hypermutation operator exploit the surrounding search space and is expected to find better solution to replace the current solution. By doing so, the gravitational search and clonal selection is combined together to optimization the problems. Finally, the best-so-far solution during the population is output when the algorithm termination conditions are fulfilled.

The procedure of GSCSA can be described as follows:

Step 1: a number of N agents with positions of $[x^1(t), x^2(t), ..., x^D(t)]$ are randomly generated within the search space $[low, up]^D$, where D is the dimension of the function, t denotes the iteration number.

Step 2: calculate the fitness for each agent according to the fitness function.

Step 3: the mass of each agent is calculated after computing current population's fitness as follows:

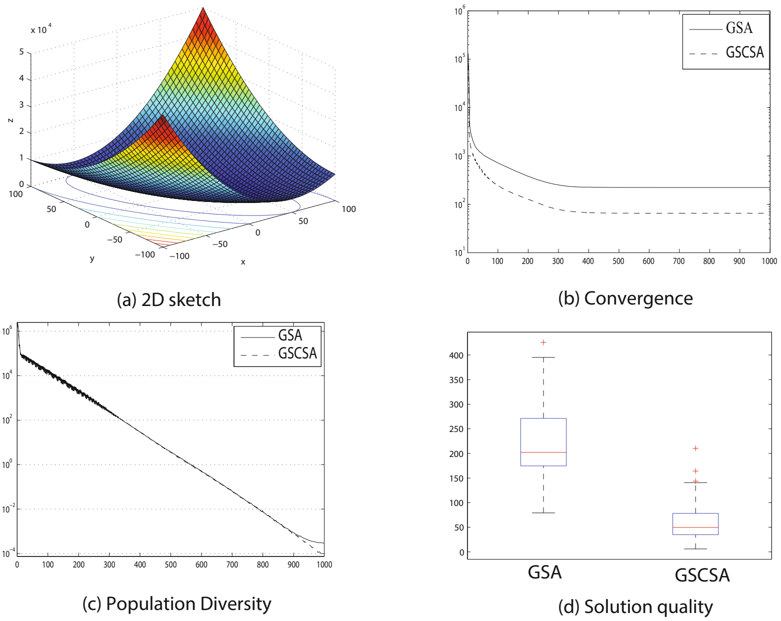
$$q_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)}, \quad M_i(t) = \frac{q_i(t)}{\sum_{j=1}^{s} q_j(t)} \tag{1}$$

where $M_i(t)$ and $fit_i(t)$ represent the mass and the fitness value of the agent i at the iteration t, respectively. Without loss of generality, $best(t)$ and $worst(t)$ are defined for a minimization problem as follows:

$$best(t) = \min_{j\in\{1,...,S\}} fit_j(t), \quad worst(t) = \max_{j\in\{1,...,S\}} fit_j(t) \tag{2}$$

Step 4: the total force that acts on agent i in a dimension d is exerted from other agents set $Kbest$:

$$F_i^d(t) = \sum_{j\in Kbest, j\neq i} r_j G(t) \frac{M_j(t) M_i(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)) \tag{3}$$



(a) 2D sketch

(b) Convergence

(c) Population Diversity

(d) Solution quality

Fig. 1. The 2D sketch (a), convergence graph (b), population diversity (c), and solution quality (d) for function f_2

where $Kbest$ is the set of first K agents with the best fitness values, and the size of K is a function of time, initialized to K_0 at the beginning and decreasing with time. The symbol r_j is a uniformly distributed random number in the interval $[0, 1]$, ε is a small value, $R_{ij}(t)$ is the Euclidean distance between two agents i and j, defined as $||X_i(t), X_j(t)||_2$, The gravitational constant G takes an initial value, G_0, and it will be reduced with time:

$$K = \lfloor(\beta + (1 - \frac{t}{T_{max}})(1 - \beta))K_0\rfloor, \quad G(t) = G_0 \exp(-\alpha\frac{t}{T_{max}}) \tag{4}$$

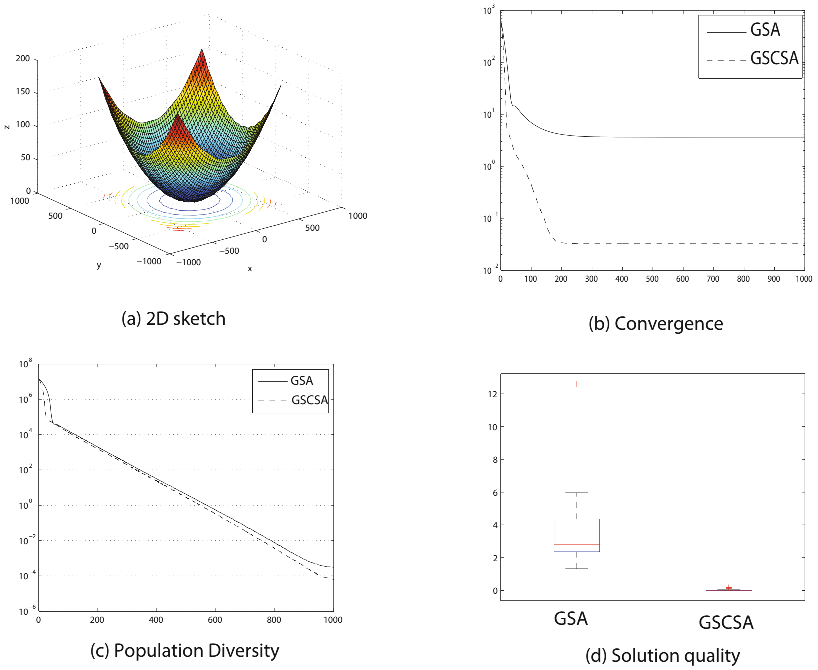
Step 5: attracted by other agents, the agent i will move along with its velocity. The movement direction and position update rule are calculated according to the following equations, respectively.

$$v_i^d(t+1) = r_i v_i^d(t) + \frac{F_i^d(t)}{M_i(t)}, \quad x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{5}$$

where r_i is a uniformly distributed random number in the interval $[0, 1]$, aiming to improve the stochastic ability of the searching algorithm.

Step 6: the cloning operator proliferate NC identical agents to each solution found by previous gravitational search. Then every cloned agent is undergone the hypermutation progress:

$$x_i^d(t+1) = (1 - \beta)x_i^d(t+1) + \beta x_{ran}^d(t+1) \tag{6}$$



(a) 2D sketch

(b) Convergence

(c) Population Diversity

(d) Solution quality

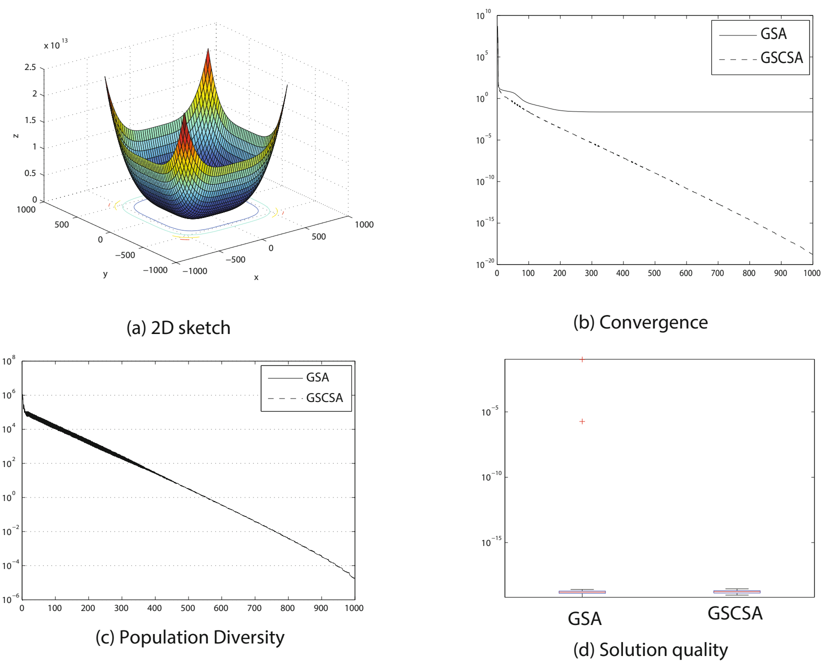
Fig. 2. The 2D sketch (a), convergence graph (b), population diversity (c), and solution quality (d) for function f_5

where the parameter β determines the self-learning ratio, ran is a randomly generated integer from $[1, 2, ..., N]$ and $ran \neq i$. It is worth emphasizing that β is randomly generated in the interval $[0, 1]$. Although this might not be the best choice for β, but it indeed bring no further parameter turning burden for the combined algorithm, and therefore seems to be a compromise between algorithmic design and the algorithm's performance.

Step 7: repeat the steps 2-6 until the maximum iteration number T_{max} is reached.

3 Experimental Results

The performance of the proposed GSCSA is assessed by carrying out optimization on the eight benchmark functions listed in Table 1 [7]. These functions can be divided into three categories: $f_1 - f_3$ are unimodal functions which are relatively easy to be optimized, but the difficulty increases as the dimension size increases; $f_4 - f_7$ are multimodal functions with plenty of local minima which represent the most difficult class of problems for many optimization algorithms; f_8 is a multimodal function with only a few local optima. The different type of benchmark functions test the searching ability of GSCSA from different aspects. When comparing the performance of different optimization algorithms,



(a) 2D sketch

(b) Convergence

(c) Population Diversity

(d) Solution quality

Fig. 3. The 2D sketch (a), convergence graph (b), population diversity (c), and solution quality (d) for function f_6

Unimodal functions trend to reflect the convergence speed of the algorithms in a direct manner, while multimodal functions are likely to estimate the algorithms' capacities of escaping from local optima.

To evaluate the effect of clonal selection on gravitational search, the performance of GSCSA is compared with traditional GSA using the eight functions. The two algorithms are coded using VC++ in Visual Studio 2010 on a personal PC (Intel Core i3 2.40GHz, 2.0GB RAM). The same parameters are used in both algorithms. The population size of agents is set as $N = 50$. The maximum iteration number is set as $T_{max} = 1000$ for all tested functions. The narrowing coefficient of attracting scope is set as $\beta = 0.02$. The initial value gravitational constant is set as $G_0 = 100$, while its decline coefficient of gravitational constant is fixed as $\alpha = 20$. For GSCSA, the cloning size is set to be a relative small value as $NC = 5$ to reduce the computational cost brought by the clonal selection. Moreover, the algorithms are implemented 30 independent runs to make a statistical analysis.

Table 2 records the mean values, the median values and the standard deviation of the simulation results for all tested functions. From this table, we can find that the average quality of the obtained solutions by GSCSA is better than that of GSA for all functions. The GSCSA performs significant better than GSA on the instances f_2, f_5, f_6 and f_7. The median solution of GSCSA is also better than

Table 1. Benchmark problems used in the experiments

Function Definition	Dim.	Domain S
$f_1(X) = \sum_{i=1}^{n}\lvert x_i\rvert + \prod_{i=1}^{n}\lvert x_i\rvert$	30	$[-10, 10]^n$
$f_2(X) = \sum_{i=1}^{n}(\sum_{j=1}^{i} x_j)^2$	30	$[-100, 100]^n$
$f_3(X) = \sum_{i=1}^{n-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	$[-30, 30]^n$
$f_4(X) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10]$	30	$[-5.12, 5.12]^n$
$f_5(X) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}}) + 1$	30	$[-600, 600]^n$
$f_6(X) = \frac{\pi}{n}\{10\sin^2(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})]$ $+(y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4),$ $y_i = 1 + \frac{1}{4}(x_i + 1)$ $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a \leq x_i \geq a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$	30	$[-50, 50]^n$
$f_7(X) = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n-1}(x_i - 1)^2[1 + \sin^2(3\pi x_{i+1})]$ $+(x_n - 1)[1 + \sin^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	$[-50, 50]^n$
$f_8(X) = \sum_{i=1}^{11}[a_i - \frac{x_i(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$	4	$[-5, 5]^n$

Table 2. Comparative Results between GSCSA and the traditional GSA

function	GSA			GSCSA		
	Mean	Median	Std dev	Mean	Media	Std dev
f_1	2.39E-08	2.34E-08	4.53E-09	**2.26E-08**	2.18E-08	4.65E-09
f_2	2.23E+02	2.06E+02	8.25E+01	**6.54E+01**	4.95E+01	4.86E+01
f_3	3.25E+01	2.60E+01	2.99E+01	**2.55E+01**	2.56E+01	1.63E-01
f_4	1.74E+01	1.79E+01	4.12E+00	**1.59E+01**	1.59E+01	3.03E+00
f_5	3.63E+00	3.11E+00	1.91E+00	**3.23E-02**	1.35E-02	5.19E-02
f_6	2.39E-02	1.52E-19	6.73E-02	**1.74E-19**	1.73E-19	5.19E-20
f_7	5.78E-02	3.62E-17	4.15E-01	**4.26E-17**	4.03E-17	9.86E-18
f_8	6.51E-02	6.66E-02	1.68E-02	**4.57E-02**	4.56E-02	4.47E-03

that of GSA, expect for the instance f_6 and f_7. On the contrary, the standard deviation of GSCSA for f_6 and f_7 is much smaller than those of GSA. The reason is that the solution found by GSA in some runs might have trapped into local optima, resulting in some particularly worse solutions. Relative smaller standard deviation of GSCSA indicates that the hybrid algorithm is able to balance the exploration and exploitation very well, thus seems to be more suitable to be a function optimizer.

To further track the effect of clonal selection mechanism on gravitational search, the population diversity and the convergence graph are depicted. The population diversity *DIV* is calculated as follows:

$$DIV = \sum_{i}^{N}\sum_{j}^{D}\lvert x_i^j - x_{\text{gbest}}^j\rvert \tag{7}$$

where $X_{\text{gbest}} = [x^1_{\text{gbest}}, x^2_{\text{gbest}}, ..., x^D_{\text{gbest}}]$ is the global best solution in the population. The larger value of the *DIV* is, the more diversity of the population maintains.

Figs. 1, 2 and 3 illustrate the 2D sketch of the function, the convergence graph, the population diversity, and the distribution of solutions for the function f_2, f_5 and f_6, respectively. For the function f_2, which is a unimodal function, Fig. 1(b) clearly indicates that the convergence speed of GSCSA is much faster than that of GSA. Another evident that demonstrates the effectiveness of clonal selection can be found in Fig. 1(d), in which the solutions found by GSCSA is significant better than those of GSA. With the iteration number increases, the population diversity of both algorithms declines rapidly, suggesting that both algorithms have a very powerful exploitation capacity for solutions. Moreover, in the latter search phase, the population diversity of GSCSA still declines while that of GSA has somewhat slowed down. This phenomenon suggests that GSCSA can always find new similar solutions around the obtained solutions. Compared with GSA, these new similar solutions, which are possibly better solutions, are acquired by the clonal selection procedure. As for the function f_5 and f_6, the same characteristics can be remarked from Fig. 2 and Fig. 3. As a result, we can conclude that the clonal selection procedure enables gravitational search algorithm to find better solutions in terms of average qualities and robustness. In addition, by exploiting the neighborhood of the solutions found by gravitational search, the clonal selection can obtain similar but more promising solutions, realizing a further smaller population diversity but a more accurate solution for the function.

4 Conclusions

In this paper, we proposed a hybrid algorithm by incorporating the clonal selection into the gravitational search algorithm. The hybrid algorithm utilized the global exploration ability of gravitational search and the local exploitation ability of clonal selection, to construct a more powerful function optimizer. Numerical experiments were conducted based on eight benchmark functions including both unimodal and multimodal types. The results indicated that the hybrid algorithm performed better than the original gravitational search algorithm in terms of solution quality and convergence speed for all test functions. In the future, we plan to embed sophisticated mutation operators into the hybrid algorithm to further improve its performance.

Acknowledgments. This work is partially supported by the National Natural Science Foundation of China under Grants 61203325, 61271114 and 61003205, Genguang Project of Shanghai Educational Development Foundation (No. 12CG35), Ph.D. Program Foundation of Ministry of Education of China (No. 20120075120004), and the Fundamental Research Funds for the Central Universities.

References

1. Yao, X., Xu, Y.: Recent advances in evolutionary computation. Journal of Computer Science and Technology 21(1), 1–18 (2006)
2. Deb, K., Saha, A.: Multimodal optimization using a bi-objective evolutionary algorithm. Evolutionary Computation 20(1), 27–62 (2012)
3. Dasgupta, D., Yu, S., Nino, F.: Recent advances in artificial immune systems: Models and applications. Applied Soft Computing 11, 1574–1587 (2011)
4. Chandra Mohan, B., Baskaran, R.: A survey: Ant colony optimization based recent research and implementation on several engineering domain. Expert Systems with Applications 39(4), 4618–4627 (2012)
5. Karaboga, D., Akay, B.: A survey: algorithms simulating bee swarm intelligence. Artificial Intelligence Review 31(1), 61–85 (2009)
6. Del Valle, Y., Venayagamoorthy, G.K., Mohagheghi, S., Hernandez, J.C., Harley, R.G.: Particle swarm optimization: basic concepts, variants and applications in power systems. IEEE Transactions on Evolutionary Computation 12(2), 171–195 (2008)
7. Rashedi, E., Nezamabadi-pour, H., Saryazdi, S.: Gsa: A gravitational search algorithm. Information Sciences 179(13), 2232–2248 (2009)
8. Rashedi, E., Nezamabadi-pour, H., Saryazdi, S.: Bgsa: binary gravitational search algorithm. Natural Computing 9(3), 727–745 (2010)
9. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation 1, 67–82 (1997)
10. Chen, C.Y., Chang, K.C., Ho, S.H.: Improved framework for particle swarm optimization: Swarm intelligence with diversity-guided random walking. Expert Systems with Applications 10(38), 12214–12220 (2011)
11. Pedersen, M.E.H., Chipperfield, A.J.: Simplifying particle swarm optimization. Applied Soft Computing 10(2), 618–628 (2010)
12. Leong, W.F., Yen, G.G.: Pso-based multiobjective optimization with dynamic population size and adaptive local archives. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 38(5), 1270–1293 (2008)
13. Sun, J., Wu, X., Palade, V., Fang, W., Lai, C.H., Xu, W.: Convergence analysis and improvements of quantum-behaved particle swarm optimization. Information Sciences 193, 81–103 (2012)
14. Alatas, B., Akin, E., Ozer, A.B.: Chaos embedded particle swarm optimization algorithms. Chaos, Solitons & Fractals 40(4), 1715–1734 (2009)
15. Van den Bergh, F., Engelbrecht, A.P.: A cooperative approach to particle swarm optimization. IEEE Transactions on Evolutionary Computation 8(3), 225–239 (2004)
16. El-Abd, M., Kamel, M.S.: A hierarchal cooperative particle swarm optimizer. In: Proc. IEEE Swarm Intell. Symp., pp. 43–47 (2006)
17. Mirjalili, S., Hashim, S.Z.M., Sardroudi, H.M.: Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. Applied Mathematics and Computation 218(22), 11125–11137 (2012)
18. Lopez-Molina, C., Bustince, H., Fernandez, J., Couto, P., Baets, B.D.: A gravitational approach to edge detection based on triangular norms. Pattern Recognition 43, 3730–3741 (2010)
19. Li, C.S., Zhou, J.Z.: Parameters identification of hydraulic turbine governing system using improved gravitational search algorithm. Energy Conversion and Management 1(52), 374–381 (2011)

20. González-Álvarez, D.L., Vega-Rodríguez, M.A., Gómez-Pulido, J.A., Sánchez-Pérez, J.M.: Applying a multiobjective gravitational search algorithm (mo-gsa) to discover motifs. In: Cabestany, J., Rojas, I., Joya, G. (eds.) IWANN 2011, Part II. LNCS, vol. 6692, pp. 372–379. Springer, Heidelberg (2011)
21. Li, C.: Ts fuzzy model identification with gravitational search based hyper-plane clustering algorithm. IEEE Transactions on Fuzzy Systems 99, 1–12 (2011)
22. Han, X., Chang, X.: A chaotic digital secure communication based on a modified gravitational search algorithm filter. Information Science 208, 14–27 (2012)
23. Precup, R.E., David, R.C., Petriu, E.M., Preitl, S., Radac, M.B.: Novel adaptive gravitational search algorithm for fuzzy controlled servo systems. IEEE Transactions on Industrial Informatics 8(4), 791–800 (2012)
24. de Castro, L., Zuben, F.J.V.: Learning and optimization using clonal selection principle. IEEE Trans. on Evolutionary Computation 6(3), 239–251 (2002)
25. Gao, S., Wang, R.L., Tamura, H., Tang, Z.: A Multi-Layered Immune System for Graph Planarization Problem. IEICE Trans. on Information and Systems E92-D(12), 2498–2507 (2009)

A Hybrid Genetic Programming with Particle Swarm Optimization

Feng Qi[1], Yinghong Ma[1], Xiyu Liu[1], and Guangyong Ji[2]

[1] Shandong Normal University, Jinan 250014, China
qfsdnu@126.com
[2] Yandtai Nanshan University, Yantai 265706, China

Abstract. By changing the linear encoding and redefining the evolving rules, particle swarm algorithm is introduced into genetic programming and an hybrid genetic programming with particle swarm optimization (HGPPSO) is proposed. The performance of the proposed algorithm is tested on tow symbolic regression problem in genetic programming and the simulation results show that HGPPSO is better than genetic programming in both convergence times and average convergence generations and is a promising hybrid genetic programming algorithm.

Keywords: Genetic Programming, Particle Swarm Optimization, Evolving Rules, Symbolic Regression Problem.

1 Introduction

Genetic programming (GP) is a domain-independent problem-solving approach in which computer programs are evolved to solve, or approximately solve, problems. GP is based on the Darwinian principle of reproduction and survival of the fittest and analogs of naturally occurring genetic operations such as crossover and mutation. Genetic programming was first proposed and normalized by Koza [1] in 1992, since then it became another new topic field after genetic algorithms in evolutionary computation. In the past 20 years, it has been greatly developed and obtained numbers of meaningful research results, such as [2][3][4][5][6][7][8]. At the same time, GP is applied in solving various problems, such as classification[9][10][11], neural network designing[12], time series forecasting[13][14][15], etc. and shows its higher nonlinear ability and performance.

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995[16], inspired by social behavior of bird flocking. PSO shares many similarities with evolutionary computation techniques. The system is initialized with a population of random solutions and searches for optima by updating generations and especially it has no evolution operators such as crossover and mutation. In past several years, PSO has attracted many scholars to enter this area and developed significant research work such as [17][18][19][20]. More and more research results

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 11–18, 2013.

demonstrate that PSO can get better solutions in a faster and cheaper way compared with other methods. Totally speaking, few parameters to adjust and One version, with slight variations, works well in a wide variety of applications are the typical advantages which make the PSO a more potential and promising algorithm.

In order to find a new way for improving the performance of genetic programming, this paper attempts to introduce and modify PSO algorithm into genetic programming. Based on this idea, we propose a hybrid genetic programming with particle swarm optimization (HGPPSO), applied for controlling and evolving populations in each generations. The paper is organized as follows, a brief introduction on genetic programming and particle swarm optimization is showed in Section 2. Section 3 gives the details of the proposed hybrid algorithm. Simulation results are presented in Section 4. Finally, some conclusions are summarized in Section 5.

2 A Brief Summary for Basic GP and PSO

In this paper, the genetic programming and particle swarm optimization which we refer are both their basic forms.

In genetic programming, the genetic algorithm operates on a population of computer programs by varying sizes and shapes[1]. Genetic programming starts with thousands or millions of randomly generated computer programs composed of the available programmatic ingredients and then applies the principles in evolution of species to breed an improved population of programs. During evolution of the population, the Darwinian principle of survival of the fittest is applied also with an analog of the naturally-occurring genetic operation of crossover (sexual recombination), and occasional mutation. The crossover operation is the main operator and designed to create syntactically valid offspring programs (given closure amongst the set of programmatic ingredients). Genetic programming combines the expressive high-level symbolic representations of computer programs with the near-optimal efficiency of learning from genetic algorithm. A computer program that solves (or approximately solves) a given problem often emerges from this process.

The procedure of genetic programming can be described by the following three steps:

(1). Generate an initial population of random compositions of the functions and terminals of the problem.
(2). Iteratively run the following substeps until the termination criterion has been satisfied:
    (a). Evaluate each individual in the population by fitness measure function.
    (b). Create a new population by applying the following genetic operations. The operations are applied to individuals chosen from the population with a probability based on their fitness.
        (i). Reproduction: Reproduce an existing individual by copying it into the new population.

(ii). Crossover: Create two new offsprings from two existing parents by recombining randomly chosen parts of two existing parents using crossover operation applied at a randomly chosen crossover point within each individual.

(iii). Mutation: Create one new offspring from one existing individual by mutating a randomly chosen part of the individual.

(3). The individual identified by the method of result designation is considered as the result for the run (e.g., the best-so-far individual). This result may be a solution (or an approximate solution) to the problem.

In particle swarm optimization, the potential solutions, called particles, fly through the problem space by following the current optimum particles. Each particle keeps track of its coordinates in problem space which are associated with the best solution (fitness) it has achieved so far and is called ”pbest”. Another best value is tracked by the particle swarm optimizer, obtained so far by all particles in the neighbors of the particle and is named ”lbest”. When a particle takes all the population as its topological neighbors, the best value is a global best and is called ”gbest”. The particle swarm optimization concept consists of, at each time step, changing the velocity of each particle toward its ”pbest” and ”lbest” locations. Acceleration is weighted by a random term, with separate random numbers generated for ”pbest” and ”lbest” locations.

The procedure of particle swarm optimization can be described by the following three steps:

(1). Generate an initial population of random compositions of the problem.

(2). Iteratively run the following substeps until the termination criterion has been satisfied:

(a). Evaluate each particle in the population by fitness measure function and if the fitness value is better than the best fitness value (*pbest*) in history, then set current value as the new *pbest*.

(b). Choose the particle with the best fitness value of all the particles as the *gbest*

(c). Create a new population by applying the following velocity and position formula on each particle:

$$v(t+1) = v(t)+c_1*R1*(pbest(t)-x(t))+c_2*R2*(lbest(t)-x(t)) \ . \quad (1)$$

$$x(t+1) = x(t) + v(t+1) \ . \quad (2)$$

(3). The individual identified by *pbest* is considered as the result for the run (e.g., the best-so-far individual). This result may be a solution (or an approximate solution) to the problem.

From the above description GP and PSO, the main ideas of two algorithms can be summarized as follows:

(a). Decide the encoding way of individuals via problem;

(b). Create individuals in initial population;

(c). Evaluate the population;
(d). Obtain evolution message by a certain strategy and generate the new population;
(e). Goto step (c) until some stopping conditions satisfied;

Clearly, step (a) and (d) is the main difference between GP and PSO. Step (a) denotes the tree and linear encoding method and step (d) denotes the different evolution strategy. So when PSO algorithm is introduced into GP, the encoding and evolving way must be modified in order to fit GP algorithm, details of this work is described in next section.

3 Hybrid Genetic Programming with Particle Swarm Optimization

The most common form of genetic programming (and the one considered in this paper) evolves programs as LISP-like program-trees of function nodes [1]. These trees serve both as the genetic material and as the resultant program individual. For particle swarm optimization, all the particles are the form of linear vectors and their evolving strategy is based on this encoding.

In this paper, in order to introduce PSO into GP, two modifications are applied in PSO, firstly, change the linear encoding to tree encoding and secondly, redefine the evolving rules based on the new encoding. Actually, each particle is evolved by combining velocity updating rule (1) and position updating rule (2), which show that for each particle x its new evolved form x_{new} is decided by its old form x_{old}, distance measure between x_{old} and *pbest*, distance measure between x_{old} and *gbest*. This relation ship can be described as the following function form:

$$x_{new} = f(x_{old}, x_{old} - pbest, x_{old} - gbest) \ . \tag{3}$$

Where, f is a decision function which combines the information of x_{old}, $x_{old} - pbest$ and $x_{old} - gbest$, then gives the new evolved particle. $x_{old} - pbest$ and $x_{old} - gbest$) are called self study and population study, respectively.

Clearly, if the linear encoding becomes tree encoding in PSO, updating rule (3) need to be redefined and then the tree encoding based PSO can be used in GP. In this paper, the following definitions with considering genetic operators in GP are proposed:

- $x_{old} - pbest$ is redefined as Select(Cross(x_{old},*pbest*));
- $x_{old} - gbest$ is redefined as Select(Cross(x_{old},*gbest*));
- f is redefined as RWS, which means roulette wheel selection based on fitness of particle;

where, x_{old}, *pbest*, *gbest* are tree encoded particle; *Cross* denotes the cross operation in GP which creates two offsprings; *Select* denotes selection method for offsprings created by cross operation. By comparing with the fitness of the two offsprings, the higher one is selected as final result. Then, the updating rule for tree encoded PSO for GP is as follows:

$$x_{new} = RWS(x_{old}, Select(Cross(x_{old}, pbest)), Select(Cross(x_{old}, gbest))). \quad (4)$$

The procedure of the proposed hybrid genetic programming with particle swarm optimization(HGPPSO) can be described as follows:

(1). Generate an initial population of random compositions of the functions and terminals of the problem.

(2). Iteratively run the following substeps until the termination criterion has been satisfied:

  (a). Evaluate each tree encoded particle in the population by fitness measure function and if the fitness value is better than the best fitness value (*pbest*) in history, then set current value as the new *pbest*.

  (b). Choose the tree encoded particle with the best fitness value of all the particles as the *gbest*

  (c). Create a new population by applying the following evolving rule (4) on each particle:

(3). The individual identified by *pbest* is considered as the result for the run (e.g., the best-so-far individual). This result may be a solution (or an approximate solution) to the problem.

4 Simulations

In order to test the performance of the proposed HGPPSO algorithm with classic GP(without mutation), the classic problem in genetic programming called symbolic regression problem[1] is applied with two complex functions $x^4+x^3+x^2+x$ and $\sin(x)+\sin(2x)+\sin(3x)$, where $x \in [-1,1]$ and $x \in [0,2\pi]$, respectively.

For both the two functions, settings of the simulations are listed as follows:

(1) Training set: it is formed by 20 (x_i,y_i) data points, in which values of x_i are chosen at random in the interval [-1,1](or [0,2π]);

(2) Function set: {+, -, *, /, sin, cos, exp};

(3) Terminal set: {x};

(4) Raw fitness: it is calculated by summing the absolute value of difference between the value produced by the function and the target value y_i;

(5) Population size: 500;

(6) Maximum evolution generation: 51;

(7) Running times for each simulation: 20;

The simulation results of the two functions on HGPPSO and GP are list in Table 1.

In column *Algorithms* of Table 1, GP-fun1 and HGPPSO-fun1 denote that applying GP and HGPPSO on function $x^4+x^3+x^2+x$, respectively. GP-fun2 and HGPPSO-fun2 mean that using GP and HGPPSO on function $\sin(x)+\sin(2x)+\sin(3x)$, respectively. Column *CT* denotes that running 20 times for each algorithm listed in column *Algorithms* with count of the convergence times

Table 1. Table of comparison results about HGPPSO and GP

Algorithms	CT	ACG	ACT
GP-fun1	12	26	6.865s
HGPPSO-fun1	19	15	4.432s
GP-fun2	9	32	8.512s
HGPPSO-fun2	16	18	5.673s

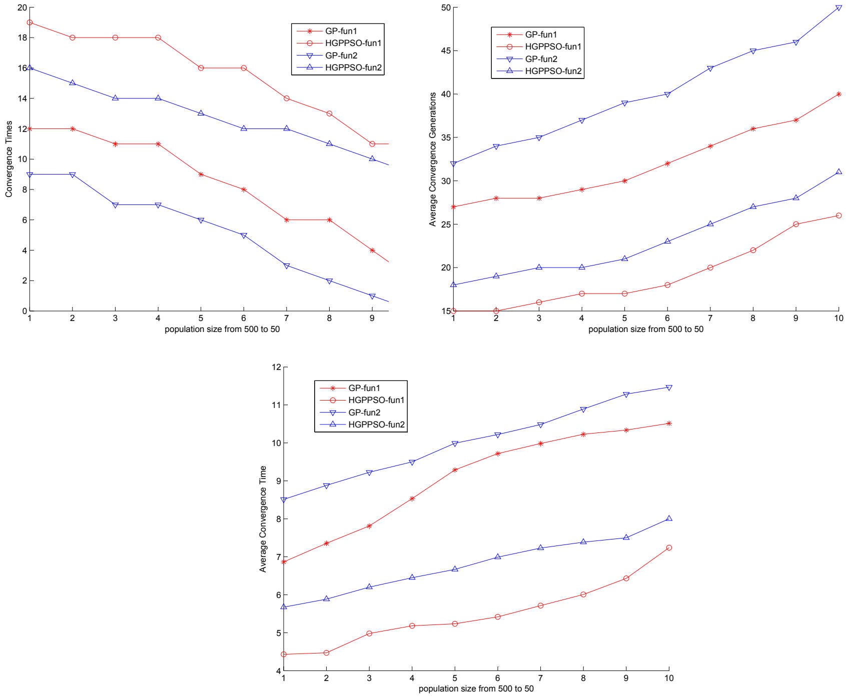


Fig. 1. Changing trend of CT, ACG and ACT for the two functions with algorithms

when the algorithm has found the global optimum. Column *ACG* means that the average of the total convergence generations for each 20 times. The total average time for the 20 times' running are listed in column *ACT* From the comparison results, firstly, in aspect of convergence times , the convergence times about the two functions of HGPPSO are all more than that of GP and have a clear trend of growth in convergence times; Secondly, in aspect of average convergence generations, the convergence generations of HGPPSO are all coming earlier than that of GP and have an obvious improvement in the average convergence generations. Finally, in aspect of average convergence time, for the two functions, the average time of HGPPSO used are all less than classic GP. From the above analysis, the performance of the proposed HGPPSO is better than classic GP in the above three aspects and is a promising improvement in GP.

In order to testify the performance of the proposed HGPPSO algorithm further more, population size of the above simulations is minus 50(beginning with 500) for each case and then redo the above simulations. By this way, we investigate the fluctuation trend of CT, ACG and ACT. The simulation results are showed in Fig 1.

From the above Fig 1, we can see that, with the reduce of the population size, the fluctuation trend of HGPPSO in CT, ACG, ACT changes smoothly and stably. That is to say, HGPPSO can get better performance in a small population size, this is the most significant feature of HGPPSO which is useful in solving problems.

5 Conclusions

By changing the linear encoding to tree encoding and redefining the evolving rules based on the new encoding, PSO algorithm is introduced into genetic programming and an hybrid genetic programming with particle swarm optimization (HGPPSO) is proposed. The procedure of HGPPSO and evolving rules are illustrated clearly in this paper. The performance of the proposed algorithm is tested on two complex functions in the symbolic regression problem and simulation results show that HGPPSO is better than GP in both convergence times and average convergence generations and is a promising hybrid algorithms. In future works, more benchmark problems in genetic programming will be resolved by the proposed algorithm in order to testify the performance of HGPPSO.

Acknowledgement. This Research is supported by Natural Science Foundation of China(No. 60873058), Natural Science Foundation of Shandong Province(No. ZR2011FM001), Shandong Soft Science Major Project(No. 2010RKMA2005), Natural Science Foundation of China(No. 71071090) and Shandong Young Scientists Award Fund(NO. BS2009DX005).

References

1. Koza, J.R.: Genetic programming: on the programming of computers by means of natural selection. MIT Press, Cambridge (1992)
2. Luke, S.: Two fast tree-creation algorithms for genetic programming. IEEE Transactions on Evolutionary Computation 4(3), 274–283 (2000)
3. Kushchu, I.: Genetic programming and evolutionary genralization. IEEE Transactions on Evolutionary Computation 6(5), 431–442 (2002)
4. Gustafson, S., Kendall, G.: Diversity in genetic programming an analysis of measures and correlation with fitness. IEEE Transactions on Evolutionary Computation 8(1), 47–62 (2004)
5. Yun-Seog, Y., Won-Sun, R., Young-Soon, Y., Nam-Joon, K.: Implementing linear models in genetic programming. IEEE Transactions on Evolutionary Computation 8(6), 542–566 (2004)

6. Nguyen, X.H., McKay, R.I., Essam, E.: Representation and structural difficulty in genetic programming. IEEE Transactions on Evolutionary Computation 10(2), 157–166 (2006)
7. Uy, N.Q., Hien, N.T., Hoai, N.X., O'Neill, M.: Improving the generalisation ability of genetic programming with semantic similarity based crossover. In: Esparcia-Alcázar, A.I., Ekárt, A., Silva, S., Dignum, S., Uyar, A.Ş. (eds.) EuroGP 2010. LNCS, vol. 6021, pp. 184–195. Springer, Heidelberg (2010)
8. Castelli, M., Manzoni, L., Silva, S., Vanneschi, L.: A quantitative Study of Learning and Generalization in Genetic Programming. In: Silva, S., Foster, J.A., Nicolau, M., Machado, P., Giacobini, M. (eds.) EuroGP 2011. LNCS, vol. 6621, pp. 25–36. Springer, Heidelberg (2011)
9. Muni, D.P., Pal, N.R., Das, J.: A novel approach to design classifiers using genetic programming. IEEE Transactions on Evolutionary Computation 8(2), 183–196 (2004)
10. Pedro, G.E., Sebastian, V., Francisco, H.: A survey on the application of genetic programming to classification. IEEE Transactions on Systems, Man, and Cybernetics 40(2), 121–144 (2010)
11. Hajira, J., Abdul, R.B.: Review of classification using genetic programming. International Journal of Engineering Science and Technology 2(2), 94–103 (2010)
12. Qi, F., Liu, X.Y., Ma, Y.H.: Synthesis of neural tree models by improved breeder genetic programming. Neural Computing and Application 3, 515–521 (2012)
13. Neal, W., Zbigniew, M., Moutaz, J.K., Rob, R.M.: Time series forecasting for dynamic environments the DyFor genetic program model. IEEE Transactions on Evolutionary Computation 11(4), 433–452 (2007)
14. Emiliano, C.J.: Long memory time series forecasting by using genetic programming. Genet. Program Evolvable Mach. 12, 429–456 (2011)
15. Bartoli, A., Davanzo, G., De Lorenzo, A., Medvet, E.: GP-Based electricity price forecasting. In: Silva, S., Foster, J.A., Nicolau, M., Machado, P., Giacobini, M. (eds.) EuroGP 2011. LNCS, vol. 6621, pp. 37–48. Springer, Heidelberg (2011)
16. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, pp. 1942–1948. Perth (1995)
17. Clerc, M., Kennedy, J.: The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Transactions on Evolutionary Computation 6(1), 58–73 (2002)
18. Mendes, R., Kennedy, J., Neves, J.: The fully informed particle swarm: Simpler, maybe better. IEEE Transactions on Evolutionary Computation 8, 204–210 (2004)
19. Janson, S., Middendorf, M.: A hierarchical particle swarm optimizer and its adaptive variant. IEEE Transactions on Systems, Man and Cybernetics - Part B 3(6), 1272–1282 (2005)
20. Liang, J.J., Qin, A.K., Suganthan, P.N., et al.: Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. IEEE Transactions on Evolutionary Computation 10(3), 281–295 (2006)

A *Physarum* Network Evolution Model Based on IBTM

Yuxin Liu[1], Zili Zhang[1,2,⋆], Chao Gao[1], Yuheng Wu[1], and Tao Qian[1]

[1] School of Computer and Information Science
Southwest University, Chongqing 400715, China
[2] School of Information Technology, Deakin University, VIC 3217, Australia
zhangzl@swu.edu.cn

Abstract. The traditional Cellular Automation-based *Physarum* model reveals the process of amoebic self-organized movement and self-adaptive network formation based on bubble transportation. However, a bubble in the traditional *Physarum* model often transports within active zones and has little change to explore new areas. And the efficiency of evolution is very low because there is only one bubble in the system. This paper proposes an improved model, named as Improved Bubble Transportation Model (IBTM). Our model adds a time label for each grid of environment in order to drive bubbles to explore new areas, and deploys multiple bubbles in order to improve the evolving efficiency of *Physarum* network. We first evaluate the morphological characteristics of IBTM with the real *Physarum*, and then compare the evolving time between the traditional model and IBTM. The results show that IBTM can obtain higher efficiency and stability in the process of forming an adaptive network.

Keywords: *Physarum Polycephalum*, *Physarum* Model, IBTM, Network Evolution.

1 Introduction

Physarum Polycephalum is a unicellular and multi-headed slime mold which lives in a dark and moist environment. Biological experiments show that the plasmodium — a 'vegetative' phase of *Physarum* — has the ability to form self-adaptive and high efficient networks without central control mechanism. Statistical analyses show that such networks have the shortest path with higher fault tolerance [1,2]. These *Physarum* networks are comparable to real-world infrastructure networks, such as UK motorways [3], the Tokyo rail system [4], the longest road in USA and the longest national motorway in Europe [5]. Taking advantage of the simple structure and self-organized abilities, *Physarum* has been a research hotspot of distributed computing over the past decade [6,7].

Nakagaki et al. [8] are the first to illustrate the primitive intelligence of the plasmodium of *Physarum* — which can solve the shortest-path between two points in a labyrinth. Tero et al. [9] have constructed a mathematical model

[⋆] Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 19–26, 2013.

based on feedback regulations between the internal protoplasmic flow and the thickness of each tube. Adamatzky [10,11] has utilized an Oregonator model to simulate the growth and foraging behavior of plasmodium of *Physarum*. Jones [6,12] has illustrated the evolving process of *Physarum*-like transport networks through a bottom-up multi-agent based system.

However, the aforementioned models are unable to further explain the relationship between the process of network formation and the movement of amoeba, which plays a central role in *Physarum* computing [13]. Hence, Gunji et al. [13] have adopted a CA-based *Physarum* model, named as CELL, to explore and unveil the relation underlying network formation. The evolution of CELL depends on loop cutting, tentacle withdrawal, and the choice of shorter paths by the bubble transportation. Experimental results show that CELL can solve a maze problem, a Steiner minimum tree problem as well as a spanning tree problem [13,14,15]. However, a bubble often transports within active zones and cannot explore new areas during the evolution of CELL, which will affect the evolving efficiency. Moreover, the mechanism of single bubble's transportation also restricts the calculating speed.

Therefore, we make one step forward towards developing a high computational efficient model, called Improved Bubble Transportation Model (IBTM). The contributions are two-fold: 1) We first add a time label for each grid in an environment, which is used to drive bubbles to transport to inactive zones. 2) We further propose a multi-bubble parallel computing scheme to accelerate the convergence speed of whole system.

The remainder of this paper is organized as follows. Section 2 introduces the traditional *Physarum* model CELL and describes two strategies of IBTM. Section 3 validates IBTM using a real organism, and shows the efficiency of IBTM through comparing the running time with CELL. We conclude our researches in Section 4.

2 An Improved *Physarum* Model Based on IBTM

2.1 A Traditional *Physarum* Model CELL

The traditional *Physarum* model CELL uses an agent system to simulate the biological evolution mechanism of the plasmodium of *Physarum* (Fig. 1). The system consists of three parts: environment, agent and agent's behaviors. An

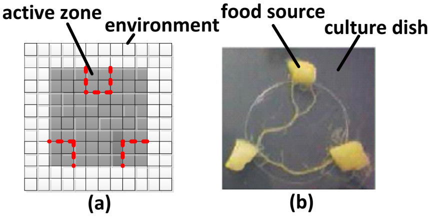


Fig. 1. The mapping between (a) an agent-based system and (b) a biological experiment of the plasmodium of *Physarum*

environment represents a culture dish with M*N planar grids, as shown in Fig. 1(a). And each grid has four neighbors. During the evolving process of *Physarum*, grids are divided into two groups: internal grids and external grids. Especially, some internal grids, which are adjacent to the external grids, are defined as "boundary grids". The regions corresponding to the positions of food sources are defined as active zones, which are some grids surrounded by dotted lines in Fig. 1(a). An agent represents a bubble in CELL, and agent's behaviors include generation, movement and displacement.

Figure 2 illustrates a life-cycle of an agent in an environment. First, an agent is randomly generated, based on "generation behavior", from external grids that are adjacent to boundary grids of active zones (Fig. 2(b)). Second, the agent randomly chooses one internal grid from its four neighbors to reside based on "movement behavior", which has not been resided by itself (Fig. 2(c) and Fig. 2(d)). Those internal grids that have been resided by the agent compose a trace of the agent (the black arrows in Fig. 2(c) and Fig. 2(d)). Finally, if there is no internal grid for the agent to reside, the agent exchanges the current internal grid with the initial external grid based on "displacement behavior" (as shown in Fig. 2(e)). After that, a new agent is randomly generated again.

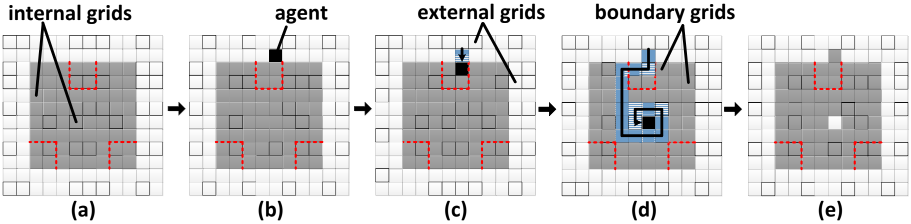


Fig. 2. A life-cycle of an agent in an environment. The grey, white and black grids represent internal grids, external grids and agents respectively.

CELL continually moves internal grids to active zones based on the bubble transportation, until one route connecting active zones appears. This final route approximates an efficient network produced by the plasmodium of *Physarum*. However, with the increment of active zones, an agent often transports within active zones, which restricts the evolving efficiency of CELL. Meanwhile, the mechanism of single agent's transportation also restricts the efficiency of evolution. With those observations in mind, this paper proposes an improved model IBTM in order to improve the efficiency of CELL.

2.2 A *Physarum* Model IBTM

IBTM improves the evolving efficiency of the traditional *Physarum* model based on two optimization strategies. The first strategy (S1) adds a time label for each grid in a CELL environment. The second strategy (S2) applies the parallel computing of multiple agents.

2.2.1 Marking Each Grid with a Time Label

In order to drive an agent to transport out of the active zones, an environment is modified by adding a time label p for each grid. Meanwhile, an agent adjusts its basic behaviors — modifying the "movement behavior" and adding an "updating behavior" after the "displacement behavior".

cell(i, j).p represents the value of time label at the position *(i, j)* in an environment. Initially, *cell(i, j).p* is zero. The "movement behavior" of an agent is modified as follows: an agent randomly selects one internal grid from its four neighbors as a target to transport. The target is not on the agent's trace, and its time label is not smaller than other alternative grids. "Updating behavior" includes *cell(i, j).p*++ for internal grids, and *cell(i, j).p*=0 for external grids. Through "updating behavior", an agent sets *cell(i, j).p*=1 for its initial external grid (that is internal grid now) and sets *cell(i, j).p*=0 for its final internal grid (that is external grid now). So the later a grid emerges in active zones, the smaller its value of time label p is. This strategy can drive an agent to transport towards the inside of CELL, and improve the effectiveness of each agent as well as reduce the total number of agents required for a final network.

2.2.2 Multi-agent Parallel Computing

In order to deploy multiple agents in *Physarum* network evolution process, agent's behaviors are redefined. First, the basic "movement behavior" is modified. Second, two new behaviors after the "movement behavior", named as "disappearance behavior" and "waiting behavior", are added.

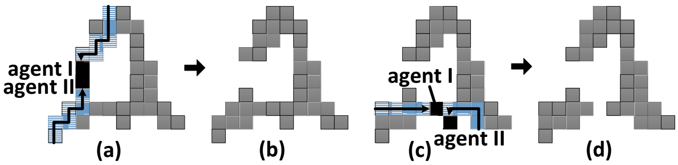


Fig. 3. The mutually-exclusive relationship breaks off all routes connecting the left-bottom active zone when two agents transport in parallel

The general idea of this strategy is as follows. First, multiple agents are randomly generated with different positions based on "generation behavior" respectively. Second, each agent randomly chooses and transports to one adjacent internal grid that has not been resided by any agent based on the modified "movement behavior". Note that there is a mutually-exclusive relationship between agents. Through this mutually-exclusive relationship, agents can easily explore new areas in an environment. However, the mutually-exclusive relationship may break off all routes that connect active zones. Taking Fig. 3 as an example, after two processes of "displacement behavior" of agents (i.e., Fig. 3(a) to Fig. 3(b) and Fig. 3(c) to Fig. 3(d) respectively), two routes that connect the left-bottom active zone are both interrupted. Therefore, it cannot obtain a solution for solving graph problem. To reduce the probability of such situations, we introduce two constraints C1 and C2. When an agent has no target to transport and satisfies the constraints of C1 or C2, the agent disappears (i.e. "disappearance behavior"). Otherwise, the agent stays at the same place (i.e. "waiting behavior"). Finally, if all agents have no target to transport, each agent performs "displacement behavior" respectively.

C1: An agent resides in the neighbor position of the other agent, and satisfies one of the following conditions: 1) at least one agent's north-south or east-west neighbors are external grids; 2) the moving directions of these two agents are adverse and the other two neighbors of both are external grids (Fig. 4(a)).

C2: Among the four neighbors of an agent, at least two neighbors are on agents' traces. Meanwhile, at least one neighbor is on other agents' trace. Besides, the rest neighbors are external grids (Fig. 4(b)).

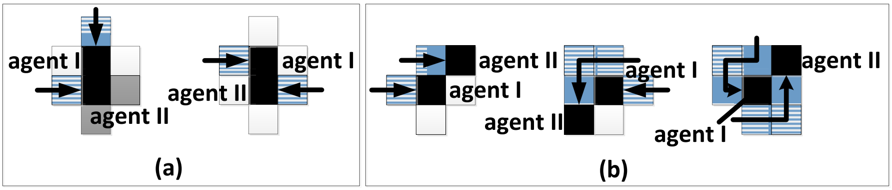


Fig. 4. The illustration of (a) constraint C1 and (b) constraint C2

In section 3, we take experiments to show the validity of the aforementioned two constrains. Since our experiments are limited, it is impossible to list and avoid all kinds of situations in which all routes that connect two active zones are interrupted. These two constrains can solve the most common problems and avoid them to a certain extend.

2.2.3 The Algorithm Description of IBTM

Algorithm 1 presents the procedure of IBTM. A life-cycle of multiple agents begins from "generation behavior" of multiple agents and ends with "updating behavior".

3 Simulation Experiments

3.1 The Experimental Setup

In this section, we evaluate the evolving process of IBTM based on patterns generated by the real *Physarum* obtained in [14], and compare the evolving efficiency between CELL and IBTM. All experiments are undertaken in one computer with CPU: Inter(R) Core(TM)2 Duo E4500 2.20GHz, RAM: 2.00GB, OS: Windows 7. The application development environment is Microsoft Visual Studio 2010. In accordance with the arrangement of food sources in [14], we set the internal grids as an aggregation consisting of 25*25 grids. The size of each grid is 10*10 square-millimeters. Then, we arrange three active zones with regions of 5*5 grids at two adjacent angles and one opposite side in the aggregation, forming an equilateral triangle (Fig. 1(a)).

Algorithm 1. A life-cycle of multiple agents

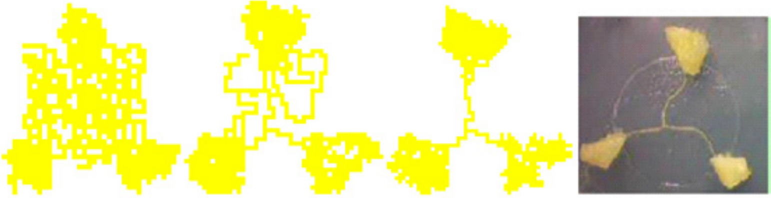
```
Initialization(); //randomly generate $n$ agents
Dim [ ] *StopMoving* as bool
While there have any agents $i$ whose *StopMoving[i]* == false
    For each agent $i$ do
        If *StopMoving[i]*==false then
            If agent $i$ has target to transport then
                Move(); // "movement behavior" in multi-agent system
            Else
                *StopMoving[i]*=true;
                If agent $i$ satisfies C1 or C2 then
                    Disappear(); //"disappearance behavior"
                Else
                    Wait(); //"waiting behavior"
EndWhile
For each agent do
    Replace(); //"displacement behavior" in multi-agent system
UpdateLabel(); //"updating behavior"
```

3.2 Verification with Biological Experiments

Figure 5 shows the evolving process of an adaptive network based on IBTM. A time label is used to drive agents to transport towards the inside of internal grids. And we deploy three agents in parallel. As time goes on, a stable route comes out. The final pattern of Fig. 5(c) is very similar to the biological pattern observed in a real organism as shown in Fig. 5(d).

3.3 Computational Efficiency

Table 1 compares the evolving time between CELL and IBTM. Fig. 6 shows the mean and standard value of evolving time based on the data in Table 1.



(a) time=5.38m (b) time=26.02m (c) time=52.32m (d)

Fig. 5. The evolution of a network based on IBTM and the comparison with a real organism. (a) to (c) is the evolving process of IBTM to form a network connecting three active zones, and (d) is a network generated by the plasmodium of *Physarum*. The time unit is minute.

Table 1. The running time between CELL and IBTM (unit: minute)

Model/runningtime/time	1	2	3	4	5
CELL	202	190	155	159	164
IBTM	33.37	52.32	31.78	38.82	35.95

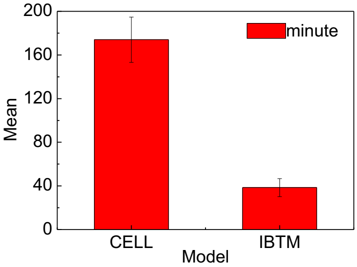


Fig. 6. The mean and standard value of evolving time between CELL and IBTM after running five times

It indicates that the average evolving time and standard value of CELL is reduced 77.90% and 60.52% by IBTM respectively. Hence, this confirms that the efficiency and stability of CELL have a significant improvement by the two optimization strategies.

4 Conclusion

With regard to the low efficiency and slow execution of the traditional *Physarum* model, we propose an improved model IBTM in the paper. Our proposed model improves the evolving efficiency utilizing two optimization strategies: 1) Adding a label to mark the existence time of internal grids, which can drive bubbles to transport to new areas; 2) Formulating constraints to realize multi-bubble parallel computing. In particular, experimental results confirm that IBTM has the ability to efficiently and stably simulate the evolving process of a *Physarum* network. We further compare the average evolving time and standard value between CELL and IBTM. The simulation results show that our proposed model is more efficient and stable than the traditional model.

Acknowledgement. This project was supported by the Natural Science Foundation of Chongqing (No. cstc2012jjA40013) and Specialized Research Fund for the Doctoral Program of Higher Education (No. 20120182120016).

References

1. Nakagaki, T., Yamada, H., Hara, M.: Smart Network Solutions in an Amoeboid Organism. Biophysical Chemistry 107(1), 1–6 (2004)
2. Nakagaki, T., Kobayashi, R., Nishiura, Y., Ueda, T.: Obtaining Multiple Separate Food Sources: Behavioural Intelligence in the Physarum Plasmodium. Proceedings of the Royal Society of London. Series B: Biological Sciences 271(1554), 2305–2310 (2004)
3. Adamatzky, A., Jones, J.: Road Planning with Slime Mould: If Physarum Built Motorways It Would Route M6/M74 through Newcastle. International Journal of Bifurcation and Chaos 20(10), 3065–3084 (2010)
4. Tero, A., Takagi, S., Saigusa, T., Ito, K., Bebber, D.P., Fricker, M.D., Yumiki, K., Kobayashi, R., Nakagaki, T.: Rules for Biologically Inspired Adaptive Network Design. Science Signalling 327(5964), 439 (2010)
5. Adamatzky, A.: Route 20, Autobahn 7 and Physarum Polycephalum: Approximating Longest Roads in USA and Germany with Slime Mould on 3D Terrains. arXiv preprint arXiv:1211.0519 (2012)
6. Jones, J.: Characteristics of Pattern Formation and Evolution in Approximations of Physarum Transport Networks. Artificial Life 16(2), 127–153 (2010)
7. Nakagaki, T., Guy, R.D.: Intelligent Behaviors of Amoeboid Movement Based on Complex Dynamics of Soft Matter. Soft Matter 4(1), 57–67 (2007)
8. Nakagaki, T., Yamada, H., Toth, A.: Maze-solving by an Amoeboid Organism. Nature 407(6803), 470 (2000)
9. Tero, A., Kobayashi, R., Nakagaki, T.: A Mathematical Model for Adaptive Transport Network in Path Finding by True Slime Mold. Journal of Theoretical Biology 244(4), 553–564 (2007)
10. Adamatzky, A.: Physarum Machines: Computers from Slime Mould. World Scientific Publishing Company Incorporated (2010)
11. Adamatzky, A.: Physarum Machines: Encapsulating Reaction-diffusion to Compute Spanning Tree. Naturwissenschaften 94(12), 975–980 (2007)
12. Jones, J.: The Emergence and Dynamical Evolution of Complex Transport Networks from Simple Low-level Behaviours. International Journal of Unconventional Computing 6(2), 125–144 (2010)
13. Gunji, Y.P., Shirakawa, T., Niizato, T., Haruna, T.: Minimal Model of a Cell Connecting Amoebic Motion and Adaptive Transport Networks. Journal of Theoretical Biology 253(4), 659–667 (2008)
14. Niizato, T., Shirakawa, T., Gunji, Y.P.: A Model of Network Formation by Physarum Plasmodium: Interplay between Cell Mobility and Morphogenesis. Biosystems 100(2), 108–112 (2010)
15. Gunji, Y.P., Shirakawa, T., Niizato, T., Yamachiyo, M., Tani, I.: An Adaptive and Robust Biological Network Based on the Vacant-particle Transportation Model. Journal of Theoretical Biology 272(1), 187 (2011)
16. Tero, A., Kobayashi, R., Nakagaki, T.: Physarum Solver: A Biologically Inspired Method of Road-network Navigation. Physica A: Statistical Mechanics and Its Applications 363(1), 115–119 (2006)

Cultural Algorithms for the Set Covering Problem

Broderick Crawford[1,2], Ricardo Soto[1,3], and Eric Monfroy[4]

[1] Pontificia Universidad Católica de Valparaíso, Chile
FirstName.Name@ucv.cl
[2] Universidad Finis Terrae, Chile
[3] Universidad Autónoma de Chile, Chile
[4] LINA, Université de Nantes, France
FirstName.Name@univ-nantes.fr

Abstract. This paper addresses the solution of weighted set covering problems using cultural algorithms. The weighted set covering problem is a reasonably well known NP-complete optimization problem with many real world applications. We use a cultural evolutionary architecture to maintain knowledge of diversity and fitness learned over each generation during the search process. The proposed approach is validated using benchmark instances, and its results are compared with respect to other approaches which have been previously adopted to solve the problem. Our results indicate that the approach is able to produce very competitive results in compare with other algorithms solving the portfolio of test problems taken from the ORLIB.

Keywords: Weighted Set Covering Problem, Cultural Algorithm, Genetic and Evolutionary Computation.

1 Introduction

The Weighted Set Covering Problem (WSCP) is a kind of problem that can model several real life situations [7, 3]. In this work, we solve some benchmarks of WSCP with an evolutive approach: Cultural Algorithms [13–15]. Cultural Algorithms are a technique that incorporates knowledge obtained during the evolutionary process trying to make the search process more efficient. Cultural algorithms have been successfully applied to several types of optimization problems [4, 10]. However, only a few papers had proposed a cultural algorithm for SCP and solving a few instances [6].

Here, the proposed approach is validated using 45 instances and its results are compared with respect to 8 other approaches. This paper is organized as follows: In Section 2, we formally describe WSCP using mathematical programming models. In section 3 we present the Cultural Evolutionary Architecture. In sections 4 and 5 we show the Population Space and the Belief Space considered to solve WSCP with Cultural Algorithms mantaining Diversity and Fitness knowledge. In Section 6, we present experimental results obtained when applying the algorithm for solving the standard benchmarks taken from the ORLIB [1]. Finally, in Section 7 we conclude the paper.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 27–34, 2013.

2 Problem Description

The WSCP is the NP-complete problem of partitioning a given set into subsets while minimizing a cost function defined as the sum of the costs associated to each of the eligible subsets [3]. In the WSCP matrix formulation we are given a $m \times n$ matrix $A = (a_{ij})$ in which all the matrix elements are either zero or one. Additionally, each column is given a weight (non-negative cost) c_j. We say that a column j can cover a row i if $a_{ij} = 1$. Let J denotes the set of the columns and x_j a binary variable which is one if column j is chosen and zero otherwise. The WSCP can be defined formally as follows:

$$Minimize \qquad f(x) = \sum_{j=1}^{n} c_j x_j \tag{1}$$

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1; \qquad \forall i = 1, \ldots, m \tag{2}$$

The goal in the WSCP is to choose a subset of the columns of minimal weight which covers every row.

3 Evolutionary Architecture: Cultural Algorithms

The Cultural Algorithms were developed by Robert G. Reynolds [13–15], as a complement to the metaphor used by Evolutionary Algorithms that are mainly focused on natural selection and genetic concepts. The Cultural Algorithms are based on some theories which try to model cultural as an inheritance process operating at two levels: a *Micro-evolutionary level*, which consists of the genetic material that an offspring inherits from its parent, and a *Macro-evolutionary level*, which is the knowledge acquired by the individuals through generations. This knowledge, once encoded and stored, it serves to guide the behavior of the individuals that belong to a population. Considering that evolution can be seen like an optimization process, Reynolds developed a computational model of cultural evolution that can have applications in optimization [4, 10]. He considered the phenomenon of double inheritance with the purpose of increase the learning or convergence rates of an evolutionary algorithm. In this model each one of the levels is represented by a space. The micro-evolutionary level is represented by the Population Space and the macro-evolutionary level by the Belief Space.

The *Population Space* can be adopted by anyone of the paradigms of evolutionary computation, in all of them there is a set of individuals where each one has a set of independent characteristics with which it is possible to determine his aptitude or fitness. Through time, such individuals could be replaced by some of their descendants, obtained from a set of operators (crossover and mutation, for example) applied to the population.

The *Belief Space* is the "store of the knowledge" acquired by the individuals along the generations. The information in this space must be available for the population of individuals. There is a protocol of communication established to dictate rules about the type of information that it is necessary to interchange between the spaces. This protocol defines two functions: *Acceptance*, this function extracts the information (or experience) from the individuals of a generation putting it into the Belief Space; and *Influence*, this function is in charge ”to influence” in the selection and the variation operators of the individuals (as the crossover and mutation in the case of the genetic algorithms). This means that this function exerts a type of pressure according to the information stored in the Belief Space.

3.1 Types of Knowledge

Knowledge that are important in the Belief Space of any cultural evolution model: Situational, Normative, Topographic, Historical or Temporal, and Domain Knowledge. According to Reynolds and Bing [15, 12], they conform a complete set, that is any other type of knowledge that is desired to add can be generated by means of a combination of two or more of the previous types of knowledge. The pseudo-code of a cultural algorithm is shown in Algorithm 1 [9]. Most of the steps of a cultural algorithm correspond with the steps of a traditional evolutionary algorithm. It can be clearly seen that the main difference lies in the fact that cultural algorithm use a Belief Space. In the main loop of the algorithm, we have the update of the belief space. It is at this point in which the belief space incorporates the individual experiences of a select group of members with the acceptance function, which is applied to the entire population.

Algorithm 1. Sketch of the Cultural Algorithm

```
Generate the initial population (Population Space)
Initialize the Belief Space
Evaluate the initial population
repeat
    Update the Belief Space (Acceptance function)
    Apply the variation operators (considering Influence function)
    Evaluate each child
    Perform selection
until the end condition is satisfied
```

4 Micro-evolutionary Level: Population Space

In the design and development of our cultural algorithm solving WSCP we considered in the Population Space a Genetic Algorithm with binary representation. An individual, solution or chromosome is a n-bit string, where a value 1 in the bit indicates that the column is in the solution and zero in another case. The initial population was generated with *size of the population* selected individuals randomly with a repair process in order to assure the feasibility of the individuals. For the selection of parents we used tournament. For the process of variation we used the operator of fusion crossover proposed by Beasley and

Chu [2], for mutation we used interchange and multibit. For the treatment of not feasible individuals we applied the repairing heuristic proposed by Beasley and Chu too [2]. In the replacement of individuals we use the strategy steady state and the heuristic proposed by Lozano et al. [11], which is based on the level of diversity contribution of the new offspring. The genetic diversity was calculated by the Hamming distance, which is defined as the number of bit differences between two solutions. The main idea is try to replace a solution with worse fitness and with lower contribution of diversity than the one provided by the offspring. In this way, we are working with two underlying objectives simultaneously: to optimize the fitness and to promote useful diversity.

5 Macro-evolutionary Level: Belief Space

In the cultural algorithm, the shared belief space is the foundation on which the efficiency of the search process depends. In order to find better solutions and improve the convergence speed we incorporated information about the diversity in the Belief Space. We stored in the Belief Space the individual with better fitness of the current generation and the individual who delivers major diversity to the population. With this type of knowledge Situational and Historical or Temporal, each one of the new individuals generated tries to follow a leader.

Initializing the Belief Space. A Situational-Fitness knowledge procedure selects from the initial population the individual with better fitness, which will be a leader in the Situational-Fitness space of beliefs. A Situational-Diverse knowledge procedure selects from the initial population the most diverse individual of the population, which will be a leader in the Situational-Diverse space of beliefs.

Applying the Variation Operators. Here we implemented the Influence of Situational-Fitness knowledge in the operator of Crossover. The influence initially appears at the moment of the election of the parents, the father 1 will be chosen with the method of binary tournament and the father 2 will be the individual with better fitness stored in the space of beliefs. Influence of Situational-Diverse knowledge in the operator of Crossover. This procedure works recombining the individual with better fitness of every generation with the most diverse stored in the space of beliefs, with this option we expect to deliver diversity to the population.

Updating the Belief Space. Updating the Situational Belief Space procedure, it implies that the Situational space of beliefs will be updated in all generations of the evolutionary process. The update of the Situational space of beliefs consists in the replacement of the individuals by current generation individuals if they are better considering Fitness and Diversity.

6 Experiments and Results

The performance of the algorithm was evaluated experimentally solving WSCP benchmarks from ORLIB [1]. Table 1 shows their detailed information. The first

Table 1. Problem instances

Problem	Number of constraints (m)	Number of variables (n)	Density(%)	Cost range	Best-known solution
wscp41	200	1000	2	[1-100]	429
wscp42	200	1000	2	[1-100]	512
wscp43	200	1000	2	[1-100]	516
wscp44	200	1000	2	[1-100]	494
wscp45	200	1000	2	[1-100]	512
wscp46	200	1000	2	[1-100]	560
wscp47	200	1000	2	[1-100]	430
wscp48	200	1000	2	[1-100]	492
wscp49	200	1000	2	[1-100]	641
wscp410	200	1000	2	[1-100]	514
wscp51	200	2000	2	[1-100]	253
wscp52	200	2000	2	[1-100]	302
wscp53	200	2000	2	[1-100]	226
wscp54	200	2000	2	[1-100]	242
wscp55	200	2000	2	[1-100]	211
wscp56	200	2000	2	[1-100]	213
wscp57	200	2000	2	[1-100]	293
wscp58	200	2000	2	[1-100]	288
wscp59	200	2000	2	[1-100]	279
wscp510	200	2000	2	[1-100]	265
wscp61	200	1000	5	[1-100]	138
wscp62	200	1000	5	[1-100]	146
wscp63	200	1000	5	[1-100]	145
wscp64	200	1000	5	[1-100]	131
wscp65	200	1000	5	[1-100]	161
wscpa1	300	3000	2	[1-100]	253
wscpa2	300	3000	2	[1-100]	252
wscpa3	300	3000	2	[1-100]	232
wscpa4	300	3000	2	[1-100]	234
wscpa5	300	3000	2	[1-100]	236
wscpb1	300	3000	5	[1-100]	69
wscpb2	300	3000	5	[1-100]	76
wscpb3	300	3000	5	[1-100]	80
wscpb4	300	3000	5	[1-100]	79
wscpb5	300	3000	5	[1-100]	72
wscpc1	400	4000	2	[1-100]	227
wscpc2	400	4000	2	[1-100]	219
wscpc3	400	4000	2	[1-100]	243
wscpc4	400	4000	2	[1-100]	219
wscpc5	400	4000	2	[1-100]	215
wscpd1	400	4000	5	[1-100]	60
wscpd2	400	4000	5	[1-100]	66
wscpd3	400	4000	5	[1-100]	72
wscpd4	400	4000	5	[1-100]	62
wscpd5	400	4000	5	[1-100]	61

column presents the problem code, the second and third columns show the number of constraints (m) and the number of variables (n). The fourth column shows the density (it is the percentage of non-zero entries in the WSCP matrix). The fifth column shows the range of costs of the variables. The last column presents the best known solution for each instance. Table 2 presents the results (the best cost obtained) when applying our algorithm for solving the WSCP benchmarks. The first two columns present the problem code and the best known solution for each instance. The following columns show the results applying Ant System (AS) and Ant Colony System (ACS) taken from [5] and Round, Dual-LP, Primal-Dual, Greedy taken from [8]. The next columns show the costs from Genetic Algorithms *(GA-1 and GA-2 using only the micro-evolutionary level)* with the basic proposal described in 4 without considering diversity. And the costs obtained by Cultural Algorithms *(CA-1 and CA-2 using micro and macro-evolution).* With GA-1 and

Table 2. Cost obtained using different algorithms

Problem	Best-known	AS	ACS	Round	Dual-LP	Primal-Dual	Greedy	GA-1	CA-1	GA-2	CA-2
wscp41	429	473	463	429	505	521	463	506	462	448	448
wscp42	512	594	590					609	582	642	603
wscp43	516							554	591	532	540
wscp44	494							551	577	512	512
wscp45	512							531	545	527	520
wscp46	560							574	620	568	605
wscp47	430							458	483	437	447
wscp48	492	524	522		522		499	560	549	609	548
wscp49	641							700	763	675	671
wscp410	514			539	664	669	556	548	596	553	533
wscp51	253	289	280	405	324	334	293	298	296	380	309
wscp52	302							353	335	320	330
wscp53	226							264	245	239	232
wscp54	242							281	265	244	250
wscp55	211							245	230	219	218
wscp56	213							243	224	232	227
wscp57	293							328	314	309	310
wscp58	288							326	315	306	311
wscp59	279							325	285	292	292
wscp510	265							292	280	277	278
wscp61	138	157	154	301	210	204	155	172	156	162	155
wscp62	146	169	163	347	209	232	170	162	162	188	171
wscp63	145	161	157				167	170	164	178	176
wscp64	131							145	138	133	141
wscp65	161							183	181	179	186
wscpa1	253			592	331	348	288	319	263	253	303
wscpa2	252			531	376	378	285	289	266	267	272
wscpa3	232			473	295	319	270	267	261	245	245
wscpa4	234			375	301	333	278	257	257	247	251
wscpa5	236			349	335	353	272	262	247	239	248
wscpb1	69			196	115	101	75	102	95	101	87
wscpb2	76			243	110	117	87	118	84	81	78
wscpb3	80			207	117	112	89	119	87	83	85
wscpb4	79							123	89	84	83
wscpb5	72							106	79	77	75
wscpc1	227			442	317	305	261	260	260	308	254
wscpc2	219			484	311	309	260	281	241	233	225
wscpc3	243			551	328	367	268	302	270	264	259
wscpc4	219			523	303	324	259	265	240	237	240
wscpc5	215							271	233	219	219
wscpd1	60			184	105	92	72	139	69	62	68
wscpd2	66			209	113	96	74	157	69	70	71
wscpd3	72			221	119	111	83	149	78	79	80
wscpd4	62							151	68	66	67
wscpd5	61							151	65	64	66

CA-1 we used Fusion Crossover and Mutation Interchange. With GA-2 and CA-2 we used Fusion Crossover and Mutation MultiBit. The algorithm has been run with the following parameters setting: size of the population (n)=100, size of the tournament (t)=2, number of generations (g)=30, with interchange and multibit mutation we affect the $5^0/_{00}$of the bits with a probability of mutation (p_m)=0.2. The algorithm were implemented using ANSI C, GCC 3.3.6, under Microsoft Windows XP Professional version 2002.

It is possible to observe that the incorporation of diversity in the Genetic Algorithm produced an improvement in the performance of the algorithm. The results indicate that the approach is able to produce very competitive results in compare with other approximation algorithms solving the portfolio of test problems taken from the ORLIB.

7 Conclusions

In this paper, we propose the use of knowledge of diversity to improve the performance of an evolutionary algorithm when solving the set covering problem. The executed experiments provided encouraging results in compare with other approaches. Our computational results confirm that incorporating information about the diversity of solutions we can obtain good results in the majority of the experiments. Our main conclusion from this work is that we can improve the performance of genetic algorithms considering additional information in the evolutionary process.

By other side, genetic algorithms tends to lose diversity very quickly. In order to deal with this problem, we have shown that maintaining diversity in the belief space we can improve the computational efficiency.

References

1. Beasley, J.E.: Or-library:distributing test problem by electronic mail. Journal of Operational Research Society 41(11), 1069–1072 (1990), `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`
2. Beasley, J.E., Chu, P.C.: A genetic algorithm for the set covering problem. European Journal of Operational Research 94(2), 392–404 (1996)
3. Chu, P.C., Beasley, J.E.: Constraint handling in genetic algorithms: The set partitioning problem. Journal of Heuristics 4(4), 323–357 (1998)
4. Coello, C.A., Landa, R.: Constrained Optimization Using an Evolutionary Programming-Based Cultural Algorithm. In: Parmee, I. (ed.) Proceedings of the Fifth International Conference on Adaptive Computing Design and Manufacture (ACDM 2002), University of Exeter, Devon, UK, vol. 5, pp. 317–328. Springer (April 2002)
5. Crawford, B., Castro, C.: Integrating lookahead and post processing procedures with aco for solving set partitioning and covering problems. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) ICAISC 2006. LNCS (LNAI), vol. 4029, pp. 1082–1090. Springer, Heidelberg (2006)
6. Crawford, B., Lagos, C., Castro, C., Paredes, F.: A cultural algorithm for solving the set covering problem. In: Melin, P., Castillo, O., Gómez-Ramírez, E., Kacprzyk, J., Pedrycz, W. (eds.) Analysis and Design of Intelligent Systems using Soft Computing Techniques. ASC, vol. 41, pp. 408–415. Springer, Heidelberg (2007)
7. Feo, T., Resende, M.: A probabilistic heuristic for a computationally difficult set covering problem. Operations Research Letters 8, 67–71 (1989)
8. Gomes, F.C., Meneses, C.N., Pardalos, P.M., Viana, G.V.R.: Experimental analysis of approximation algorithms for the vertex cover and set covering problems. Comput. Oper. Res. 33(12), 3520–3534 (2006)
9. Landa, R., Coello, C.A.: Optimization with constraints using a cultured differential evolution approach. In: GECCO 2005: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation, pp. 27–34. ACM Press, New York (2005)
10. Landa, R., Coello, C.A.: Use of domain information to improve the performance of an evolutionary algorithm. In: GECCO 2005: Proceedings of the 2005 Workshops on Genetic and Evolutionary Computation, pp. 362–365. ACM Press, New York (2005)

11. Lozano, M., Herrera, F., Cano, J.R.: Replacement strategies to preserve useful diversity in steady-state genetic algorithms. In: Proceedings of the 8th Online World Conference on Soft Computing in Industrial Applications (September 2003)
12. Peng, B.: Knowledge and population swarms in cultural algorithms for dynamic environments. PhD thesis, Detroit, MI, USA, Adviser-Reynolds, R.G. (2005)
13. Reynolds, R.: An introduction to cultural algorithms. In: Third Annual Conference on Evolutionary Programming, pp. 131–139 (1994)
14. Reynolds, R.G.: Cultural algorithms: theory and applications. In: New Ideas in Optimization, pp. 367–378. McGraw-Hill Ltd., Maidenhead (1999)
15. Reynolds, R.G., Peng, B.: Cultural algorithms: Modeling of how cultures learn to solve problems. In: ICTAI, pp. 166–172. IEEE Computer Society (2004)

Impulse Engine Ignition Algorithm Based on Genetic Particle Swarm Optimization

Xiaolong Liang[1], Zhonghai Yin[2], Yali Wang[2], and Qiang Sun[2]

[1] Air Traffic Control and Navigation College,
Air Force Engineering University, Xi'an, China
xiaolong.liang@hotmail.com
[2] Science College, Air Force Engineering University, Xi'an, China

Abstract. The Concerning the problem of near space target intercepting, double-goal optimization model is established in order to maximize the response pace and to minimize the engine consumption at the same time. Impulse engine ignition algorithm based on Genetic Particle Swarm Optimization is presented. The global optimal solution of the optimal problem could be obtained fast by this algorithm. Simulation results indicate that the Ignition control algorithm can obtain high control precision with low engine consumption.

Keywords: Kinetic Interceptor, Composite Control, Ignition Control Algorithm.

1 Introduction

Near space is the region of the surface height of 20~100km area, between convention aviation and spaceflight area[1]. Auto-interception target in near space, because of near space is low air density, long range, high precision, the traditional pneumatic rudder control is slow response speed, and it can’t provide enough available overload, thus, it can intercept target of highly maneuverable or high speed, it is an effective method use of direct force and aerodynamics force compound control[2]. Generally, direct force and aerodynamics force compound control that is decoupled direct force and aerodynamics force, and design control loop respectively, then control signal is decomposed by the control allocation algorithm, it is performed by the direct force automation system and aerodynamic control system. In the direct force of the automatic control system design, how to effectively control the direction and size of the pulse vectors generated by the engine power has become one of the important issues of the compound control system design. In this paper, the pulse ignition control algorithm is mainly studied; impulse engine ignition algorithm based on Genetic Particle optimization algorithm is presented. The method has the advantages of high accuracy, high speed and so on.

2 Compound Control Strategy

As the near space intercept airspace may be greater, it can reach up to dozens of or even hundreds of kilometers. At this height, the atmosphere is very thin, relying

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 35–43, 2013.

solely on the torque generated by the aerodynamic rudder have been unable to provide sufficient control torque. Then you can use direct force system which is provided to control torque. When the interceptor height is lower than a certain value, the aerodynamic rudder to provide control torque is sufficient to accomplish the control tasks; we can be fully use aerodynamic rudder to control. When the flight altitude is higher than 80km, the aerodynamic forces can be ignored completely for the air very thin, full use of direct force to control; There is a transitional phase between these two phases that is full use of direct force control and full use of aerodynamic rudder to control. Full use of direct force control and full use of aerodynamic rudder to control these two phases there is a transitional stage, complete the flight control by direct force and aerodynamic rudder together.

Compound control interception principle is shown in Figure 1. Trajectory control commands are generated by the guidance law, by aerodynamic force, direct force control mechanism to perform respectively. The core of the compound control is how to reasonable decompose control instructions that control distribution, and make full use of the pulse engine, to ensure a successful intercept of strong maneuvering ability goals.

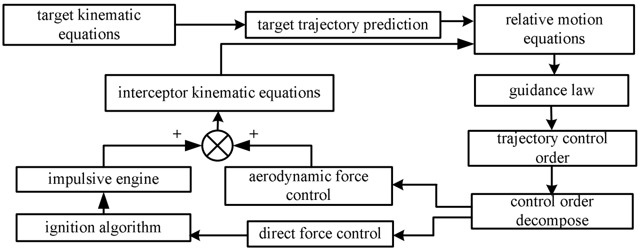

target kinematic equations
target trajectory prediction
relative motion equations
interceptor kinematic equations
guidance law
+
+
trajectory control order
impulsive engine
aerodynamic force control
control order decompose
ignition algorithm
direct force control

Fig. 1. Composite Control Model

If the interceptor missile is considered as particle, assume that the acceleration of gravity size is constant, direction of vertical down, then the interception missile center of mass dynamics model for composite control is

$$m\begin{bmatrix}\dot{V}_x\\ \dot{V}_y\\ \dot{V}_z\end{bmatrix}=\begin{bmatrix}0\\ 0\\ mg\end{bmatrix}+L_{al}\begin{bmatrix}F_D\\ F_C\\ F_L\end{bmatrix}+L_{ml}\begin{bmatrix}P\\ F_C^{'}\\ F_L^{'}\end{bmatrix}+F_\varepsilon \tag{1}$$

Where L_{al} is a transformation matrix which generated by the airflow coordinates to launch coordinate, F_D , F_C , F_L is the aerodynamic drag, lateral force and lift by intercept, L_{ml} is a transformation matrix which generated by the body coordinate system to launch coordinate. P is the thrust of the tail booster, $F_C^{'}$, $F_L^{'}$ is the direct force of pulse engine by the interceptor missile, F_ε is a variety of disturbance force vector sum in launch coordinate system.

Considered Radar, infrared composite seeker is used in the interceptor missile, the relative distance between interceptor missile and target can be measured which can also be estimated by the remaining time of flight using angle information. The threshold of start-up of composite control by direct force/aerodynamic force is determined by the relative distance (the remaining time of flight). The method can satisfy the special demand that the interceptor missile directive collision with near space target warhead. The target mobile has barely effect on the performance of composite control. Considered the height information of interceptor missile is obtained (also can be determined by the flight height). When the height led to air power shortage then direct force control is launched[3].

3 Impulse Engine Ignition Model

Micro pulse engines on the interceptor missile working is *just open*. The working period of these small engine is generally only about 20ms, the small engine have worded can't be reused, thrust size of pulse engine is not adjustable, and it's opening and closing requires a certain amount of time[4,5]. Engine along the body surface circumferential uniform distribution installation, can produce control force in all directions. To make full use of these pulse engines, the interceptor can be designed to be a constant low speed rotation, by selecting the appropriate opening strategy to complete the attitude control.

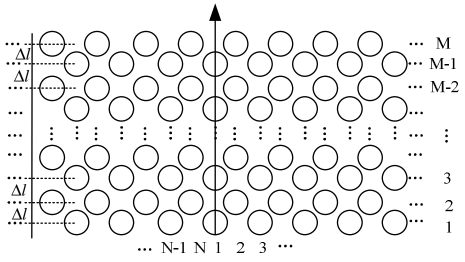


Fig. 2. Pulse engine ring spread display

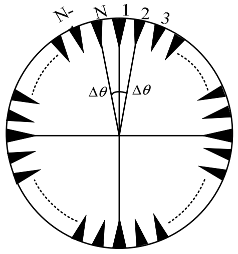


Fig. 3. Pulse engine layout display

Assuming the direct control power that control command distribution in missile coordinate system Y axis and Z axis components respectively is $F'_{y'}$, $F'_{z'}$, then the size and direction of the resultant of forces is:

$$F' = \sqrt{F'_{y'} + F'_{z'}} \tag{2}$$

$$\theta = \arctan\left(F'_{z'} / F'_{y'}\right) \tag{3}$$

Where θ is the angle between resultant of forces and the Y-axis. During the guidance flight, only according to angle θ ignite small engine from the nearest, the numbers can be identified by F' .

To determine the number of engine to be open according to the size of resultant of forces in ignition control algorithm, according to the direction of resultant of forces combined with the transient position of the current engine decided to open the position of the engine on the airframe. For the modeling and convenience for solution, the pulse engine in different spatial positions is treated with the secondary equivalent force. For the first time a single engine is equivalent processing, and second spatial location is equivalent treated.

A. *single engine equivalent force*

Assuming that each pulse engine after ignition combustion duration t is equal, produced by the same impulse *I*. the average thrust generated by a single engine for $F{=}I\,/\,t$. Because the body is rotating around its own vertical axis, so the scanning angle is $\gamma{=}\omega t$ during the engine injection progress. The equivalent reaction resultant force that is generated from small nozzle on the bisector of angle γ, its size be

$$F_{e1} = 2/\gamma\int_0^{\gamma/2} F\cos\varphi d\varphi = F\sin(\omega t\,/\,2)/\omega t\,/\,2 = \eta\cdot F \tag{4}$$

Where η called the efficiency coefficient, when $\omega{=}3r\,/\,S$, $t{=}18\text{ms}$, $\eta{=}99.52\%$.

According to the concept of equivalent resultant force, in micro engine injection procession the magnitude and direction of the force generated are fixed, the size can be obtained by the formula (4), the direction parallels bisector of scan angle.

B. *Single Engine Equivalent Force in Space*

Average jet force generated by single engine is F_{e1}, and multiple engine ignite at the same time without considering the influence between jet. Assume that are *M* rings, there are *N* engines on each ring, shown in Figure 2, 3. The moment equivalent process must be carried to the jet force which is produced by engines on every circle because the different distance between the centroid and engines on every circle. All the forces take the moment relative to the centroid, the distance between the engine center on the first ring and centroid is the reference length l, the distance from intersection to the centroid is $l_i = l + (i-1)\times\Delta l$, where the intersection is that line of action of direct force of the *i*-th circle engine cut axis of symmetry of the body. Supposing that the size of single engine on the *i*-th circle's equivalent jet force be called F_{e2}^i, there are

$$F_{e2}^i = F_{e1}\cdot\left(l + (i-1)\times\Delta l/l\right),\ (\,l{>}0\ ,\quad i{=}1,2,\cdots,M\ ) \tag{5}$$

$$F_{e2}{=}F_{e1}\ ,\ (\,l{=}0\,) \tag{6}$$

The generated force by specified means (force to be generated) is $F^{'}$, take the average jet force F_{e1} generated by single engine as reference, we can obtain be separately treated for generated force and the equivalent jet force of engine:

$$\bar{F}^{'} = F^{'}/F_{e1} \tag{7}$$

$$\bar{F}_{e2}^i = F_{e2}^i/F_{e1} = \left(l + (i-1)\times\Delta l\right)/l \qquad (\ i{=}1,2,\cdots,M\ ) \tag{8}$$

A bi-objective pulse ignition model is established with response time speediness (i.e. which produce large direct force) and minimize engine consumption.

$$\begin{aligned}
&\max\ G_1 = \sum_{i=1}^{N}\sum_{k=1}^{M}\bar{F}_{e2}^i S_k^i\\
&\min\ G_2 = \sum_{i=1}^{N}\sum_{k=1}^{M} S_k^i\\
&st.\ \left|\sum_{i=1}^{N}\sum_{k=1}^{M}\bar{F}_{e2}^i S_k^i - \bar{F}'\right| \le \varepsilon\\
&S_k^i = \begin{cases} 0 & \textit{it has not yet start or has been used once}\\ 1 & \textit{start}\end{cases}
\end{aligned} \tag{9}$$

Where G_1, G_2 is the objective function, the former is resultant force generated with open the engine, the latter is the total number to start the engine; *M* is the rings' number and *N* is the engines' number; ε is permissible error limitation of $\bar{F}$; S_k^i is the switching function which can be defined as: when the *k*-th pulse engine on the *i*-th ring has not yet open or has been used once, $S_k^i = 0$; when the *k*-th pulse engine on the *i*-th ring is turned on, $S_k^i = 1$.

4 Impulse Engine Ignition Algorithm Based on Genetic Particle Swarm Optimization

It can be seen that the pulse ignition issues are multi-objective 0-1 programming problem from the mathematical model, a single-objective optimization problem is deducted from a bi-objective optimization problem to facilitate the computation.

$$\begin{aligned} &\max \quad G = G_1 / G_2 = \sum_{i=1}^{N}\sum_{k=1}^{M} \bar{F}_{e2}^i S_k^i \Big/ \sum_{i=1}^{N}\sum_{k=1}^{M} S_k^i \\ &st. \quad \left| \sum_{i=1}^{N}\sum_{k=1}^{M} \bar{F}_{e2}^i S_k^i - \bar{F}' \right| \le \varepsilon \\ &S_k^i = \begin{cases} 0 & \textit{it has not yet start or has been used once} \\ 1 & \textit{start} \end{cases} \end{aligned} \tag{10}$$

At present, the main methods to solve 0-1 programming problem are: dynamic programming, the ant colony algorithm, the greedy algorithm and simulated annealing algorithm, DNA algorithm, branch-and bound method, the particle swarm optimization algorithm, the artificial neural network, recursive method, backtracking algorithm, clonal selection algorithm, tabu search algorithm and GA, hybrid algorithm, etc. The study of computation complexity shows that 0-1 knapsack problem is a classic NP problem, there is no perfect method, and all intelligent algorithms are solved in a certain range. Impulse engine ignition algorithm based on Genetic Particle Swarm Optimization is given below.

Particle Swarm Optimization[6] (PSO) is a new evolutionary algorithm for global optimization developed by Kennedy and Eberhart, which originated from the simulation of prey behavior of the birds. PSO algorithm[7] is

$$V(t+1) = \omega V(t) + c_1 rand(p_{\text{best}}(t) - x(t)) + c_2 rand(g_{\text{best}}(t) - x(t)) \tag{11}$$

$$x(t+1) = x(t) + V(t+1) \tag{12}$$

In each iteration, the particles update itself through tracking two extreme: one is the optimal position found by the particle itself that is individual extreme p_{best}; another is the optimal location found currently by the whole group namely global extreme g_{best}. where the *i*-th particle is an n-dimensional vector $x_i = (x_{i1}, x_{i2}, \cdots, x_{in})$, that is the position of the *i*-th partial is x_i.Each particle represents a potential solution, V is the velocity of a practical, *rand* is a random number between [0,1], c_1 and c_2 are learning factors, ω is inertia weight.

From the model of pulse engine ignition, we can get that the problem is a combination optimization problem of the 0-1 knapsack with discontinuous and discrete, the iterative formula of the basic PSO algorithm obviously can't be applied directly. Therefore, the equal value exchange and the different value exchange sequence is composed of the two exchange results are introduce to describe the discrete PSO to the problem.

A. Equivalent Exchange and Different Value Exchange, Exchange Sequences

In the impulse engine ignition algorithm, each engine in ignition algorithm corresponds to a particle in the PSO algorithm, i.e. component value on the d-th-vector of any particle i are only two possible values: 1 means that the engine ignition is enabled, 0 indicates that ignition isn't enabled.

Definition1. The d-th dimension component of the particle i before and after one iteration keep constant is named the equivalent transformation, denoted as E . If the exchange is operated between 0 and 1, the exchange is called the different value exchange, denoted as D. The particles spatial exchange locations before and after iteration is named the exchange sequence, denoted as S.

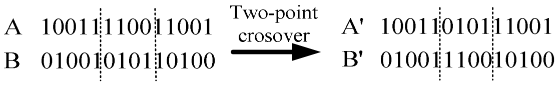
Obviously, the exchange sequence is composed of E and D, denoted as S_i : $S_i=(E,E,D,D,\cdots)$. Two exchanges can be arbitrary arranged in the sequence. Therefore, the location of a particle before and after one iteration can be represented by a specific exchange sequence, $S=(S_1,S_2,\cdots,S_d,\cdots,S_n)$.

B. Particle Update

① **Velocity Update.** Equation (11) means that the speed doesn't apply to the ignition problem of impulse engine .Considering the GA successful applying to solve combinatorial optimization problems, it adopts the idea of crossover and mutation in GA.

Crossover operator $c_1 rand(p_{\text{best}}(t)-x(t))+c_2 rand(g_{\text{best}}(t)-x(t))$ in (11) can be seen the crossover of GA, the current solutions respectively take crossover with the individual extreme and the global extreme, the solution produced as a new position. The proposed crossover operator which has more new structure generation ability and less breakage for good mode is a two-point crossover operator.

First, c_1 , c_2 is defined as a probability combination, P_{c1} and P_{c2} are respectively denote individual extreme and global extreme probability of crossover operation, generally speaking, $P_{c1}=P_{c2}$. Two individual select at random from the population to pairing, then in individual coding series, two crossing points are set random, according to given the crossover probability P_{c1} and P_{c2} , coding between the two crossing points exchange each other, and then resulting in two new individuals.



Mutation operator $\omega V(t)$ in (11) can be seen the mutation of GA, its essence is a different value to exchange D . This paper adopt non-uniform mutation operator, the specific process is as follows:

Inertia weight coefficient ω is defined as a combination of probability, the probability of the different value exchange D corresponding to the i-th location in the velocity permutation sequence is

$$P_{\omega}=(P_1,P_2,\cdots,P_i,\cdots,P_n)\ ,\ \ P_i\in[0,1]$$

where $P_i=1/(1+\exp(-{}^{t}\!/_{T}\lambda_i))$, t is the current evolving generation, T is the maximum evolving generation, λ_i is a parameter for determining the degree of non-uniform, plays a role on regulating local search area. For λ_i , the effect of $t_i^{'}$, a_i , p_i must be considered, set $\lambda_i=t_i'/(\alpha a_i+\beta p_i)$, viz., the objective value in (10) get maximum with least engine and maximal generated force.

Therefore, the larger λ_i , the closer optimal ignition results, in which case the probability of variation should be smaller; on the contrary, the probability of variation should be greater.

The speed formula though crossover operator and mutation operator handling can be expressed as

$$V(k+1)=P_{\omega}V(k)+P_{c1}(p_{best}(k)-x(k))+P_{c2}(g_{best}(k)-x(k)) \tag{13}$$

② **Location Update.** Speed item is an exchange sequence in (12), the exchange sequence acting on position of the particle before iteration is lead to position changing of particle before and after iteration. It can be obtained the relation between the position after the k-th iteration and the position $x(k)$ before the iteration with this exchange sequence $V(k)$ as

$$x(k+1)=x(k)+(S_{k1},S_{k2},\cdots,S_{kd},\cdots,S_{kn}) \tag{14}$$

It can be obtained that two position difference is a switching sequence in iteration process of the particle.

C. Constraints Processing

Considering the optimal solution if the constraints optimization problems are generally located in the constraint boundary or near the boundaries, part of the acceptable infeasible solutions and feasible solutions is compared according to the value of the objective function, in order to retained part of infeasible solution particle with the better performance in the group. The following comparison criteria is adopted in this paper

① When two particles are feasible particle, comparing the value of the objective function, the smaller objective function value , the superior the individual.

② When a particle is feasible and the other particles isn't feasible, the best infeasible solution join in the feasible solution with a certain probability, comparing their objective function value, the particle with the smaller function value is excellent.

D. Algorithm Procession

Through the above analysis, a modified particle swarm algorithm is presented in this paper, the algorithm process can be described as follows:

Step 1. Initialization particle population, the position of each particle randomly generated by $x_{ij}(0)=\begin{cases}0, rand(0,1)<0.5\\1, rand(0,1)\geq 0.5\end{cases}$, the speed randomly generated by $V_{ij}(0)=V_{\min}+rand(0,1)(V_{\max}-V_{\min})$;

Step 2. Set the particle p_{best} as the current personal best position and g_{best} as position of the best particle in the group;

Step 3. Judging the stopping criteria of algorithm(usually this is a good enough objective function fitness or reaches a preset maximum number of iteration) whether is satisfied, if it is met then transferred to Step 7, otherwise transferred to Step 4;

Step 4. Take crossover and mutation operators for all the particles, and in accordance with (13), (14) to update the speed and position of the particle, to calculate the objective function's fitness by (10);

Step 5. According to the above standard of comparison of the particles updates each particle p_{best};

Step 6. Update the best position of all particles, i.e. the globe extremum g_{best}, then transferred to Step3;

Step 7. Output g_{best}, algorithm stops.

5 Simulation Example and Conclusion

Assuming interceptor missile have 36 pulse engines, space layout parameters are N=6 , M=6, the average jet force produced by each engine is F_{e1}=250 N. Aim at pulse engine ignition mathematical model, greedy algorithm and genetic particle swarm algorithm is used for solving respectively. Table 1 shows the 8 different results of expectative force, it can be seen that genetic particle swarm optimization algorithm has higher precision under the same calculation conditions, engine consumption is less, it is slower than the greedy algorithm, but it can also meet the real time requirements of the interceptor missile.

Table 1. The comparison of two ignition algorithm simulation results

expectative force /N	Greedy algorithm			Genetic Particle Swarm Optimization algorithm		
	Jet force/N	Numbers of engine	Cost time/s	Jet force/N	Numbers of engine	Cost time/s
100	95.26	2	5.6×10^{-6}	96.3	2	6.2×10^{-6}
200	196.73	2	9.6×10^{-6}	796.1	2	9.9×10^{-6}
300	290.8	1	1.06×10^{-5}	298.4	1	2.24×10^{-5}
500	516.7	4	1.97×10^{-5}	507.6	3	2.64×10^{-5}
800	809.7	5	2.24×10^{-5}	799.4	4	2.94×10^{-5}
1200	1193.5	6	3.48×10^{-5}	1198.6	4	3.64×10^{-5}
2500	2497.8	10	3.98×10^{-5}	2503.1	9	4.24×10^{-5}
6000	6016.7	22	4.22×10^{-5}	5989.4	20	4.94×10^{-5}

Direct force/ aerodynamic composite control distribute the control instruction using relative distance (the remaining flight time). Impulse engine ignition algorithm based on Genetic Particle Swarm Optimization can give full play to the ability of direct force, save the number of pulse engines use. The ignition algotithm and comopsite control system be perfect match, the simulation results show that impulse engine ignition algorithm based on Genetic Particle Swarm Optimization satisfied the composite control requirements.

References

1. Li, Y., Shen, H.: Key Technologies of Developing Near Space Aerocraft Systems. Journal of the Academy of Equipment Command & Technology 17(5), 52–55 (2006) (in Chinese)
2. Zhu, L., Tan, G.G., Yu, M.: A Survey of Combined Control Methods with Lateral Thrust and Aerodynamic Force for Antiaircraft Missiles. Journal of Projectiles, Rockets, Missiles and Guidance 29(1), 11–14 (2009)
3. Yin, Y., Yang, M., Wang, Z.: Modeling and Control of the Interception Missile by Combined Control of Lateral Thrust and Aerodynamic Force. Aerospace Control 24(4), 18–21 (2006) (in Chinese)
4. Jitpraphai, T., Costello, M.: Dispersion Reduction of a Direct Fire Rocket Using Lateral Pulse Jets. AIAA, USA (2002)
5. Hirokawa, R., Sato, K.: Autopilot design for a missile with reaction-jet using coefficient diagram method. In: AIAA Guidance, Navigation, and Control Conference, pp. 739–746 (2001)
6. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, Piscataway, pp. 1942–1948. IEEE Press, New York (1995)
7. Liang, X., Feng, J., Yang, Y.: Optimal control of switched systems numerical approach based on particle swarm algorithm. Systems Engineering and Electronics 31(3), 642–645 (2009) (in Chinese)

Learning by Imitation for the Improvement of the Individual and the Social Behaviors of Self-organized Autonomous Agents

Abdelhak Chatty[1,2], Philippe Gaussier[2], Ilhem Kallel[1], Philippe Laroque[2], and Adel M. Alimi[1]

[1] REGIM: REsearch Groups on Intelligent Machine,
National School of Engineers (ENIS), Sfax University, Sfax, Tunisia
[2] ETIS: Neuro-cybernetic team, Image and signal Processing,
National School of Electronics and Its Applications (ENSEA),
Cergy-Pontoise University, Paris, France

Abstract. This paper shows that learning by imitation leads to a positive effect not only in human behavior but also in the behavior of the autonomous agents (AA) in the field of self-organized creation deposits. Indeed, for each agent, the individual discoveries (i.e. goals) have an effect on the performance of the population level and therefore they induce a new learning capability at the individual level. Particularly, we show through a set of experiments that adding a simple imitation capability to our bio-inspired architecture allows increasing the ability of agents to share more information and improving the overall performance of the whole system. We will conclude with robotics' experiments which will feature how our approach applies accurately to real life environments.

Keywords: Learning by Imitation, Cognitive Map, Emergent Structures.

1 Introduction

Swarm-based systems [1, 2] are now a classical approach to dealing with collective intelligence problems. In such approaches, to produce global emergent behaviors for Autonomous Agents (AA) the interaction between agents needs not to be complex [3, 4]. Based on this idea, researchers have been able to design a number of successful algorithms in the field of self-organized creation deposits. [5] has proposed a model relying on biologically plausible assumptions to account for the phenomenon of the clustering of dead bodies performed by ants. [6] showed that acting on objects simplifies the reasoning needed by a multi-agent system and allows the deposit of scattered objects. In the same field of creation deposits we assume an environment composed of several animats and three plants (A, B and C). The AA are motivated by the simulation of three types of needs related to the three plants and each need can be satisfied by a corresponding plant. The level of each type of need is internally represented by an essential variable, $e_i(t)$ whose value is in [0; 1] and varies with time $de_i/dt = -\alpha_n e_i(t)$, where α_n represents the decreasing rate of the essential variable. If a plant from the corresponding type has not been found, the agent dies. Thus, in order to maintain the

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 44–52, 2013.

satisfaction level of our AA, keep them alive and optimize their planning, instead of only navigating between the three plants, it would be interesting if the agents were able to create relevant warehouses. So as to improve the performance of the system, several benefits can be expected from the imitation capability [7–9]. It can be considered as a powerful skill that would enable the AA to learn and discover new tasks and places. Learning by imitation is an intuitive and natural method, it is not only a skill useful for learning but also a way to speed up the learning process. Therefore, the researchers consider imitation as a powerful behavior which enables learning by observation even if the imitation was not intentional (i.e. imitation emerging from the ambiguity of the perception in a simple sensori-motor system) [7]. The aim of this work is to show the positive effect of the learning by imitation on the improvement of the performance of the deposits' system. To validate our system we performed a series of experiments with simulated agents and with swarm robots. The remainder of this paper is organized as follow: in section 2 the bio-inspired architecture is presented. Section 3 describes the behaviors of the AA. Sections 4 and 5 are devoted respectively to the description of the relevance of the emerging warehouses and to the explanation of the imitation process. Before concluding, section 6 and section 7 are concerned respectively with the analysis of the positive feedback induced by the imitation strategy in AA and in swarm robots.

2 The Bio-inspired N.N Architecture of the Agent

Starting from neurobiological hypotheses on the role of the hippocampus in the spatial navigation, [10] produced a model of the cognitive map in the hippocampus representing the entire environment and not only the shortest paths to a given goal. The work of [11] revealed special cells in the rodents' hippocampus that strike off when the animal is at a precise location. These neurons have been called place cells (PC). In our model, we do not directly use PC we rather use neurons called transition cells (TC) [12]. A transition cell encodes for a spatio-temporal transition between two PCs consecutively winning the competition, respectively at time t and $t + \delta$t. The set of the PCs and the TCs constitute a non-cartesian cognitive map. A schematic view of the architecture of our AA is shown in fig. 1.

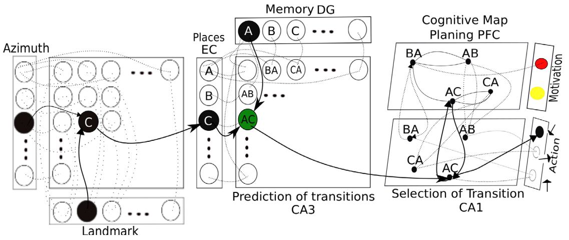
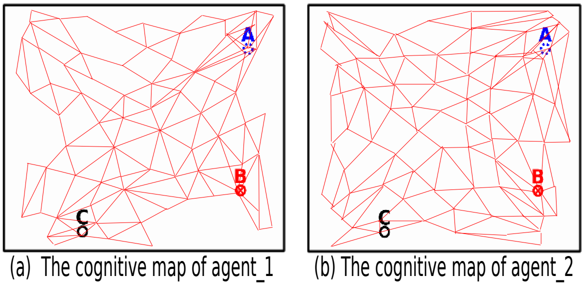


Fig. 1. Model of hippocampo-cortical for the building of an agent cognitive map

To create the PC, the agent takes, a visual panorama of the surrounding environment. The views are processed to extract visual landmarks. After learning these landmarks, a visual code is created by combining the landmarks of a panorama with their azimuth. This configuration serves as a code for PCs. The signals provided by the EC (the entorhinal cortex) are solely spatial and consistent with spatial cells activities. Spatial cells activities are submitted to a Winner-Take-All competition in order to only select the cell with the strongest response at a specific location. We will subsequently speak about the current location by indicating the spatial cell which has the highest activity at a given location. The temporal function at the level of the DG (dentate gyrus) is reduced to the memorization of a previous location. The acquired association at the level of CA3/CA1 (the pyramidal cells) is then the transition from a location to another in addition o the information concerning the time spent on carrying out this transition. Once the association from the previous location and the new one is learned, every new entry will reactivate the corresponding memory in the DG. During exploration of the environment, the cognitive map is gradually created as the agent moves (see fig. 2).



(a) The cognitive map of agent_1 (b) The cognitive map of agent_2

Fig. 2. The cognitive map of two agents at 5104 time steps. The shape of the cognitive map in (a) and (b) proves that its construction is related to the agent's own perception.

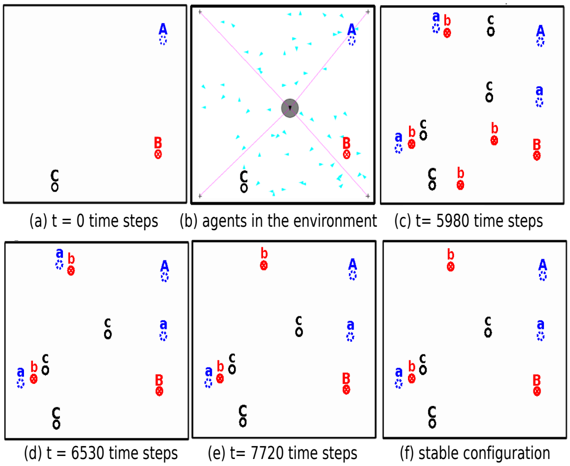
3 The Behaviors of the Autonomous Agents

In the context of situated cognition, local rules can lead to create emergent structures allowing the creation of warehouses relevant to the sorting strategy used by [5]. We propose local generic rules depending on the number of withdrawals and deposits of warehouses according to the number of agents perceived. The agent is indeed ought to favor the creation of warehouses in locations which contain other agents rather than empty regions. Thus, the perception of local agents is responsible for controlling the rules of withdrawals and deposits. The condition for withdrawing is computed by $Pr_{(Taken)} = \exp^{-\lambda N_A}$ where N_A is the number of agents in the neighborhood, λ is a positive constant: The probability that an agent needs to take some plant goods increases when it perceives that the plant is less used by other agents. So, the more agents are near a plant, the less withdrawal there is and vice versa. The condition for deposits is computed by $Pr_{(Deposits)} = (1 - \exp^{-\alpha N_A}) * (1 - \exp^{-\beta t})$ where α, β are environmental factors, N_A is the number of agents in the neighborhood and t is the time since the taking: the probability of deposit increases with time and distance from the origin plant (when the agent is far from the origin plant where it took the last goods) and it depends on the number of the agents in the neighborhood (when the current place of the agent is frequented by other agents). For this reason, we tried to restrict the ability of agents to perceive the environment. Moreover, the deposit operation is also built on the concept of refueling : the agent puts goods in the warehouses that already exist.

4 The Emergence of Warehouses

We use an environment with 60 agents (under 20 agents, the system will not be able to create a fixed number of warehouses in fixed places) and three original plants. This environment is continuous and in order to cross it diagonally, an agent needs 200 time steps. This experiment is just a general scenario to describe the behavior of the agents and the creation of relevant warehouses.

The agents start to move randomly in the environment, with a limited range of visual perception (see Fig. 3b). While passing through a plant, an agent increases its level of satisfaction and applies the local rules of taking and transporting a quantity from the associated product. If the decision of transporting some products is taken, the agent continues its journey. If deposited, goods represent a new warehouse (which allows 10 visits for agents) permitting others to increase



(a) t = 0 time steps (b) agents in the environment (c) t= 5980 time steps

(d) t = 6530 time steps (e) t= 7720 time steps (f) stable configuration

Fig. 3. The figure shows an example with 3 plants A, B and C whose positions are fixed and can deliver an unlimited amount of products. They provide warehouses of 3 different kinds ("a" is a warehouse-type A, "b" is a warehouse-type B and "c" is a warehouse-type C). Through the individual deposit process of agents, warehouses emerge and become stable after a while.

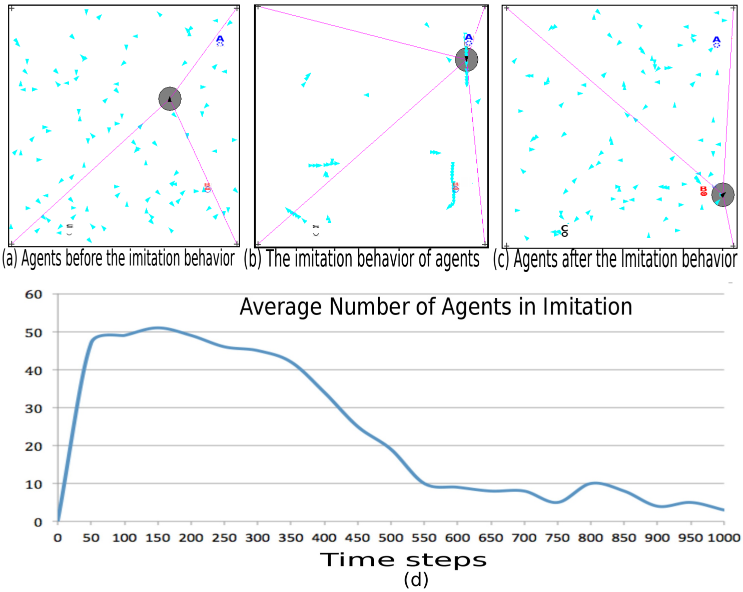
their satisfaction level. Agents also have the possibility of refueling warehouses by adding products to them (the available products in the warehouses will increase). This provides stability for warehouses in relevant locations which are close to several agents in order to prevent loss. However, warehouses which are abandoned or poorly visited will eventually disappear since the amount of goods available will decrease rapidly. Fig. 3c and d show the disappearance of isolated warehouses. When a planning agent tries to reach a warehouse and realizes that it has disappeared, the agent dissociates the current PC from the formerly-corresponding warehouse and resets the motivation to 0. Hence the PC does not fire any more when the agent feels the need for this warehouse. Similarly, when a new matching warehouse is discovered, the paths leading to the warehouse are immediately reinforced, modifying the cognitive map synchronously with the environment. In fig. 3e the AA converges to a stable solution with a fixed number of warehouses in fixed places at 7720 time steps and remains the same for more than 20000 time steps (see fig. 3f).

5 The Imitation Process

Collective learning experience gathered by individual agents cannot be of use to others, if there is no means to communicate information between agents. One way to add knowledge transmission among agents is to make them able to imitate one another. Thus, at the individual level, agents can rely on an on-line, continuous building of a cognitive map whose structure depends on their own experience and discovery of the environment; and at the social level, they can take advantage of the ability to imitate one another using simple agent-following strategies to transmit parts of one agent's cognitive map to another's, leading to some kind of naturally distributed knowledge, similar to what can be achieved in swarm-intelligence systems, except that shared knowledge does not use the physical space as repository but uses individual cognitive maps instead. We implemented and studied a simple imitation strategy based on the azimuth: if a single agent is visible, it becomes the chosen imitation target; if several agents are visible, then the chosen target is the closest one to the direction of the agent which tries to imitate. The decision of imitation is controlled by a probabilistic function (see eq. 1): the probability of imitation decreases when the number of discovering of different kind of resources(plants/warehouses) N_D increases.

$$Pr_{(Imitation)} = \eta \exp^{-\rho N_D} \tag{1}$$

η and ρ are positive values. To better understand the imitation behavior of agents, fig. 4a shows that the agents represented by triangles, started to move randomly in the environment. Due to the imitation behavior fig. 4b shows that subgroups of agents appeared in the environment. According to the curve (d), this phase coincides with an increase in the number of agents in imitation. Indeed, the agents try to follow each other so as to localize and learn the resources' positions. We can also see that some subgroups, have succeeded to discover the plant A and B and to learn their positions. After discovering all kinds of resources (where the average time is 505 time steps. However without imitation, the average time to reach the resources is 1683 time steps) the probability of the imitation will decrease (see fig. 4c). Thus, the number of agents in imitation shown by the curve (d) also decreases, since agents are now able to return to the resources on their own.



(a) Agents before the imitation behavior (b) The imitation behavior of agents (c) Agents after the Imitation behavior

(d)

Fig. 4. The evolution of the agents' number in imitation through 1000 time steps

6 The Positive Feedback of the Imitation Behavior

The behaviors of the AA are not deterministic because they do not always provide the same results on several tests with the same parameters. In order to obtain complete results, we kept the same previous parameters ($\alpha = \beta = 0.3, \lambda = 0.7, \eta = 0.1, \rho = 3, ST = 50$ and the number of agents is 60) and we conducted ten tests of AA for each experiment until 20000 time steps (the number of tests is determined by the statistical Fisher test). Through the imitation behavior we show that the agents can more rapidly (i) learn the position of the resources (ii) create a relevant warehouse and (iii) satisfy their needs. Thus, the agents would be able to improve the performance of the whole system.

6.1 Optimization of Warehouses' Numbers and Convergence Time

We started to experimentally keep track of the number of warehouses' visits when the agents imitate as compared to the number of warehouses' visits when the agents do not have imitation capabilities (for 20000 time steps). The average number of visits to warehouses in the first case (246) is more important than in the second one (115). This shows that with the imitation behavior, the warehouses are more frequented by the agents. We defined the performance of the deposit system through the convergence time and the number of warehouses in the environment. The analysis of the agents' behavior in Table 1 enabled us to show that due to the imitation behavior the AA were able to deposit a fixed number of warehouses which can emerge in fixed positions while minimizing the warehouses numbers and the convergence time without having to use thresholds in order to limit the number of warehouses nor to specify their locations.

Table 1. Average of warehouses number and convergence time

	Warehouses Number	Convergence Time
Without Imitation Behavior	7	9000 time steps
With Imitation Behavior	6	8123 time steps

6.2 Optimization of Agents' Planning and Their Survival Rate

The AA were also able to create emergent and stable warehouses allowing an optimization of the planning. Table 2 shows that due to the imitation, agents can optimize their planning time with the help of relevant warehouses. This leads to a higher average of satisfaction level. Indeed, based on the imitation, the AA is able to better optimize the planning time of agents thanks to the faster learning of the new places of warehouses and to improve their level of satisfaction. To study the influence of the imitation strategy on the survival rate of the populations, we used the same environment and launched 20, 30, 40 then 60 agents, and counted the number of agents that survived, or died for not having found all of the three types of resource. The results show that adding an imitation capability can dramatically enhance the survival rate of the population from 45,66% (without imitation behavior) to 60,64% (with imitation). Indeed, coupling imitation behavior with the cognitive map allows agents to discover and to learn the position of the resources in the environment more rapidly.

Table 2. Optimization of planning time

	Without Imitation Behavior	With Imitation Behavior
Planning Time	450 time steps	218 time steps
Satisfaction Level	88.07	91.25

7 The Effect of Imitation Behavior on Swarm Robots

We have also validated the positive feedback of the imitation strategy on a minimal robotics setup [13] based on the same bio-inspired architecture which has been validated in real robots in [14]. We conducted 20 experiments (this number is determined by the statistical Fisher test) composed of an imitator robot IR (having an imitation behavior) and with a leader robot LR (that has already learned the position of the goals G1 and G2 with a threshold of vigilance to learn new places equal to 0.65 and a duration of learning equal to 30 minutes). Fig. 5a, b, d and e show that the IR tries to follow the LR which is looking for the goals. Fig. 5c and f demonstrate that the IR succeeded in discovering both goals during the following of the LR. As in the experiments with AA, this experience allows us to show through imitation that the IR was able to find both goals and to optimize the time of their discovery. Indeed, the IR takes 5 minutes to find both goals. However, a robot moving randomly (without an imitation behavior) takes 22 minutes (about 4 times more) to discover the two goals.

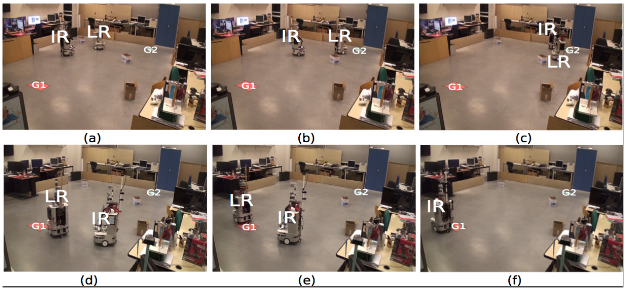


Fig. 5. The influence of the imitation behavior in swarm robots

8 Conclusion

In this paper we have tried to show the importance of the learning by imitation which leads to the improvement of the performance of the whole system. As a possible application, we have added in [15] customers to the environment in order to resolve the classical warehouse location problem. We also highlighted the importance of the imitation strategy in real robots which allows to solve the navigation task and optimizes the time to explore the plants. As prospects, we try to study the limits of emergent structures in the real world.

Acknowledgment. The authors would like to thank the financial support of the Tunisian General Direction of Scientific Research and Technological Renovation (DGRSRT), under the ARUB program 01/UR/11 02, the Institut Français de Tunisie and the NEUROBOT french project referenced ANR-BLAN-SIMI2-L2-100617-13-01.

References

1. Bonabeau, E., Theraulaz, G.: Intelligence Collective. Hermes (1994)
2. Brooks, R.A.: Coherent behavior from many adaptive processes. In: Proceedings of the Third International Conference on Simulation of Adaptive Behavior, pp. 22–29. MIT Press, Cambridge (1994)
3. Mataric, M.J.: Designing Emergent Behaviors: From Local Interactions to Collective Intelligence. In: Meyer, J.A., Roitblat, H., Wilson, S. (eds.) Proceedings of the Second Conference on Simulation of Adaptive Behavior, pp. 1–6. MIT Press (1992)
4. Chatty, A., Kallel, I., Gaussier, P., Alimi, A.: Emergent complex behaviors for swarm robotic systems by local rules. In: IEEE Symposium on Computational Intelligence on Robotic Intelligence In Informationally Structured Space (RiiSS), pp. 69–76 (April 2011)
5. Deneubourg, J.L., Goss, S., Franks, N., Franks, A.S., Detrain, C., Chrétien, L.: The dynamics of collective sorting robot-like ants and ant-like robots. In: Proceedings of the First International Conference on Simulation of Adaptive Behavior on From Animals to Animats, pp. 356–363. MIT Press, Cambridge (1990)
6. Gaussier, P., Zrehen, S.: Avoiding the world model trap: An acting robot does not need to be so smart! Robotics and Computer-Integrated Manufacturing 11(4), 279–286 (1994)
7. Gaussier, P., Moga, S., Banquet, J.P., Quoy, M., Modelisations, N.E.: From perception-action loops to imitation processes: A bottom-up approach of learning by imitation (1997)
8. Schaal, S., Peters, J., Nakanishi, J., Ijspeert, A.: Control, planning, learning, and imitation with dynamic movement primitives. In: Workshop on Bilateral Paradigms on Humans and Humanoids, IEEE International Conference on Intelligent Robots and Systems, IROS 2003 (2003)
9. Chaminade, T., Oztop, E., Cheng, G., Kawato, M.: From self-observation to imitation: visuomotor association on a robotic hand. Brain Research Bulletin 75(6), 775–784 (2008)
10. Muller, R.U., Stead, M., Pach, J.: The hippocampus as a cognitive graph (1996)
11. O'Keefe, J., Nadel, L.: The hippocampus as a cognitive map / John O'Keefe and Lynn Nadel. Clarendon Press, Oxford University Press, Oxford (1978)
12. Gaussier, P., Revel, A., Banquet, J.P., Babeau, V.: From view cells and place cells to cognitive map learning: processing stages of the hippocampal system. Biological Cybernetics 86(1), 15–28 (2002)
13. Chatty, A., Hasnain, S., Gaussier, P., Kallel, I., Laroque, P., Alimi, A.: Effect of low level imitation strategy on an autonomous multi-robot system using on-line learning for cognitive map building. In: IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1–6 (December 2012)
14. Chatty, A., Gaussier, P., Kallel, I., Laroque, P., Alimi, A.: Adaptation capability of cognitive map improves behaviors of social robots. In: IEEE Conference on Development and Learning and the Epigenetic Robotics, pp. 1–6 (November 2012)
15. Chatty, A., Gaussier, P., Kallel, I., Laroque, P., Florance, P., Alimi, A.: The evaluation of emergent structures in a cognitive multi-agent system based on on-line building and learning of a cognitive map. In: 5th International Conference on Agents and Artificial Intelligence (ICAART), pp. 269–275 (February 2013)

An Indexed K-D Tree for Neighborhood Generation in Swarm Robotics Simulation

Zhongyang Zheng and Ying Tan*

[1] Key Laboratory of Machine Perception and Intelligence(Peking University),
Ministry of Education
[2] Department of Machine Intelligence,
School of Electronics Engineering and Computer Science,
Peking University, Beijing, 100871, China
ytan@pku.edu.cn

Abstract. In this paper, an indexed K-D tree is proposed to solve the problem of neighborhoods generation in swarm robotic simulation. The problem of neighborhoods generation for both robots and obstacles can be converted as a set of range searches to locate the robots within the sensing areas. The indexed K-D tree provides an indexed structure for a quick search for the robots' neighbors in the tree generated by robots' positions, which is the most time consuming operation in the process of neighborhood generation. The structure takes full advantage of the fact that the matrix generated by robots neighborhoods is symmetric and avoids duplicated search operations to a large extent. Simulation results demonstrate that the indexed K-D tree is significantly quicker than normal K-D tree and other methods for neighborhood generation when the population is larger than 10.

Keywords: Neighborhood generation, k-d tree, simulation, range search, indexed k-d tree, swarm robotics.

1 Introduction

The researches of swarm robotic systems require plenty of physical robots, which makes it hard to afford for many research institutions [1]. Simulations on computers are developed to visually test the structures and algorithms on computer. Although the final aim is real robots, it is often very useful to perform simulations prior to investigations with real robots. Simulations are easier to setup, less expensive, normally faster and more convenient to use than physical swarms [2]. There exists several commonly-used simulation platforms, such as Player/Stage [3], Swarmanoid Simulator [4] and etc.

In the real-life swarm robotics applications, robots in the swarm can detect other robots within certain sensing ranges (they are inferred as neighbors in this paper) to exchange positions, current running states and other environment information. All these detections can be done with the help of on-board sensors. However, the problem becomes complicated in simulation, as we judge if

* Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 53–62, 2013.

two robots are neighbors through calculating their distance. Thus, research on generating neighbors for simulations is necessary.

1.1 Neighborhood Generation in Simulation

In the simulation of swarm robotics, sensing range of robots is fixed and the area it covers shapes as a square or circle (cube or sphere in 3-D situations). We denote all the robots inside the sensing area of a robot as the neighborhood of this robot, which needs to be calculated every iteration to determine all the neighbors of that robot.

We also need to detect if robots are near any obstacles in the environment, i.e. generate the obstacles' neighborhoods. Obstacles can vary in many aspects. They can be static or dynamic, appear or disappear during the simulation. Their sizes and shapes can be different, from small points to large polygons and they can have different sensing areas and only robots within the areas can detect them.

The problem of neighborhood generation can be defined as a set of range search problems. There exists a constant D and two collections, collection of dynamic points R (stands for robots) and collection of search ranges Q (sensing areas of obstacles). For any points P in R, we need to find all the points in R that are within the distance D to P. We also need to find all the ranges in Q that contain P. Positions of points in R and searching ranges in Q may be changing over time, and the results are calculated every iteration.

For a naive implementation, we calculate the distances of every two robots to see if they are within the sensing range. The computation complexity is $O(n^2)$, where n is population size. The time is quite short when n is small, but it becomes intolerantly large when n grows which is just the case in swarm robotics, as the population size should be at least tens or hundreds. So a smarter method should be introduced.

1.2 K-D Tree

K-d tree (shorted for k-dimensional tree), proposed by Bentley at 1975 [5], is a space-partitioning data structure for organizing points in a k-dimensional space. K-d tree is a binary tree of k-dimensional points. Every non-leaf node can be thought as a splitting hyper plane that divides the space into two half-spaces at a specified dimension. For example, if x axis is chosen for a node, points with a smaller x values than the node appear in the left sub tree and points with larger x values are in the right sub tree.

K-d tree is a useful data structure for several applications, such as searches involving a multidimensional search key (e.g. range searches and nearest neighbor searches) [6], calculating multi-scale entropy [7] or triangle culling for ray-triangle intersection tests in ray tracing [8]. So far, researches using k-d trees has concentrated on traversing them quickly, as well as on building them to be efficient for general applications [9], but usages for special applications are not focused.

2 The Indexed K-D Tree

Range searches in the neighborhood generation problem can be clustered into two types: search for ranges centered at robots or obstacles. Since the search ranges of obstacles may change or obstacles may be added into or removed from the environment, it's hard to build an optimized space indexing data structure for obstacle range searches. Thus, we focus on optimizing range searches for generating robots' neighborhoods while the detection of obstacles remains the same as the normal k-d tree in the proposed indexed k-d tree.

2.1 Motivation

We denote the results of all neighborhoods of the swarm as a matrix N, where $N(i, j)$ indicates robot j is a neighbor of robot i. Since the sensing ranges of each robot are the same, the matrix is symmetric. After searching a node for its neighbors, we can also determine whether it's the neighbor of other robots in the same time. Thus, this node can be removed from tree to shorten the searching time for the remaining nodes. However, in a normal k-d tree, removal of a node will take $O(logn)$ time, where n is size of the tree. In the next iteration, the very node will be added back to the tree when re-building the entire tree, which will take another $O(logn)$ time. We have to apply these remove and insert operations for $n - 1$ points each iteration. With such strategy, we spend more time for searching although we originally purpose is to save time.

We propose an indexed k-d tree to take full advantage of this strategy and guarantee the strategy really saves time in the searching process. Each node in the tree is assigned a zero-based index, ranges from 0 to $n-1$ and distinct to each other. It should be noted that a index is assigned to a node in the tree structure rather than the specific robot whose position is the value of the node. The tree is re-arranged every iteration since the robots are moving. The index of the tree node stays the same while it may refer to different robots in different iterations. The easiest way of implementing such method in the computer program is to store all the nodes in a array and index in the array indicates the index of the node.

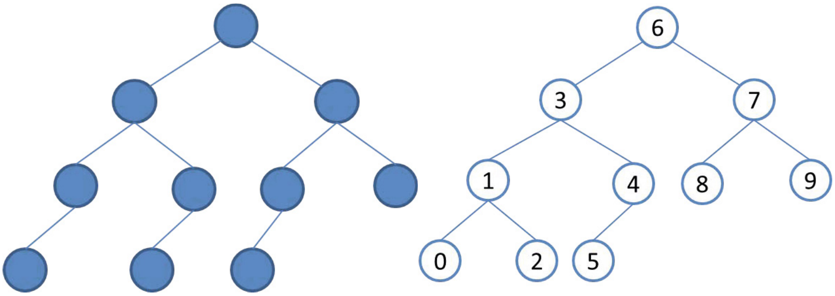
To simplify the removal operation, we do not really remove the node from the tree, but we ignore all the nodes that have been removed. The nodes should be removed (or ignored) carefully in a certain order that can benefit searching for the rest nodes. The ignored nodes should be distinguished easily by its index during the search, i.e. no longer than $O(1)$ time. All the children nodes of a tree branch must be searched before we can ignore the root node of entire tree branch safely. In our indexed k-d tree, indexes of nodes are assigned following a simple principle:

Tree Indexing Principle. $\forall i \in [0, n-1)$, nodes indexed $0, \ldots, i$ are connected and the sub tree formed by these nodes is a valid k-d tree with the height of not more than $log(i) + 1$.

With this simple principle, we can search and remove the indexed nodes in the descending order. We can ignore all the nodes with larger indexes than that of the center of current search range, when we apply range searches on the rest of the nodes, which still form a valid k-d tree according to the principle. The restriction of the height of the sub tree is to optimize the indexing so that the sub tree is more balanced and saves searching time.

2.2 The Indexed Structure

The structure of the proposed indexed k-d tree is shown in Figure 1. We can easily see from the figure that, unlike normal k-d tree, the tree is unbalanced as nodes in the last layer are left-aligned. Since the searching time is more related with the height of the tree rather than size, we should always try to keep the remaining sub trees lower. As we remove the nodes with larger indexes first when searching, the height of sub trees of the unbalanced tree will descend more quickly than a normal tree.



(a) Normal k-d tree (b) Indexed k-d tree

Fig. 1. Comparison between normal and indexed k-d tree

The tree structure is initialized before the simulation starts according to the population size of the swarm. The nodes will fill all the layers of the tree from top to bottom and fill a layer from left to right. After that, tree nodes are indexed according to the following three rules.

Tree Indexing Rule 1. The indexes of a node and all its sub-nodes are continuous.

When building and updating the tree, we need to split the positions of the robots at certain dimension in k-d tree. We store all these values in an array. Continuous indexes can benefit the split process for easier program readability and faster execution time since reading continuous memory blocks is quicker than sparse ones.

Tree Indexing Rule 2. The indexes of nodes in the right sub tree of node i are always larger than i and nodes in the left sub tree.

A larger index indicating the node will be removed earlier in the search process. Since the tree is unbalanced and the right side is usually smaller than the left side, nodes in the right sub tree should have larger indexes.

Tree Indexing Rule 3. The indexes of nodes in the left sub tree of node i is smaller than i if node i is the first node of that layer, and vice versa.

As for indexing the left sub tree, there exist two situations. First nodes of the layer (indexed 6, 3, 1 or 0 in Figure 1b) should have larger indexes than nodes in their left sub trees, since they are currently the root of the current search tree with an empty right sub tree. In the next step of search process, the root node is “removed” and the height of the tree will reduce by one. For other nodes, their left sub trees should be “removed” earlier than themselves, so they have smaller indexes than nodes in left sub trees.

3 Neighborhood Generation Using the Indexed K-D Tree

When using the indexed k-d tree in simulations, we first initialize the tree structure before simulation starts using the rules proposed in the previous section. Every iteration in the simulation, we first update the values in the tree nodes for splitting the space and then apply range searches for neighborhood generation.

The update operation works similar with the construction of the tree. In a normal construction of k-d tree, a split operation is applied for each node. Nodes in the left and right sub trees of one node are split by the node itself at specified dimension. The complete operation has a computational complexity of $O(nlogn)$ where n is size of the tree. However, the robots have a limited maximum moving speed, which is normally small compared to the sensing range. So we assume that the places of the nodes in the tree remain almost the same in two consequent iterations. Therefore, we always try to split the updated sub trees using the same old node that split the values in previous iteration. The complete construction will take $O(n)$ time in the best condition. The average construction time is $O(nlogn)$ and is guaranteed to be not more than that of a normal tree.

After updating the tree, we generate the neighborhoods for all robots. With each tree node indexed, searching becomes simple. We traverse the node index i from $n-1$ downwards to 0. For each index i, we search for neighbors of node i in the tree formed only by the nodes indexed from 0 to $i-1$ which is a valid indexed k-d tree according to the principle. Since the result matrix N is symmetric, we assign all $N(j,i) = N(i,j)$ where $j < i$. We also assign $N(i,i)$ to be true or false according to the requirements of the application. After assigning all the values related with robot i in matrix N, we step to next index $i-1$ until i reaches 0. From Figure 1b, we can see that the root of the tree changes during the search and the tree gradually shrinks to the left corner.

If obstacles are involved in the simulation, we search the neighbors of these obstacles in the complete tree just like normal range searching in a normal k-d tree.

The pseudo code of neighborhood generation using the indexed k-d tree is shown in Algorithm 1.

Algorithm 1. Code for Neighborhood Generation at Every Iteration using Indexed K-D Tree

```
Update tree, trying to use the old split nodes
for $i = n - 1$ to 0 do                //Robots' neighborhoods
    Set up search range $R$ centered at node $i$
    Search the tree formed by nodes [0,i-1) using range $R$
end for
for all obstacle $o$ do                //Obstacles' neighborhoods
    Set up search range $R$ centered at obstacle $o$
    Search the entire tree using range $R$
end for
Return neighborhoods
```

4 Results and Discussions

In this section, experimental results and discussions are presented. We first compare the efficiency of the improved k-d tree with other implementations in different population sizes, and then compare the performance under obstacle environments.

To test the efficiency of neighborhood generation for different methods, we run tests on our self-built simulation platform. The robots wander in the environment with randomly generated directions. They bounce at the borders and have a rate of 5% to change their direction with randomly generated ones. All our experiments are repeated 10 times, each has 10,000 iterations. The time in our results are the total time used for generating neighborhoods for all these iterations. The time that spends on initializing and calculating robots' movements is not included.

4.1 Time Comparison among Different Population Sizes

We compare the efficiency of our indexed k-d tree with normal k-d tree and the naive implementation under different population sizes in this section. The results are shown in Table 1. In the table, naive stands for the simple implementation that calculates all the distances of every two robots, normal stands for the normal k-d tree and indexed stands for our proposed k-d tree in this paper. The normal k-d tree use the same updating strategy as our indexed k-d tree and searches the entire tree for every robot, which is faster than removing a node and adding it back using the normal way. The running times in milliseconds stands for the time for all 100,000 iterations (10,000 iterations each for 10 times).

From the table, we can see that as population grows, time for indexed k-d tree grows slower than naive, as the computational complexity tells. Our indexed k-d tree speeds less time than naive when population becomes more than 13. As population increases, we can see that indexed k-d tree becomes 20 to 30 percent quicker than naive when population is fewer than 30. When population grows to 100 and 1000, this percentage becomes even larger to 60 and even 75. We

Table 1. Time Comparison among Different Populations (Lower is better)

Population	Running Time (ms)			Time Ratio	
	Naive	K-D Tree		Indexed /	Indexed /
		Normal	Indexed	Naive	Normal
2	**17.91**	52.38	41.56	232.05%	79.34%
3	**34.32**	98.05	68.28	198.97%	69.64%
4	**57.48**	166.79	115.92	201.66%	69.50%
5	**87.08**	239.69	153.04	175.74%	63.85%
6	**125.34**	360.77	264.63	211.14%	73.35%
7	**169.88**	384.96	270.77	159.39%	70.34%
8	**217.81**	480.22	328.61	150.87%	68.43%
9	**283.61**	633.33	375.43	132.38%	59.28%
10	**337.62**	632.06	425.74	126.10%	67.36%
11	**416.04**	838.45	502.89	120.88%	59.98%
12	**490.28**	905.74	610.45	124.51%	67.40%
13	**575.31**	1040.42	624.12	108.48%	59.99%
14	668.96	1046.03	**645.76**	96.53%	61.73%
15	768.43	1254.40	**715.47**	93.11%	57.04%
16	859.12	1456.15	**781.45**	90.96%	53.67%
17	969.94	1666.82	**928.95**	95.77%	55.73%
18	1091.89	1692.73	**977.72**	89.54%	57.76%
19	1221.11	1880.05	**985.22**	80.68%	52.40%
20	1360.52	2001.45	**1145.62**	84.20%	57.24%
21	1474.70	1971.13	**1246.12**	84.50%	63.22%
22	1632.14	2172.43	**1343.55**	82.32%	61.85%
23	1789.37	2197.13	**1347.18**	75.29%	61.32%
24	1924.72	2444.18	**1532.71**	79.63%	62.71%
25	2132.77	2860.92	**1605.33**	75.27%	56.11%
26	2295.74	2755.71	**1614.63**	70.33%	58.59%
27	2467.44	3045.49	**1925.85**	78.05%	63.24%
28	2632.62	3031.58	**1962.45**	74.54%	64.73%
29	2870.51	3334.43	**2036.15**	70.93%	61.06%
30	3036.72	3472.01	**2104.65**	69.31%	60.62%
100	32,774	19,775	**12,946**	39.50%	65.47%
1000	3,520,591	1,148,533	**862,571**	24.50%	75.10%

can say, according to the results, the indexed k-d tree over-performs the naive implementation even when population is small, not to mention the advances when population size blows.

We can also see that our indexed k-d tree is 40% quicker than normal k-d tree when population is larger than 9. We can see the advantage of indexed k-d tree drops a little when population is very large. The reason for such situation is that updating time of the tree increases quickly than the searching time. As for normal k-d tree, we can see that even when population is 30, it's still slower than the naive implementation. Actually, in our experiment, it won't be faster than naive until population reaches 35.

In summary, our indexed k-d tree is 40% quicker than normal k-d tree and at least 20-30% quicker than the naive implementation when population is more than 10. The advantage becomes more notable when population grows. We can defer that our indexed k-d tree is applicable in real-time simulations, especially for swarm robotics applications which normally have a population size of at least 10.

4.2 Time Comparison under Obstacle Situation

In obstructive environments, calculation of the neighborhoods of obstacles is also involved as we mentioned in previous sections. In a swarm robotic simulation, if obstacles are introduced, the number of obstacles can vary from a few to hundreds. In this section, we compare the performance of algorithms under environments with 10-50 obstacles since the trends are displayed completely

Table 2. Running Time Ratio in Obstacle Range Searches (Lower is better)

		Number of Obstacles				
		10	20	30	40	50
Population	2	226.41%	217.73%	188.16%	213.31%	219.46%
	3	171.87%	152.51%	162.49%	164.14%	164.37%
	4	167.75%	142.51%	145.66%	143.43%	136.36%
	5	138.93%	123.78%	123.29%	121.46%	122.81%
	6	125.93%	117.00%	111.10%	108.74%	105.85%
	7	106.89%	107.05%	103.89%	97.81%	97.81%
	8	107.50%	100.14%	92.45%	93.75%	97.75%
	9	97.78%	92.98%	95.99%	90.68%	86.43%
	10	97.42%	86.80%	88.04%	86.08%	79.39%
	11	94.91%	85.88%	81.64%	78.10%	76.23%
	12	86.09%	81.33%	77.98%	78.10%	75.26%
	13	81.66%	77.44%	73.96%	71.87%	72.49%
	14	83.94%	71.90%	71.43%	69.54%	69.55%
	15	80.64%	71.42%	69.46%	67.02%	68.03%
	16	78.30%	70.90%	65.33%	65.03%	64.35%
	17	85.30%	70.23%	65.09%	64.17%	63.53%
	18	75.45%	66.59%	67.01%	63.02%	61.64%
	19	68.46%	68.44%	63.01%	61.26%	57.20%
	20	68.17%	63.48%	63.88%	59.92%	58.05%
	21	66.97%	65.14%	58.36%	58.33%	58.07%
	22	67.02%	61.49%	57.44%	55.48%	54.69%
	23	67.16%	63.74%	59.85%	56.51%	54.20%
	24	64.98%	64.17%	57.00%	56.66%	53.84%
	25	63.88%	61.49%	56.44%	55.14%	52.76%
	26	63.69%	57.10%	55.23%	54.14%	50.73%
	27	64.77%	58.33%	53.90%	51.37%	50.49%
	28	60.12%	56.90%	53.81%	52.45%	48.89%
	29	62.14%	56.54%	50.08%	50.14%	52.67%
	30	65.10%	58.79%	52.99%	48.59%	46.05%

with these results. From the previous section, we can see our indexed k-d tree is always quicker than normal k-d tree. Since these two k-d trees use the same strategy for searching obstacles, we only compare the result between indexed k-d tree and naive implementation. The time ratios (indexed / naive) under different populations and numbers of obstacles are shown in Table 2.

From the table, we can see that the two algorithms have the same trend as the situation without obstacles when population increases. As number of obstacles increases, time ratio is decreasing rapidly and our indexed k-d tree has a larger time advantage. The populations, at which indexed k-d tree becomes quicker than naive implementation, decrease to 7-9 due to different number of obstacles, compared with 13 when no obstacles are involved. As population increases, the time ratio decreases to 50-60 which means indexed k-d tree spends almost only half the time than the naive implementation. The results demonstrate that our indexed k-d tree has a greater advantage than in no obstacle environments and it's more applicable in simulations since numbers of obstacles can be tens or hundreds in most swarm robotic simulations.

5 Conclusion

In this paper, we consider the problem of neighborhood generation in swarm robotic simulations. The problem can be converted into a series of range search problems. Based on the characteristics of this problem, we proposed an indexed k-d tree, which provides a structure for quickly searching a set of ranges centered at the points which formed the k-d tree. The simulation results show that our indexed k-d tree is almost 40% quicker for calculating robots' neighborhoods than normal k-d tree. It is also significantly quicker than the naive implementation in most population sizes and numbers of obstacles for neighborhood generation.

Due to the tree structure and node indexes, the indexed k-d tree is 30-40% faster than traditional methods for calculating neighborhoods of all the robots in the swarm, which is the most time consuming operation of neighborhood generation problem in swarm robotics simulations. This result demonstrates that the indexed k-d tree can accelerate the process of generating neighborhoods significantly and is very applicable for swarm robotics simulations.

Acknowledgements. This work is supported by the National Natural Science Foundation of China under grants number 61170057 and 60875080. Professor Tan Ying is the corresponding author.

References

1. Shi, Z., Tu, J., Zhang, Q., Liu, L., Wei, J.: A survey of swarm robotics system. In: Tan, Y., Shi, Y., Ji, Z. (eds.) ICSI 2012, Part I. LNCS, vol. 7331, pp. 564–572. Springer, Heidelberg (2012)
2. Michel, O.: Webotstm: Professional mobile robot simulation. arXiv preprint cs/0412052 (2004)

3. Vaughan, R.: Massively multi-robot simulation in stage. Swarm Intelligence 2(2), 189–208 (2008)
4. Pinciroli, C.: The swarmanoid simulator. Technical report. Citeseer (2007)
5. Bentley, J.: Multidimensional binary search trees used for associative searching. Communications of the ACM 18(9), 509–517 (1975)
6. Arroyuelo, D., Claude, F., Dorrigiv, R., Durocher, S., He, M., López-Ortiz, A., Ian Munro, J., Nicholson, P., Salinger, A., Skala, M.: Untangled monotonic chains and adaptive range search. Theoretical Computer Science 412(32), 4200–4211 (2011)
7. Pan, Y., Lin, W., Wang, Y., Lee, K.: Computing multiscale entropy with orthogonal range search. Journal of Marine Science and Technology 19(1), 107–113 (2011)
8. Zhou, K., Hou, Q., Wang, R., Guo, B.: Real-time kd-tree construction on graphics hardware. ACM Transactions on Graphics (TOG) 27, 126 (2008)
9. Wald, I., Havran, V.: On building fast kd-trees for ray tracing, and on doing that in o (n log n). In: IEEE Symposium on Interactive Ray Tracing 2006, pp. 61–69. IEEE (2006)

Agent-Based Social Simulation and PSO

Andreas Janecek[1], Tobias Jordan[2], and Fernando Buarque de Lima-Neto[3]

[1] University of Vienna, Research Group Entertainment Computing, Austria
andreas.janecek@univie.ac.at

[2] Department of Economics and Business Engineering,
Karlsruhe Institute of Technology (KIT), Germany
tobias.jordan@student.kit.edu

[3] Polytechnic School of Engineering, Computing Engineering Program
University of Pernambuco, Recife/PE, Brazil
fbln@ecomp.poli.br

Abstract. Consumer's behavior can be modeled using a utility function that allows for measuring the *success* of an individual's decision, which consists of a tuple of goods an individual would like to buy and the hours of work necessary to pay for this purchase and consumption. The success of such a decision is measured by a utility function which incorporates not only the purchase and consumption of goods, but also leisure, which additionally increases the utility of an individual. In this paper, we present a new agent based social simulation in which the decision finding process of consumers is performed by Particle Swarm Optimization (PSO), a well-known swarm intelligence method.

PSO appears to be suitable for the underlying problem as it is based on previous *and* current information, but also contains a stochastic part which allows for modeling the uncertainty usually involved in the human decision making process. We investigate the adequacy of different bounding strategies that map particles violating the underlying budget constraints to a feasible region. Experiments indicate that one of these bounding strategies is able to achieve very fast and stable convergence for the given optimization problem. However, an even more interesting question refers to adequacy of these bounding strategies for the underlying social simulation task.

Keywords: Agent-Based Modeling, Social Simulation, Particle Swarm Optimization, PSO, Consumer's Behavior, Decision Finding Process.

1 Introduction

Economic developments depend upon various external and internal factors, limiting the possibility of human beings to predict or even understand effects that certain political decisions or social-economic changes may have on these developments. Economists use therefore to simplify the reality, to make assumptions, and to press the reality into a mathematical framework in order to be able to simulate the reality. Agent based economic models are becoming increasingly important for economists in order to simulate human behavior or human decisions

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 63–73, 2013.

in a bounded rational way [1]. In this paper, we present a new social simulation based on Swarm Intelligence (SI), a branch of nature inspired optimization based on the collective behavior of inherently decentralized and self-organized autonomous entities [2]. Particle Swarm Optimization (PSO) [2, 3], one of the most prominent current SI techniques, is here used to simulate the decisions of individuals within an agent based economic simulation model.

The underlying optimization problem of this work stems from consumer's decision theory. We assume that households are facing a multi-variant optimization problem with the goal to maximize their utility by choosing the respective best combination (tuple) of differently priced and differently preferred goods (summarized as *consumption*), and the required number of working hours needed in order to afford these tuple of goods. Each additionally bought good *and* each additional hour of leisure (i.e., time spent not working) increases the utility.

Although meta-heuristic search techniques such as PSO do not guaranty to find an optimal solution for the given optimization problem, we believe that some properties of PSO are well suited for this task. PSO is in some sense able to resemble human decision making since it is based on previous and current information, and, even more importantly, also contains a probabilistic part which allows for modeling the uncertainty usually involved in the human decision making process. Similar to human behavior, the PSO swarm allows for parallel reflection and selection of different alternatives with a mixture of probabilistic and past oriented measures. As a result, PSO allows for approximating the simulation process desirably close to human choices. In contrast to meta-heuristics, conventional methods which are able to solve the given problem exactly (e.g., Lagrangian method) are not that well suited for simulating such a behavior.

Related Work. Following the categorization of John Holland [4], *Agent Based Modeling (ABM)* refers to the computational study of social agents as evolving systems of autonomous interacting agents. ABM is a tool for studying social systems from a complex adaptive system perspective. From this perspective, a researcher is interested in how macro phenomena are emerging from micro level behavior among a heterogeneous set of interacting agents [4]. ABM has been applied in various fields, e.g., for observing racial segregation [5], political opinion building [6], consumer behavior [7], and various other fields. The research area of *Agent Based Social Simulation (ABSS)* can be found in the intersection of the three disciplines Agent Based Computing, Social Sciences and Computational Simulation [8]. Applications of ABSS in social sciences are of Serrano Filho *et al.* in demography [9] and Alvaro and Lima-Neto [10] in econometry. One advantage of social sciences conducted with ABM is that it allows for debugging and understanding macro phenomena better, hence, allowing for simulating on an experimental base without being faced to ethical or numerical problems. A detailed summary of sociology in ABSS can be found in [11]. Emerging from this ABM approach, a particular field of research has been established: *Agent Based Macroeconomics*, i.e., studying macroeconomic contexts with ABM. This type of macroeconomic simulation avoids problems with other simulation methods and gives new possibilities of research [12]. For a summary of this field of science see

Testfatsion [13]. The work of Delli Gatti [14] is a very elaborated example for the construction of a macroeconomic model based on autonomous agents.

The applicability of SI for optimizing business processes in economics has been discovered over a decade ago, e.g., [15]. Since then, PSO has been used to improve various kinds of business models. Two recent studies focus on improving cluster analysis within a decision making model [16], and on optimizing product-mix models [17]. Another recent study discusses the applicability of computational intelligence and ABM for financial forecasting [18]. In terms of social simulation, PSO has been applied for simulating human behavior in emergency situations [19], and for insurgency warfare simulation [20]. Although several heuristic methods for optimizing and simulating the consumer's decision making process have been proposed, e.g., [21], the application of SI methods to this task has not been investigated yet. To the best of our knowledge, this is the first study that examines the adequacy of PSO or any other SI technique for simulating the human decision making process.

Synopsis. We briefly review PSO in Section 2 and give a formal description of the underlying decision problem in Section 3. Modifications of PSO needed in order to customize the PSO to the decision problem are summarized in Section 4. Experiments are evaluated and discussed in Section 5, and Section 6 concludes the paper and discusses ideas for future research in this context.

2 Particle Swarm Optimization (PSO)

PSO [2, 3] is a stochastic global optimization technique inspired by the social behavior of swarms, where every individual or *particle* traces a trajectory in the allowed search space (cf. Algorithm 1). At first, the positions (locations) $\boldsymbol{l}$ and velocities $\boldsymbol{v}$ of the particles are initialized randomly in the allowed range, together with three parameters: the inertial weight w, and two acceleration coefficients, $c1$ and $c2$. Each particle i stores its current location $\boldsymbol{l}_i$ and velocity $\boldsymbol{v}_i$, the best location it has visited so far $\boldsymbol{l}_i^b$ ("personal best"), and the best location $\boldsymbol{g}^b$ visited so far by the whole swarm ("global best"). In each iteration, the particles move through the search space based on their current weighted velocity $\boldsymbol{v}_i$ incremented or decremented by a weighted sum consisting of the differences of $\boldsymbol{l}_i^b$ and $\boldsymbol{l}_i$, and $\boldsymbol{g}_b$ and $\boldsymbol{l}_i$, respectively. It may happen that the new location of a particle is outside of the allowed search space. If so, the location needs to be mapped back to the allowed range (see Section 3). At the end of each iteration, the new locations are evaluated and $\boldsymbol{l}_i^b$, and $\boldsymbol{g}_b$ updated.

3 Modeling

In accordance with microeconomic theory, we focus on rationally behaving households that follow the economic principle. Households demand consumption of goods and offer labor in order to afford these goods. We assume that each household tries to maximize its utility by either increasing consumption or leisure (i.e., time spent not working). Consumption and leisure are two interacting

Algorithm 1 – General structure of PSO algorithms.

```
Initialize $\boldsymbol{l}_i$ and $\boldsymbol{v}_i$ of initial population, as well as parameters $w$, $c1$, $c2$
repeat
    Update velocity: $\boldsymbol{v}_i = w \cdot \boldsymbol{v}_i + c1 \cdot rand \cdot (\boldsymbol{l}_i^b - \boldsymbol{l}_i) + c2 \cdot rand \cdot (\boldsymbol{g}^b - \boldsymbol{l}_i)$
    Update position: $\boldsymbol{l}_i = \boldsymbol{l}_i + \boldsymbol{v}_i$
    Map position to allowed search space if out of bounds
    Evaluation of fitness of new position $f(\boldsymbol{l}_i)$
    Update $(\boldsymbol{l}_i^b)$ if $f(\boldsymbol{l}_i) < f(\boldsymbol{l}_i^b)$, and $(\boldsymbol{g}_b)$ if $f(\boldsymbol{l}_i) < f(\boldsymbol{g}^b)$
until termination (time, max. number of iterations, convergence, ...)
```

goods – under fixed environmental conditions increasing consumption usually decreases leisure, and vice versa. We define some declaration and notation in the following:

- *Goods.* We are considering a fixed number of n distinct goods (for simplicity, we do not distinguish between basic and luxury goods). Each good does not resemble a single item but rather a collection or group of similar items, e.g., food, transport, housing, communication, education, and so on. The amount of each of the n goods consumed by an individual is abbreviated as x_i, and all goods are stored in an n-dimensional item vector $\boldsymbol{x} = (x_1, x_2, ..., x_n)$.

- *Prices.* In order to purchase each of the n goods, an individual has to pay a different price for each good (i.e., for each group of items). The prices p_i for each of the n goods are stored in an n-dimensional price vector $\boldsymbol{p} = (p_1, p_2, ..., p_n)$.

- *Work.* Work is measured as the number of working hours per decision unit (day, month, year, ...) and abbreviated as w. We assume that work is limited by the maximum number of working hours per decision unit w_{max}, which is defined by external authorities (e.g. by a legal framework, medical reasons, ...). We point out that it is also possible to set w_{max} to the number of hours during some relevant period of time. However, in this case one would assume that sleeping and all other necessary daily duties count as leisure.

- *Leisure time.* In most cases, working is a "bad" for individuals, however, it is possible to measure *negative* work as a "good". *Leisure time* refers to the time an individual has available to her/him during some relevant period, i.e., the time this individual does not have to work $w_{max} - w$.

- *Preferences.* Each individual has different preferences for each of the n goods, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, ..., \lambda_n)$. Additionally, each individual has some preference for leisure time, which is denoted as λ_{lt}. We note that the sum of all preferences (for leisure time and all goods) is equal to 1, i.e., $\lambda_{lt} + \sum_{i=1}^{n} \lambda_i = 1$.

Utility Function. As underlying value allocation function for this work serves the one designed by Cobb and Douglas [22]. This function provides the model with "well-behaved" preferences, which are regarded as standard preferences for the valuation of alternatives in micro economic theory [22]. In the most-simple form an individual's utility function $u(c, l)$ has only two arguments, consumption

c and leisure l, and has at least the following properties: Utility is always strictly increasing in consumption (i.e., $\partial u/\partial c > 0$), and leisure (i.e., $\partial u/\partial l > 0$). In the above utility function, consumption is the product of all goods to the power of the individual's preferences, i.e., $c = \prod_{i=1}^{n} {x_i}^{\lambda_i}$. The second argument, leisure, is computed as leisure time to the power of the preference for leisure time, i.e., $l = (w_{max} - w)^{\lambda_{lt}}$. Using a basic Cobb-Douglas utility function in the form $u(x_1, x_2) = x_1^c \times x_2^d$, utility can be computed as

$$u(c,l) = c \times l \equiv \left(\prod_{i=1}^{n} {x_i}^{\lambda_i} \right) \times (w_{max} - w)^{\lambda_{lt}} \,. \tag{1}$$

Constraints. The Cobb-Douglas function imposes some implicit minimum consumption constraints for each good, since the utility will decrease if either one or several of the goods, and/or leisure are smaller than 1. However, this property of the utility function should not harm our algorithm, since we consider any x_i as an agglomeration of different types of goods. A zero consumption would mean to unrealistically disclaim the consumption of one x_i in total. In addition to this implicit minimum consumption constraint, the above utility function is subject to two explicit constraints which cannot be violated:

- *Constraint 1: Maximum work time.* Work w is limited by the maximum amount of work hours per relevant period w_{max}, such that $w \leq w_{max}$.

- *Constraint 2: Limited budget.* The expenses e cannot be higher than the total salary s, such that $e \leq s$, where e is calculated as the sum of the products of the amount of each good times the price for this good, $e = \sum_{i=1}^{n} x_i \, p_i$, and s is calculated as work w times salary per hour s_h[1], $s = w \times s_h$.

4 Agent-Based PSO

In this study, we assume that PSO works as meta-heuristic decision system *within* the mind of a single agent or individual. Mathematically, the position vector $\boldsymbol{l}$ of PSO concatenates the item-vector $\boldsymbol{x}$ and work w, the values to be optimized by PSO. In each iteration, the swarm computes new possible decisions for the agent, and each particle resembles one considered solution. We simplify the comparison between a human choice and PSO by dividing the selection process in two steps, (i) finding, and (ii) evaluating candidate solutions (i.e., tuple, decision,...).

(i) The human decision finding process is influenced by internal and external factors: Internal factors are, e.g., curiosity and experience of an individual. The personal and global best position of PSO can be used as means to simulate experience of an individual. Formerly successful solutions are hereby considered and new solutions might be in close proximity to remembered ones. Curiosity, as well as the huge number of external factors can be simulated by the stochastic part of PSO, since each new location is partly based on some random movement.

[1] We are not investigating the influence of any kind of taxes (including income tax) on consumption or leisure. We assume that s_h is the disposable salary (income) per hour of an individual, i.e. her/his after tax income.

(*ii*) When an individual has found a new solution or tuple, the utility of this solution is evaluated. Since individuals aim at maximizing their utility, the solution that provides the maximum utility is chosen. In case of the Cobb-Douglas utility function (Equ. 1), the solutions are evaluated following the weighted additive rule that has been examined to be a human choice heuristic [21]. For PSO, this implies that the fitness of a particle's position $\boldsymbol{l}$ is evaluated by computing the utility of this position (Equ. 1), and the best position found so far is regarded as the current solution (i.e., decision) of the PSO-agent.

Similarity between PSO Parameters and a Human Decision. Table 1 gives an overview of the connection between some selected PSO features and their corresponding interpretation in the process of human decision making.

Table 1. Comparison between PSO and human perspective in decision finding

PSO	Human (homo economicus)
Particle within the allowed range	A possible (i.e., valid and affordable) human decision that might be taken into account
Update	An attempt to find a new decision with better utility (that is valid and affordable)
Velocity	Individuals' curiosity, i.e., the willingness to change his/her current decision. However, it is also connected with experience, since an "unexperienced" consumer is more likely to change its current decision than an "experienced" consumer
Fitness evaluation	Reflection of a specific decision
g^b and $\boldsymbol{l}_i^b$	A remembered solution with rather high utility. Part of the human memory with links to similar decisions (*experience*).

Bounding Strategies. If a particle is outside of the allowed search space it is mapped back to the allowed range. There are three reasons why a particle can be out of bounds. (*i*) A valid decision does not allow for consuming negative amounts of any good nor a negative amount of leisure time. Hence, all negative values of $\boldsymbol{x}$ and leisure time are set to 0. (*ii*) Constraint 1 in Section 3 cannot be violated. If work w increases the maximum amount of working hours w_{max}, w is reset to w_{max}. (*iii*) Constraint 2 in Section 3 cannot be violated. If the expenses e exceed the salary s, the following bounding strategies are applied:

Bound to Border. The particles are mapped to the indifference map corresponding to the current salary. This is done by computing the ratio r between salary and expenses $r = s/e$, and multiplying all elements of $\boldsymbol{x}$ element-wise with this ratio, $\boldsymbol{x} = \boldsymbol{x} \cdot r$. Since $r < 1$, the elements of $\boldsymbol{x}$ are diminished, and, as a result, the expenses are diminished until e equals s.

Let-Them-Fly. Actually, this strategy does not map a particle to a new position, but "ignores" its current position for updating. A solution (i.e., a location of a particle) outside the allowed search space is marked as invalid and is not used for updating $\boldsymbol{l}_i^b$ nor for updating $\boldsymbol{g}_b$, respectively (cf. Line 7 in Algorithm 1).

Initialization Strategies. Two different initialization strategies are used to closer simulate the starting situation of a single agent. Agents that are considered as “lazy” might start their search at a low work position, while hard working agents initialize their search from positions with high work.

Initialization at Low Work. An extreme initialization strategy, where all initial particles reflect consumer decisions with only a small amount of working hours and therefore only low consumption – since expenses cannot extend the salary.

Initialization at High Work. This refers to the opposite extreme initialization strategy, where all particles reflect consumer decisions with a high amount of working hours and possibly – *but not necessarily* – higher consumption.

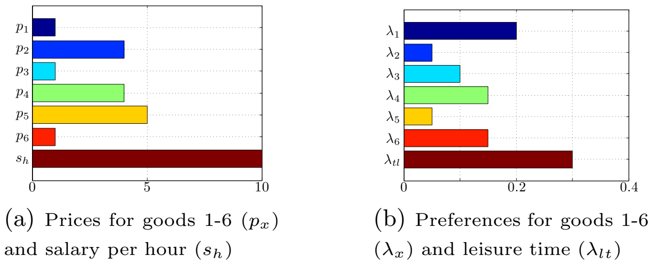
5 Experimental Evaluation

All experiments were performed with a swarm of 20 particles connected via a *gBest* topology, running for 100 iterations. The particles resemble a reasonable number of 20 alternative tuples, a human might reflect in parallel. We fixed the number of goods to six, as well as their prices and preferences (cf. Figure 1). The granularity of goods can be varied easily, however, a classification of goods is necessary due to the tremendous amount of possible goods. The salary per hour s_h and the preference for leisure time λ_{lt} are also shown in this figure. The maximum number of work hours w_{max} was set to a rather conservative value of 11 hours per day[2]. We evaluate the applicability of the two bounding strategies and the two initialization strategies presented in Section 3. A quantitative evaluation of the results is followed by a qualitative analysis in the discussion part.

Quantitative Evaluation. The results are presented in Figure 2, for the *bound to border* bounding strategy, and in Figure 3, for the *let them fly* bounding strategy. Both figures show the following information as average result over 30 independent runs: (*i*) The first graph shows the quantity or amount of each of the six goods (cf. vector $\boldsymbol{x}$ in Section 3) per iteration, as well as the number of working hours. Note the two different scales on the y-axis which are separated by the solid black line. (*ii*) The second graph shows the expenses in monetary units for each of the six goods (i.e., $\boldsymbol{x} \times \boldsymbol{p}'$). The limits of the y-axis are set to the maximum possible salary, i.e., $w_{max} \times s_h = 110$. Recall that a decision at maximum possible salary corresponds to a leisure time of 0. (*iii*) The third graph shows the utility per iteration for all single runs, as well as the average utility and the standard deviation over all 30 runs.

Bound to Border. Independent of the initial starting points of the agents, this bounding strategy is able to converge to the optimum solution within less than 100 iterations (cf. Figure 2). The optimal tuple $\boldsymbol{x}$ of goods 1-6 that is able to achieve the optimal utility of around 6.48 is (22.000, 1.375, 11.000, 4.125, 1.100, 16.500), with a corresponding amount of work of $w_h = 7.700$. This tuple is

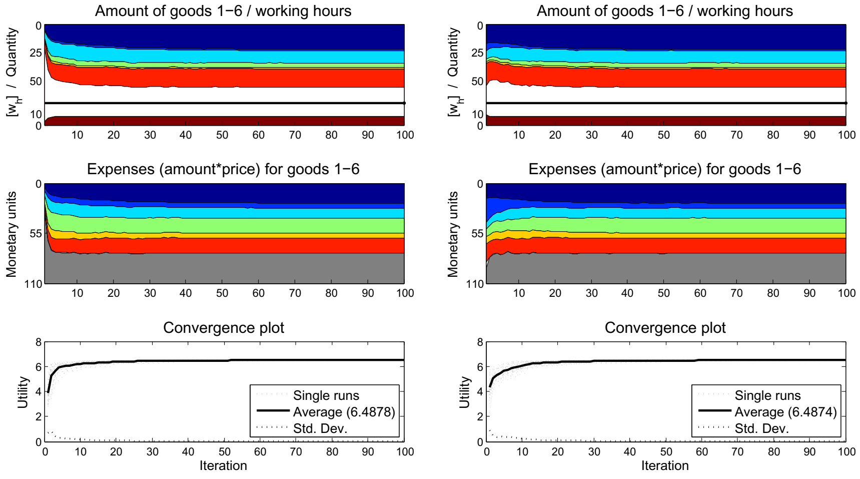
[2] Notice that preferences, prices and w_{max} are fictive values, since a realistic setting for those parameters is not required in order to evaluate the performance and general applicability of PSO for the given optimization task.



(a) Prices for goods 1-6 (p_x) and salary per hour (s_h)

(b) Preferences for goods 1-6 (λ_x) and leisure time (λ_{lt})

Fig. 1. Prices and Preferences



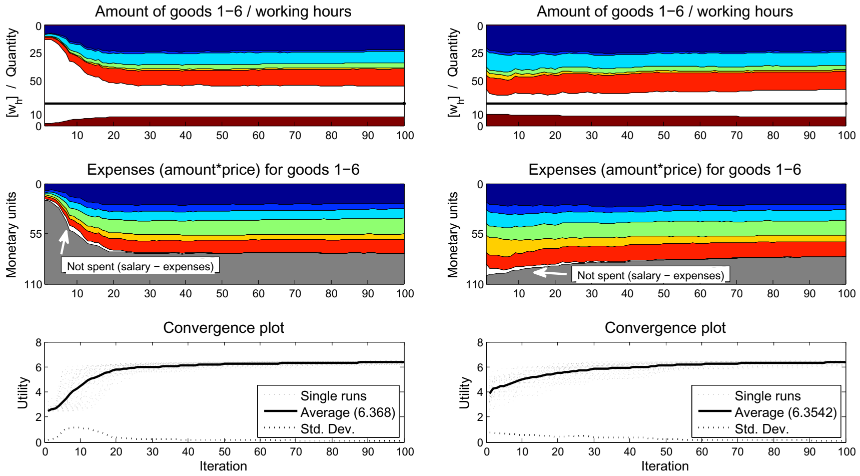
(a) Initialization at low work

(b) Initialization at high work

Fig. 2. Bounding strategy: bound to border

plotted in the first graph (iterations≥∼70) of Figures 2(a) and 2(b), respectively. Comparing this tuple with Figure 1, it can be seen that for the optimal solution the cheapest and most preferred goods (x_1, x_6) have a much higher contribution than expensive and less preferred goods (x_2, x_5).

Let Them Fly. This second bounding strategy is significantly slower in terms of convergence (cf. Figure 3). We found that this bounding strategy is able to achieve almost optimal solutions after ∼250 iterations (not shown). One reason for the slow convergence is the fact that especially at the beginning many particles are outside the allowed search space and hence cannot be used to update $\boldsymbol{g}_b$ nor $\boldsymbol{l}_i^b$. Moreover, during the first ∼40 iterations the salary is often higher than the expenses, and some income is not spent on consumption (cf. second graphs of Figures 3). In these cases, utility could be improved by simply buying more goods without increasing work i.e., without decreasing leisure. Compared to *bound to border*, initialization has a higher influence on the results during the first iterations. With increasing number of iterations the influence of the initialization strategy is only marginal.



(a) Initialization at low work

(b) Initialization at high work

Fig. 3. Bounding strategy: let them fly

Discussion. Obviously, the bounding strategies have a big influence on the quality of the solutions and the convergence speed, while the influence of the initialization strategies is much lower, especially so after a certain number of iterations. *Bound to border* achieves optimal results with only a few iterations (∼70), while *let them fly* achieves almost-optimal results only after a larger number of iterations (∼250). However, it is important to note that fast convergence is *not* the aim of our approach. Indeed a human being is not expected to be able find the optimal tuple of goods within a few tries/iterations. It is more likely that an individual optimizes her/his decision by adjusting it step-by-step towards a decision that is assumed to be better than the current tuple. In the following, we discuss the two bounding strategies with respect to this point-of-view.

Although *bound to border* achieves the quantitatively best results, it is questionable if this strategy is able to plausibly simulate the behavior of consumers. Actually, the fast and exact search of this strategy does not allow for properly imitating the human decision making process, since it is incapable of considering the inexactness and curiosity of human individuals. Even during the first iterations, all money is spent on consumption, and it never happens that the salary exceeds the expenses. Moreover, it may never happen that an individual chooses a decision where consumption exceeds the salary. Thus, this strategy can be regarded as a setting for a very rational agent who would never consider non-realistic consumption-work combinations.

Let them fly appears to be a more human-like strategy, which allows agents to "look beyond the horizon" of affordable tuples. Obviously it may happen that an individual chooses a decision where the expenses increase the salary (e.g., due to unexpected short-term increase of prices). If so, it is not likely that this decision is completely discarded. It is more likely that this decision is retained, but that it is not considered as a "good" decision in the memory of an individual (i.e., no update in PSO). In particular, this scenario could be interpreted as a temptation towards some objective tuple an agent cannot afford with the current salary or under current prices, but may be able to afford in case of environmental changes. On the other hand, during the first iterations it may happen that some income is not spent on consumption – another human-like behavior. Hence, this bounding strategy is much better suited for modeling certain properties of the human decision making process, such as uncertainty and curiosity of individuals.

6 Conclusion and Future Work

In this paper we have presented a new agent based social simulation which uses Particle Swarm Optimization (PSO) to simulate the decision finding process of consumers. The analysis and evaluation of two different bounding strategies that map particles violating the underlying budget constraints to a feasible region have revealed interesting results. Although one bounding strategies is able to achieve very fast and stable convergence for the given optimization problem, the second bounding strategy, which needs significantly more iterations to find the optimal solution, appears to be more appropriate for simulating the uncertainty and curiosity involved in the human decision making process.

Scope of the presented study. This work has covered solely the performance of PSO for a single agent of an agent based Macroeconomic model – the performance of PSO in an environment with interacting agents has not been examined yet. Future studies that use a set of different communicating agents – as common in ABM – may build upon the findings of this work. Two other points which have not been investigated in this paper are the sharing of experiences of group leaders, a strategy which is often applied in cooperative co-evolution [23], and the intertemporal impact of savings, which may rise additional challenges. Several interesting questions are raised for future studies: the influence of savings and its inter-temporal utility, the influence of more elaborated bounding strategies that include the importance of distinct items (e.g., basic vs. luxury goods), the influence of different settings for the PSO swarm and their contributions to a desired search behavior, the influence of different PSO topologies and other SI/evolutionary computing techniques, and the investigation of differently formulated fitness functions that could simulate other heuristics, e.g., in accordance with research on consumer's decision in [21]. Moreover, the simulation of societies with social dynamics or changing parameters, and the sharing of experiences of group leaders are further interesting research questions.

References

[1] Gilbert, N.: Computer simulation of social processes. Social Research Update (6) (1994)

[2] Kennedy, J.: Swarm intelligence. In: Handbook of Nature-inspired and Innovative Computing, pp. 187–219 (2006)

[3] Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)

[4] Holland, J.L.: Adaption in natural and artificial systems. MIT Press (1992)

[5] Schelling, T.C.: Models of segregation. The American Economic Review 59(2), 488–493 (1969)

[6] Deffuant, G., Amblard, F., Weisbuch, G.: How can extremism prevail? a study based on the relative agreement interaction model. Journal of Artificial Societies and Social Simulation 5(4) (2002)

[7] Janssen, M.A., Jager, W.: An integrated approach to simulating behavioural processes: A case study of the lock-in of consumption patterns. Journal of Artificial Societies and Social Simulation 2(2) (1999)

[8] Davidsson, P.: Agent based social simulation: a computer science view. Journal of Artificial Societies and Social Simulation 5 (2002)

[9] Barbosa, M., de Lima Neto, F.: Distributed agent-based social simulations: An architecture to simulate complex social phenomena on highly parallel computational environments. In: IEEE Symposium on Intelligent Agent, pp. 1–8 (2011)

[10] Barbosa Filho, H.S., Lima Neto, F.B., Fusco, W.: Migration, communication and social networks – an agent-based social simulation. In: Menezes, R., Evsukoff, A., González, M.C. (eds.) Complex Networks. SCI, vol. 424, pp. 67–74. Springer, Heidelberg (2013)

[11] Meyer, M., Lorscheid, I., Troitzsch, K.G.: The development of social simulation as reflected in the first ten years of jasss: a citation and co-citation analysis. J. Artificial Societies and Social Simulation 12(4) (2009)

[12] Colander, D., Howitt, P., Kirman, A., Leijonhufvud, A., Mehrling, P.: Beyond DSGE models: Toward an empirically based macroeconomics. The American Economic Review 98(2), 236–240 (2008)

[13] Testfatsion, L.: Agent-based computational economics: Growing economies from the bottom up. Artificial Life 8(1), 55–82 (2002)

[14] Delli-Gatti, D., Desiderio, S., Gaffeo, E., Cirillo, P., Gallegati, M.: Macroeconomics from the bottom-up (2011)

[15] Bonabeau, E., Meyer, C.: Swarm intelligence: a whole new way to think about business. Harvard Business Review (5) (2001)

[16] Nenortaite, J., Butleris, R.: Application of particle swarm optimization algorithm to decision making model incorporating cluster analysis. In: IEEE Conference on Human System Interactions, pp. 88–93 (2008)

[17] Devi, S., Panigrahi, S.K.: Intelligent decision making using particle swarm optimization for optimizing product-mix model. J. Comp. Sc. & Inf. 1 (2011)

[18] Kampouridis, M.: Computational intelligence in financial forecasting and agent-based modeling: Applications of genetic programming and self-organizing maps. Technical report, PhD Thesis. University of Essex (2011)

[19] Xue, Z.: A particle swarm optimization based behavioral and probabilistic fire evacuation model incorporating fire hazards and human behaviors. M.S. Thesis, State University of New York at Buffalo (2006)

[20] Cui, X., Potok, T.: A particle swarm social model for multi-agent based insurgency warfare simulation. In: 5th ACIS International Conference on Software Engineering Research, Management Applications, SERA 2007, pp. 177–183 (2007)

[21] Bettman, J., Johnson, E., Payne, J.: Consumer decision making. In: Handbook of Consumer Behavior, pp. 50–79 (1991)

[22] Meeusen, W., van Den Broeck, J.: Efficiency estimation from cobb-douglas production functions with composed error. Int. Economic Review, 435–444 (1977)

[23] Potter, M.A., De Jong, K.A.: A cooperative coevolutionary approach to function optimization. In: Davidor, Y., Männer, R., Schwefel, H.-P. (eds.) PPSN 1994. LNCS, vol. 866, pp. 249–257. Springer, Heidelberg (1994)

Multi-agent Oriented Stable Payoff with Cooperative Game

Tianwen Li, Feng Ma, and Weiyi Liu

School of Information Science and Engineering, Yunnan University, Kunming, China
liuweiyi2000@yahoo.com.cn

Abstract. In the field about multi-agent system, the payoff rationality is an important factor for the forming of a multi-agent coalition structure. In this paper, we regard a payoff vector belonging to the bargaining set in classical cooperative game as a stable payoff vector of multi-agents. Then, we propose an approach to find the stable payoff vector based on genetic algorithm. Finally, the experimental results and analysis are showed about success rates and running times of our proposed approach.

Keywords: Multi-agent system, Cooperative games, Stable payoff vector, Bargaining set.

1 Introduction

Multi-agent System (MAS) is a sub-field of Distributed Artificial Intelligence (DAI)[1]. In MAS, multi-agents implement given tasks cooperatively and highly efficiently. More coalitions formed by multi-agents is a partition of the multi-agents set as coalition structure[2], [3], [4]. For the coalition structure, each cooperative egent can obtain the relative avail as its payoff. The payoffs of all agents are a evaluating standard for the rationality of a coalition structure, so the payoff issue has become a research hotspot in the field of MAS.

The payoff issue of MAS can be regarded as complex interactions among rational agents. Furthermore, game theory [6] can provide many mathematic models for complex interactions among rational decision makers, which involves cooperative games and non-cooperative games [6], [7], [8]. Some ideas and solutions of cooperative games have been used to obtain rational payoffs in many different aspects of MAS. Ghazikhani et al. [3] and Wei-Yi Liu et al. [5] use the solution of Shapley value to obtain a fair payoff in MAS, which is one of important solutions in cooperative games. However it is a complex problem to obtain precise Shapley values [16]. Parag [15] uses the solutions of coalition rationality and core to solve a stable payoff in MAS. However core is sometimes an empty set [11]. Onn Shehory et al. [17] uses the solutions of kernel and local Pareto optimality to learn the formation and payoff of coalition in MAS and give a means of obtaining stable payoff in non-superadditive environments.

Bargaining set given by Aumann and Maschler [9] is another important solution in cooperative games, which is a set of stable payoff vectors. Davis and Maschler [10]

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 74–81, 2013.

and Peleg [11] have proved that bargaining set is not empty. And core belongs to bargaining set if core is not empty. However, the definition of the bargaining set only provides a method to test a payoff vector whether to belong to the bargaining set or not, and it does not even provide a method to compute any stable payoff vector to the players. Motivated by the stability of bargaining set, we propose an approach to find a stable payoff vector for a coalition structure of MAS.

In the rest of the paper is organized as follows. In Section 2, related basic concepts and theorems about our approach are given. Section 3 details the finding approach to obtain a stable payoff vector for MAS. Experimental results and analysis about success rates and running times are reported in Section 4. We conclude this paper in Section 5.

2 Preliminaries

2.1 Multi-agent System

The following assumptions [17] are necessary for the definition of MAS in our approach.

- Every agent can access resources, tasks, and payoff values of other agents.
- Agents can costly communicate with other agents and make agreements.
- Agents can use a monetary system to evaluate resources and productions. Resources and money can be transferable among the agents.

Let n agents as $N=\{1,2,\cdots,n\}$, each of which have its own resources. Many agents of them can cooperate as a coalition S to execute given tasks and obtain their correlative avail, named the coalition value $v(S)$. In the coalition S, each agent attempts to maximize its individual payoff, which is a evaluation foundation for a coalition. Based on the evaluations of all agents, c*oalition structure* [9] $\beta=\{B_1,\cdots,B_m\}$ for N can be formed as a partition of N, where $B_j \bigcap B_k = \phi, j \neq k, \bigcup_{j=1}^{m} B_j = N$. A *payoff vector* for the c*oalition structure of* MAS is a vector x, $x \in \mathbb{R}^n$, and the i th element of x shows that the i th player "receives" x_i. And x must satisfy two following conditions.

- **Efficient:** The payoff vector x is called efficient if $\sum_{i \in B_j} x_i = v(B_j), B_j \in \beta$.
- **Individual Rational:** The payoff vector x is called *individual rational* if x_i is not less than the worth that the agent i can obtain when he works alone, i.e. $x_i \geq v(\{i\}), i \in 1,2,\cdots,n$.

It is necessary to be noted that some coalitions of multi-agents are impossible such as the situation of payout more than income. The set of these coalitions that agents can form is called as Coalition Set (CS).

2.2 Cooperative Game

A cooperative game is a model that focuses on the cooperative behaviors of coalitions of players. Let $N=\{1,2,\cdots,n\}$ be a nonempty finite set of players, and an element of 2^n be called a coalition. A *cooperative game with transferable utility* (N,v) is given by a characteristic function denoted as $v:2^n \to \mathbb{R}$ with $v(\phi)=0$. In cooperative game, if (N,v) is a game and β is a coalition structure for N, then the triple (N,v,β) is called *cooperative game with coalition structure*. If there will be no confusion, we denote the cooperative game with coalition structure (N,v,β) by v. A payoff configuration (PC) is defined as an expression of the form $(x;\beta)=(x_1,x_2,\cdots x_n;B_1,B_2\cdots B_m)$. The set $I(v,\beta)$ of individually rational PC for v is denoted as

$$I(v,\beta)=\{x\in \mathbb{R}^n \mid \sum_{i\in B_j} x_i = v(B_j), for\,all\; j\in 1,2,\cdots,m \;\; and \;\; x_i \geq v(\{i\}), for\,all\, i\in N\} \tag{1}$$

2.3 Bargaining Set and Relative Concepts

In cooperative game, bargaining set given by Aumann and Maschler [9] is an important solution concept, which can give stable payoff vectors. In the bargaining set, (N,v,β) is a cooperative game with coalition structure, and $x\in I(v,\beta)$. An *objection* of one player k against another l at x is a pair (S,y) where $k,l\in B\in\beta$, S is a coalition containing k and not containing l, and y is a vector in $\mathbb{R}^S$ satisfying $\sum_{i\in S} y_i = v(S)$ and $y_i > x_i$ for all $i\in S$. A *counter-objection* to the objection is a pair (T,z), where T is a coalition containing l and not containing k and z is a vector in $\mathbb{R}^T$ satisfying $\sum_{i\in T} z_i = v(T)$, $z_i \geq y_i$ for all $i\in S\bigcap T$, and $z_i \geq x_i$ for all $i\in T\setminus S$. A payoff vector x belongs to the classical bargaining set $M(v,\beta)$, if there exists a counter-objection for any objection of one player against another at x. There is a counter-objection for each objection, so $M(v,\beta)$ is a set of stable payoff vectors.

Several following definitions and assumptions about bargaining set are also necessary for our approach. The definitions of excess and surplus given by Morton Davis and Michael Maschler [10] were defined as follows. Let v be a cooperative game with coalition structure. Let $S\in CS$ and $x\in I(v,\beta)$. The *excess* of S at x is $e(S,x)=v(S)-\sum_{i\in S} x_i$. Let v be a cooperative game with coalition structure. Let $S\in CS$, $k,l\in B\in\beta$, $k\neq l$, and $x\in I(v,\beta)$. The *surplus* of k over l at x is $S_{kl}(x)=\max e(S,x)$. If $S_{kl}(x)>S_{lk}(x)$ and $x_l > v(\{l\})$, then we say that k *outweighs* l at x and write $k \succ_x^K l$.

Note that, the coalition S can be used as an objection if and only if $e(S,x)>0$. If $S_{kl}(x)>0$, there exists a positive $e(S,x)$, where $S\in CS, k\neq l$, and then there exists an objection of k against l at x. So $S_{kl}(x)$ can be as a strength of k against l at x. The strength of k against l is greater than that of l against k, if k outweighs l at x and $S_{kl}(x)>0$.

Theorem 1. (Peleg and Sudhölter [11]) Let v be a cooperative game with coalition structure. Let $x\in I(v,\beta)$ and $k,l\in B\in\beta$, where $k\neq l$. If $l\succ_x^K k$, there exists a counter-objection for each objection of k against l at x.

In the cooperative games, Schmeidler [10] proposed the definition of the lexicographical ordering. Let v be a multi-choice game with coalition structure and $x\in I(v,\beta)$. The lexicographical ordering $\theta(x)$ of x can be defined as

$$\theta(x)=(\theta_1(x)=e(S_1,x),\theta_2(x)=e(S_2,x),\cdots\theta_k(x)=e(S_k,x),\cdots,\theta_H(x)=e(S_H,x)) \tag{2}$$

where the various excesses of coalitions are arranged in decreasing order. When $\theta_t(x)<\theta_t(y)$ for j=t and $\theta_j(x)=\theta_j(y)$ for $1\leq j<t$, where $1\leq t\leq H$, $\theta(x)$ is lexicographically smaller than $\theta(y)$, denoted as $\theta(x)<_L\theta(y)$, where $x,y\in I(v,\beta)$. When $\theta(x)<_L\theta(y)$, the most dissatisfaction degree of players to the payoff vector y is commonly more than that to the payoff vector x, that is to say, players can be more willing to accept the more stable payoff vector x. So the stability of a payoff vector can be judged in the means of comparing lexicographical ordering.

3 Obtaining a Stable Payoff Vector

Clearly, bargaining set is a set of stable payoff vectors, and core is a subset of bargaining set [9]. For each $x\in I(v,\beta)$, it provides a stable payoff distribution. However the definition of the bargaining set only provides a method to test a payoff vector whether to belong to the bargaining set or not, and it does not even provide a method to compute any stable payoff vector to the players. In this section, we propose an approach to obtain a stable payoff vector belonging to bargaining set for the coalition structure of multi-agents.

Motivated by Theorem 1, we can get an idea to obtain a payoff vector belonging to bargaining set, if there does not exist any k and l satisfying $k\succ_x^K l$ or $l\succ_x^K k$, where $k,l\in B\in\beta$ and $k\neq l$. Furthermore, lexicographical ordering can be used to evaluate the stabilities of payoff vectors. Moreover, The genetic algorithm [13], [14] is a global random optimization method, which has so many advantages, such as concurrent searching and colony optimize, and has been applied successfully in many fields of NP complete problems. Based on Theorem 1, lexicographical ordering, and genetic algorithm, we can solve a stable payoff vector to make multi-agents stably cooperative to implement given tasks.

We first give several important concepts about genetic algorithm as follows.

Individual Coding: An individual is coded in the form of floating point number, which is a payoff vector belonging to $I(v,\beta)$.

Fitness: The lexicographical orderings of any individual to selected coalitions can be obtained. The fitness of an individual is the order number in descending order of the lexicographical orderings about all individuals of a population. The less the order number of an individual is, the larger its fitness is.

Selection: Population can be selected according as the fitness in the form of a rotating disc game.

Crossover: We randomly select two individual, which are two payoff vectors, x and y. We randomly obtain a coalition B from the coalition structure β, where $B\in\beta$. Then, for each agent in B, two corresponding payoff values in x and y are reset to their average value.

Mutation: We randomly select two members, $k,l\in B\in\beta$, and solve $S_{kl}(x)=e(x,S)$ and $S_{lk}(x)=e(x,T)$, where $k,l\in B\in\beta$. If $|S_{kl}(x)-S_{lk}(x)|>\alpha$, the payoff values of k respectively increase δ and the payoff values of l respectively decrease δ, when the individual rationality can be satisfied, where $\alpha>0$ and $\alpha\geq\delta>0$. If $|S_{kl}(x)-S_{lk}(x)|\leq\alpha$, the individual need not mutate.

Based on the above concepts, a stable payoff vector can be obtained, when no selected individual mutates for continuous w generation times, that is to say, $S_{kl}(x)=S_{lk}(x)$ for every selected individual. The approach to find a stable payoff vector can be showed as Algorithm 1.

Algorithm 1.

Step 1. Initialize a population.

Step 2. Calculate the fitness for each individual in the population.

Step 3. Select the population in the form of a rotating disc game.

Step 4. Generate new population by randomly pair-matched crossover.

Step 5. Select an individual from the population randomly. $num=0$ when the individual can mutate, otherwise $num=num+1$.

Step 6. Goto Step 2, if the generation number is not reached, and $num<w$. Otherwise, the best stable payoff vector can be obtained, whose lexicographical ordering is smallest.

4 Experiments and Analysis

In this section, we implement Algorithm 1 and make performance studies about success rates and running times of our proposed approach, by changing the values of three parameters α, and w. It is worth to be noted that the experiments about success rates only consider the change of w, because the parameters α and δ mainly influence the precision of the obtained stable payoff vector and little affect the success rates to find a stable payoff vector. The experiments are based on the machine platform of Intel Core2 Duo P8600 CPU

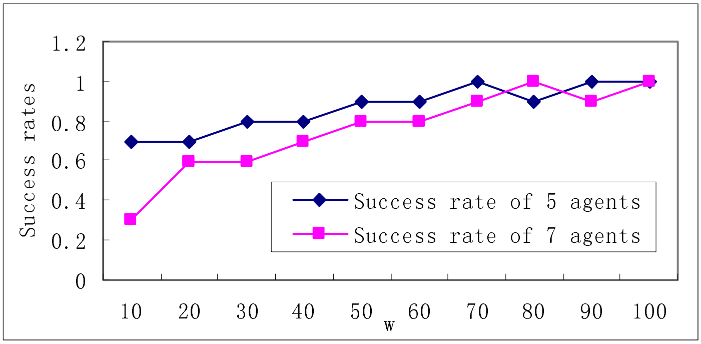


Fig. 1. Success rates of obtaining a stable payoff vector under $\delta = 0.005$ and $\alpha = 0.01$ for 5 agents and those for 7 agents, when w varies from 0 to 100 in the step of 10

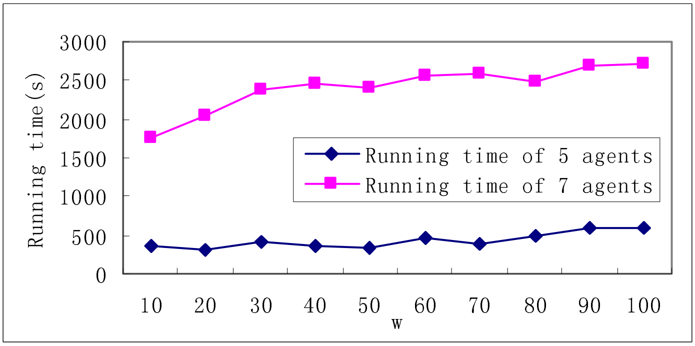


Fig. 2. Running times of obtaining a stable payoff vector under $\delta = 0.005$ and $\alpha = 0.01$ for 5 agents and those for 7 agents, when w varies from 0 to 100 in the step of 10

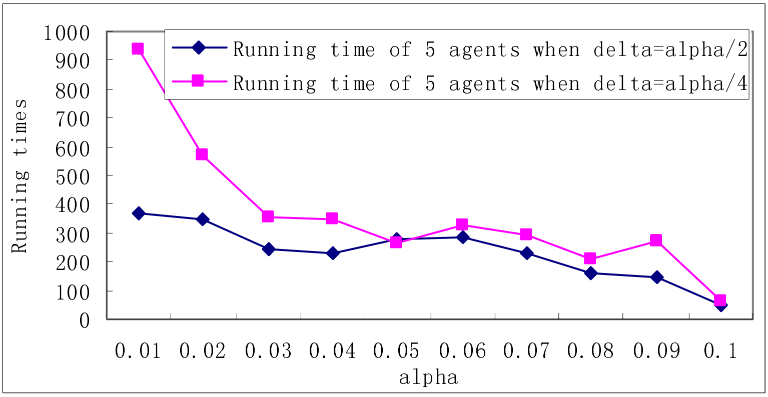


Fig. 3. Running times under $w = 100$ for 5 agents respectively when $\delta = \alpha / 2$ and $\delta = \alpha / 4$, when α varies from 0 to 0.1 in the step of 0.01

2.40GHz, 2GB main memory and Windows 7 operating system. And all codes were written in Matlab 7.6.0 (R2008a).

When w varies from 0 to 100 in the step of 10, two curves in Fig. 1 respectively show the success rates of obtaining a stable payoff vector under $\delta = 0.005$ and $\alpha = 0.01$ for 5 agents and those for 7 agents. For a fixed w, generating a stable payoff vector using Algorithm 1 is regarded as a success, and the ratio of success times to the total test times is named as the success rate under the w. For a given agent number, the success rates are gradually increasing with the increasing of w. For a fixed w, the success rates are gradually decreasing with the increasing of the agent number.

When w varies from 0 to 100 in the step of 10, two curves in Fig. 2 respectively show the running times of obtaining a stable payoff vector under $\delta = 0.005$ and $\alpha = 0.01$ for 5 agents and those for 7 agents. For a given agent number, the running times are gradually increasing with the increasing of w. For a fixed w, the running times are also gradually increasing with the increasing of the agent number.

When α varies from 0 to 0.1 in the step of 0.01, two curves in Fig. 3 show the running times under $w = 100$ for 5 agents respectively when $\delta = \alpha / 2$ and $\delta = \alpha / 4$. For a given agent number, the running times are gradually decreasing with the increasing of α. For a fixed α, the running times are gradually decreasing with the increasing of δ.

5 Conclusions

We summarize the main contributions of this paper as follows. We regard a payoff vector belonging to bargaining set as a multi-agents oriented stable payoff vector. We proposed an approach to find the stable payoff vector based on genetic algorithm. The experimental results and analysis about success rates and running times for our proposed approach are showed.

Acknowledgements. This work is supported by the National Natural Science Foundation of China (Nos. 61063009, 61163003), and the Ph.D. Programs Foundation of Ministry of Education of China (No.20105311200001), the Yunnan Provincial Foundation for Leaders of Disciplines in Science and Technology (2012HB004), the Foundation for Key Program of Department of Education of Yunnan Province (2011Z015).

References

1. Bond, A.H., Gasser, L.: An analysis of problems and research in DAI. In: Readings in Distributed Artificial Intelligence, pp. 3–35. Morgan Kaufmann, San Mateo (1988)
2. Shehory, O., Kraus, S.: Coalition Formation among Autonomous Agents: Strategies and Complexity. In: Müller, J.P., Castelfranchi, C. (eds.) MAAMAW 1993. LNCS (LNAI), vol. 957, pp. 57–72. Springer, Heidelberg (1995)

3. Ghazikhani, A., Mashadi, H.R., Monsefi, R.: A novel algorithm for coalition formation in Multi-Agent Systems using cooperative game theory. In: 2010 18th Iranian Conference on Electrical Engineering (ICEE). IEEE (2010)
4. Sandholm, T., Larson, K., Andersson, M., Shehory, O., Tohmé, F.: Anytime coalition structure generation with worst case guarantees. In: Proc. AAAI 1998, California, pp. 46–54 (1998)
5. Liu, W.Y., Yue, K., Wu, T.Y., Wei, M.J.: An Approach for Multi-objective Categorization Based on the Game Theory and Markov Process. Applied Soft Computing (2011) (accepted manuscript) (in Press)
6. Nash, J.: Non-cooperative Games. Annals of Mathematics 54, 286–295 (1951)
7. Aumann, R.J., Peleg, B.: Von Neumann-Morgenstern solutions to cooperative games without side payments. Bulletin of the American Mathematical Society 66, 173–179 (1960)
8. Shapley, L.S.: A value for n-person Games. In: Kuhn, H.W., Tucker, A.W. (eds.) Contributions to be Theory of Games, pp. 307–317. Princeton University Press (1953)
9. Auman, R.J., Maschler, M.: The bargaining sets for cooperative games. Princeton University (1961)
10. Davis, M.D., Maschler, M.: Existence of stable payoff configurations for cooperative games. Econometric Research Program. Princeton University (1962)
11. Peleg, B., Sudhölter, P.: Introduction to the theory of cooperative games, 2nd edn., pp. 52–61. Springer, Heidelberg (2007)
12. Schmeidler, D.: The Nucleolus of a characteristic function Game. SIAM Journal of Applied Mathematics 17, 1163–1170 (1969)
13. Holland, J.H.: Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor (1975)
14. Srinivas, M., Patnaik, L.M.: Genetic algorithms: a survey. Computer 27(6), 17–26 (1994)
15. Pendharkar, P.C.: Game theoretical applications for multi-agent systems. Expert Systems with Applications 39(1), 273–279 (2012)
16. Conitzer, V., Sandholm, T.: Computing Shapley values, manipulating value division schemes, and checking core membership in multi-issue domains. In: Proceedings of the National Conference on Artificial Intelligence, pp. 219–225 (2004)
17. Shehory, O., Kraus, S.: Feasible formation of coalitions among autonomous agents in nonsuperadditive environments. Computational Intelligence 15(3) (1999)

Use the Core Clusters for the Initialization of the Clustering Based on One-Class Support Vector Machine

Lei Gu[1,2]

[1] The Key Laboratory of Embedded System and Service Computing,
Ministry of Education, Tongji University, Shanghai, 200092, China
[2] School of Computer Science and Technology,
Nanjing University of Posts and Telecommunications, Nanjing, 210023, China
gulei@njupt.edu.cn

Abstract. The clustering method based on one-class support vector machine has been presented recently. Although this approach can improve the clustering accuracies, it often gains the unstable clustering results because some random datasets are employed for its initialization. In this paper, a novel initialization method based on the core clusters is used for the clustering algorithm based one-class support vector machine. The core clusters are gained by constructing the neighborhood graph and they are regarded as the initial datasets of the clustering algorithm based one-class support vector machine. To investigate the effectiveness of the proposed approach, several experiments are done on four datasets. Experimental results show that the new presented method can improve the clustering performance compared to the previous clustering algorithm based on one-class support vector machine and k-means approach.

Keywords: Initialization, One-class support vector machine, Neighbor graph, Core clusters, Clustering methods.

1 Introduction

The aim of data clustering methods is to divide data into several homogeneous groups called clusters, within each of which the similarity or dissimilarity between data is larger or less than data belonging to different groups[1]. Unsupervised clustering partitions all unlabeled data into a certain number of groups on the basis of one chosen similarity or dissimilarity measure[2,3]. Different measure of the similarity or dissimilarity can lead to various clustering methods such as k-means[4], fuzzy c-means[5], mountain clustering, subtractive clustering[6] and neural gas[7]. In these traditional clustering algorithms, k-means, which can be easily implemented, is the best-known squared error- based clustering algorithm. Recently, a novel kernel method for clustering based on one-class support vector machine(COSVM) was presented in [8]. This kernel-based clustering method can be implemented in a similar way to the classical k-means and use a one-class support vector machine as the

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 82–88, 2013.

description of each cluster rather than the center of several data. Experiments on real datasets show that the clustering algorithm in [8] is valid and can have encouraging performance.

However, this COSVM clustering algorithm is easily affected by the random initial datasets. So it often has the unstable clustering performance. To solve this problem, an initialization method based on the core clusters is applied to the COSVM (CC-COSVM) in this paper. Several core clusters can be produced by constructing the σ−neighborhood graph by the way in [9] and they also are regarded as the initial datasets of the CC-COSVM. Data points included by the core clusters are a part of the whole dataset. The number of core clusters is equal to the number of clusters. Experimental results show that the proposed CC-COSVM approach can not only obtain the stable clustering performance, but also improve the clustering accuracies when compared to the COSVM and KM method.

The remainder of this paper is organized as follows. Section 2 reports the COSVM clustering algorithm. In Section 3, the new proposed CC-COSVM clustering approach is formulated. Some experimental results are shown in Section 4 and Section 5 give some conclusions.

2 The COSVM Clustering Algorithm

The COSVM clustering algorithm is a kernel-based clustering approach. Inspired by the KM method, it can gain the better clustering accuracies than the KM. The key step is that the one-class support machine is trained.

Firstly, the one-class support vector machine method is introduced. One-class support vector machine is a kernel-based data domain description method. It tries to find the smallest sphere containing all input data in the feature space. Assume that a nonempty set of cluster m in the d-dimensional space R^d is X^m and $X^m = \left\{x_1^m, \quad x_2^m, \cdots x_n^m\right\}$. Now we construct a smallest sphere S_m for cluster m that can enclose all points x_i^m ($x_i^m \in X^m, i = 1, 2, \cdots, n$). This problem is considered as a quadratic optimization as follows[10]:

$$\min_{a,R} R \ \ s.t. \ \left(x_i^m - a\right)^T \left(x_i^m - a\right) \le R^2, i = 1, 2, \cdots, n \tag{1}$$

where a is the center of the sphere S_m and R is its radius. The constraints introduce slack variables ξ_i as follows:

$$\min_{a,R,\xi_i} R + C\sum_{i=1}^{n} \xi_i \ \ s.t. \ \left(x_i^m - a\right)^T \left(x_i^m - a\right) \le R^2 + \xi_i \tag{2}$$

where $\forall i, \xi_i \ge 0$ and the variable C gives the trade-off between the volume of the sphere and the number of target objects rejected. This is a convex optimization problem. Therefore, we can use Lagrange multipliers to guarantee it to converge to the global minimum:

$$L(R,a,\xi_i)=R^2-\sum_{i=1}^{n}\beta_i\,\xi_i+C\sum_{i=1}^{n}\xi_i-\sum_{i=1}^{n}\alpha_i\left(R^2+\xi_i-\left(x_i^m-a\right)^T\left(x_i^m-a\right)\right) \quad (3)$$

where $\forall_i,\alpha_i\geq 0,\beta_i\geq 0$. The minimization problem of Eq.(3) can be transformed into the maximization problem of the Wolfe dual form[11] as follows:

$$L=\sum_{i=1}^{n}\alpha_i\left(x_i^m\cdot x_i^m\right)-\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j\left(x_i^m\cdot x_j^m\right) \quad (4)$$

with the constraints $\forall_i,0\leq\alpha_i\leq C$ and $\sum_{i=1}^{n}\alpha_i=1$.

In real-world applications, data is not spherically distributed, even when the most outlying data points are excluded. To make more flexible descriptions of a class, the kernel functions can be applied to transforming the data examples into a high-dimensional feature space F by a nonlinear mapping $\Phi:R^d\rightarrow F$ instead of the Euclidean inner product often used as a similarity measure in a variety of fields, like pattern classification and data clustering. Let K be a positive kernel function. The inner product in F can be computed through the kernel function K in R^d :

$$K\left(x_i,x_j\right)=\Phi\left(x_x\right)\cdot\Phi\left(x_j\right) \quad (5)$$

After solving the problem of Eq.(4) by using the Karush-Kuhn-Tucker conditions[12], we can get the smallest sphere S_m of cluster m. When the smallest sphere of each cluster is attained, we finish the training process of one-class support vector machine in one iteration step of the COSVM.

Although the centers of the smallest spheres cannot be explicitly expressed, they are regarded as the clusters presented by the smallest spheres in the COSVM. Let x be any unlabeled data point. The distance D between x and the center of the sphere S_m can be calculated as the following formula:

$$D\left(x,S_m\right)=K\left(x,x\right)-2\sum_{i=1}^{n}\alpha_iK\left(x_i^m,x\right)+\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_jK\left(x_i^m,x_j^m\right) \quad (6)$$

Consequently, the COSVM gives x one cluster c by the following decision rule:

$$c=\underset{m=1,2,\cdots,k}{\arg\min}\,D\left(x,S_m\right) \quad (7)$$

where k is the number of clusters, $D\left(x,S_m\right)$ can be calculated by the Eq.(6) and $D\left(x,S_h\right)<\rho$ where ρ is a parameter explained in [8] and [13].

Secondly, suppose that the unlabeled dataset $X = \{x_1, x_2, \cdots, x_N\}$ in the d-dimensional space R^d where N is the number of all data points and X can be divided into k clusters. The procedure of the COSVM clustering algorithm can be formulated as follows:

Step1. In the initialization, some data points are selected randomly from X and are partitioned into k clusters.

Step2. After train a one-class support vector machine for each cluster, assign each data point from X to one cluster by Eq.(6) and (7).

Step3. If all elements of each cluster are not changed or the iteration times are equal to the maximum, then goto Step4; otherwise add the iteration times to 1 and goto Step2.

Step4. End the COSVM clustering algorithm.

3 The Proposed CC-COSVM Clustering Algorithm

Although the COSVM clustering algorithm can get the better clustering accuracies, it is sensitive to the initialization method. The random initial datasets of the COSVM clustering lead to the unstable clustering performance.

The CC-COSVM clustering method can overcome this weakness. In this proposed algorithm, several core clusters are used for its initialization. Firstly, assume that the unlabeled dataset $X = \{x_1, x_2, \cdots, x_N\}$ in the d-dimensional space R^d where N is the number of all unlabeled data points. The core clusters are formed as follows[9]:

Step1. Set $\sigma = l$.

Step2. Construct the σ–neighborhood graph. In this graph, each data point from X is used as one node and two data points x_i and x_j can be connected by and edge when $\frac{\|x_i - x_j\|}{\max\limits_{\mu=1,2,\cdots,N,\, \nu=1,2,\cdots,N}\left(\|x_\mu - x_\nu\|\right)} < \sigma$ where $\|\cdot\|$ represents the Euclidean norm.

Step3. Let $G(\sigma)$ be the number of components of the σ–neighborhood graph which should contain at least T vertices.

Step4. Let $\sigma = \sigma + t$. If $\sigma > 1$, then goto Step5; otherwise goto Step2.

Step5. Let $\sigma^* = \min\left(\arg\max\limits_{\sigma \in (0,1]} \left(G(\sigma)\right)\right)$.

Step6. $G(\sigma^*)$ components of the σ*–neighborhood graph are regarded as $G(\sigma^*)$ core clusters. $G(\sigma^*)$ should contain at least T vertices.

Note that $l = 0.001$ and $t = 0.05$ in [9].

Secondly, let k be the number of clusters. The CC-COSVM clustering algorithm based on core clusters is given as follows:

```
Step1. Let $H = 2$.
Step2. Obtain some core clusters according to the way mentioned above.
Step3. If the number, $G\left(\sigma^{*}\right)$, of core clusters is equal to the number, $k$, of clusters,
       then goto Step4; otherwise let $H = H + 1$ and goto Step2.
Step4. $k$ core clusters are viewed as $k$ initial datasets for the CC-COSVM.
Step5. After train a one-class support vector machine for each cluster, assign each
       data point from $X$ to one cluster by Eq.(6) and (7).
Step6. If all elements of each cluster are not changed or the iteration times are equal
       to the maximum, then goto Step7; otherwise add the iteration times to 1 and
       goto Step5.
Step7. End the CC-COSVM clustering algorithm.
```

Because same core clusters can be obtained when the proposed CC-COSVM clustering algorithm runs very time, this new method can gain the stable clustering performance.

4 Experimental Results

To demonstrate the effectiveness of the proposed CC-COSVM clustering algorithm, we compared it with the classical KM clustering method and the previous COSVM clustering approach on one artificial dataset and four UCI real datasets[14], referred to as the DUNN[15], Iris, Ionosphere and Haberman datasets. As shown in Fig.1, one artificial dataset DUNN collects 90 2-dimensional instances belonging to two classes. The Iris dataset contains 150 cases with 4-dimensional feature from three classes. The Ionosphere and Haberman datasets are three 34, 3 dimensional datasets with 351 and 306 samples respectively.

The Gaussian kernel function in the Eq.(8) is applied to the kernel-based COSVM

$$K\left(x_i, x_j\right) = \exp\left(-\left\|x_i - x_j\right\|^2 / 2\delta^2\right) \tag{8}$$

and CC-COSVM clustering algorithms. The better kernel parameter can be selected from the set $\left\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\right\}$ for each dataset. We set the regularization parameter $C = 1$ according to [8] and employ the same modification with [14], in which the parameter ρ related to the Eq.(7) is not used. Furthermore, a Matlab's function called quadprog is used to solve the quadratic optimization problems for the COSVM and CC-COSVM, and all experiments are done by Matlab on WindowsXP operating system.

The average clustering performance of the KM, COSVM and CC-COSVM are shown in Table 1 on 20 independent runs for each dataset. As shown in Table 1, we can see that the new CC-COSVM can achieve the best clustering accuracies than the KM and COSVM algorithms and because it always can gain the minimum standard deviations, it is the more stable algorithm. Note that the number in the parenthesis is the corresponding stand deviations in Table 1.

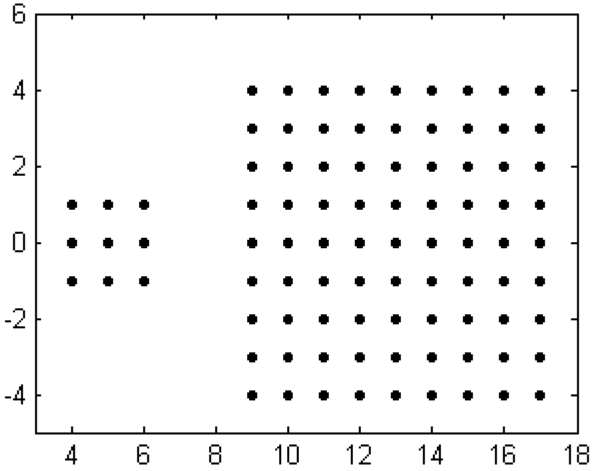


Fig. 1. The Dunn dataset

Table 1. Comparison of the average clustering performance on 20 runs

Datasets	Clustering Performance (%)		
	KM	COSVM	CC-COSVM
DUNN	68.00 (5.23)	72.28 (5.85)	**100.00 (0.00)**
Iris	83.57 (13.61)	87.30 (5.81)	**93.33 (0.00)**
Haberman	55.11 (8.96)	68.27 (6.34)	**69.61 (0.00)**
Ionosphere	51.23 (2.97)	61.92 (6.07)	**70.94 (0.00)**

5 Conclusions

In this paper, we propose the novel initialization method for the COSVM clustering algorithm called the CC-COSVM clustering method. This new approach uses several core clusters formed by constructing the σ−neighborhood graph for the initialization of the CC-COSVM clustering. Some experimental results demonstrate that the CC-COSVM clustering can obtain the better and more stable clustering performance than the traditional KM and COSVM algorithms.

Acknowledgements. This work is supported by the Open Foundation of the Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, China (No.2011-01). This work is also supported by the Scientific Research Foundation of Nanjing University of Posts and Telecommunications (No. NY210078).

References

1. Fillippone, M., Camastra, F., Masulli, F., Rovetta, S.: A survey of kernel and spectral methods for clustering. Pattern Recognition 41(1), 176–190 (2008)
2. Jain, A.K., Murty, M.N., Flyn, P.J.: Data clustering: a review. ACM Computing Surveys 32(3), 256–323 (1999)
3. Xu, R., Wunsch, D.: Survey of clustering algorithms. IEEE Transactions on Neural Networks 16(3), 645–678 (2005)
4. Tou, J.T., Gonzalez, R.C.: Pattern recognition principles. Addison-Wesley, London (1974)
5. Bezdek, J.C.: Pattern recognition with fuzzy objective function algorithms. Plenum Press, New York (1981)
6. Kim, D.W., Lee, K.Y., Lee, D., Lee, K.H.: A kernel-based subtractive clustering method. Pattern Recognition Letters 26(7), 879–891 (2005)
7. Martinetz, T.M., Berkovich, S.G., Schulten, K.J.: Neural-gas network for vector quantization and its application to time-series prediction. IEEE Transactions on Neural Networks 4(4), 558–569 (1993)
8. Camastra, F., Verri, A.: A novel kernel method for clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(5), 801–805 (2005)
9. Ormella, C., Anastasios, M., Sandhya, S., Don, K., Sijia, L., Philip, K.M., Radek, E.: DifFUZZY: a fuzzy clustering algorithm for complex datasets. International Journal of Computational Intelligence in Bioinformatics and Systems Biology 1(4), 402–417 (2010)
10. Gu, L., Sun, F.C.: Two novel kernel-based semi-supervised clustering methods by seeding. In: Proceedings of the 2009 Chinese Conference on Pattern Recognition (2009)
11. Wolfe, P.: A duality theorem for nonlinear programming. Q. Appl. Math. (19), 239–244 (1961)
12. Kukn, H.W., Tucker, A.W.: Nonlinear programming. In: Proceedings of Second Berkeley Symposium on Mathematical Statistics and Probability, pp. 481–492 (1951)
13. Bicego, M., Figueiredo, M.A.T.: Soft clustering using weighted one-class support vector machines. Pattern Recognition 42(1), 27–32 (2009)
14. UCI Machine Learning Repository, `http://www.ics.uci.edu/~mlearn/MLSummary.html`
15. Dunn, J.: A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters. Journal of Cybernet. (3), 32–57 (1974)

Terrain Image Classification with SVM

Mu-Song Chen[1], Chi-Pan Hwang[2], and Tze-Yee Ho[3]

[1]Da-Yeh University
[2]National Changhua University of Education
[3]Feng Chia University
chenms@mail.dyu.edu.tw, cphwang@cc.ncue.edu.tw, tyho@fcu.edu.tw

Abstract. Remote sensing is an important tool in a variety of scientific researches which can help to study and solve many practical environmental problems. Classification of remote sensing image, however, is usually complex in many respects that a lot of different ground objects show mixture distributions in space and change with temporal variations. Therefore, automatic classification of land covers is of practical significance to the exploration of desired information. Recently, support vector machine (SVM) has shown its capability in solving multi-class classification for different ground objects. In this paper, the extension of SVM to its online version is employed for terrain image classification. An illustration of online SVM learning and classification on San Francisco Bay area is also presented to demonstrate its applicability.

Keywords: online SVM, Remote sensing, Terrain image classification.

1 Introduction

Support vector machine (SVM) is an efficient tool for supervised learning and has been proven to outperform most other systems in a wide variety of application problems [1][2][3]. The SVM originates from the maximal margin decision strategy [4]. This formulation leads to minimization of the structural risk and to favorable generalization capabilities of the classifier which is justified by both the theoretical studies and the experimental evidence. Nevertheless, several problems are considered when the SVMs are applied for terrain image classifications. Firstly, the supervised SVM learning depends on large amount of labeled samples to improve its accuracy. Usually, the computation complexity of the SVM algorithm for optimizing a quadratic programming (QP) problem [5] is $O(n^3)$, where n is the number of training samples. Thus, the training time of the SVM becomes unacceptable whenever n is large. Secondly, labeled samples can be scarce and expensive to generate, while unlabeled data are frequently easy to obtain in large quantities in many real-world situations. For example, in the terrain image classification using Synthetic Aperture Radar (SAR) imagery, only a small portion of objects or ground truths in the image is available correctly. These unlabeled objects should strongly rely on the unsupervised learning methods to identify interested objects. Thirdly, many machine learning problems can be viewed as an online version rather than the batch modes. Indeed, the

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 89–97, 2013.

data is often collected continuously in time and the concepts to be learned may also evolve in time. In this case, the SVM model needs to be retrained from scratch whenever a new sample arrives. It is therefore not surprising that there has been much interest in devising online techniques to resolve these problems. As a consequence, the SVM learning with online manner can offer significant computational advantages over batch learning algorithms and become more evident when dealing with streaming or very large-scale data. The rest of the paper is organized as follows. In section 2, a batch supervised SVM learning is introduced briefly. The extension of the batch SVM to its online version is also described in section 3. Experimental results and performance evaluations are reported in section 4. Finally, conclusions and our future work are drawn in section 5.

2 Batch Mode of SVM

The support vector machine is a supervised learning method that utilizes a set of labeled samples $(\mathbf{x}_k, y_k)_{k=1,\ldots,n}$ to design a classifier in a batch mode. $\mathbf{x}_k$ is the feature vector representing the instance and $y_k \in \{-1,1\}$ is the corresponding label of those $\mathbf{x}_k$. The SVM generates a decision function (or a hyper-plane) to design a classifier. This decision function is optimal, in that a maximal margin classifier can be obtained. The SVM solution is obtained by minimizing the primal objective function

$$\begin{aligned} &\min L_p = \frac{1}{2} \| \mathbf{w} \|^2 + C \sum_{i=1}^{n} \xi_i \\ &\text{subject to } \ y_i(\mathbf{w} \cdot \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, \ \ \xi_i \geq 0, \ \ \forall i \end{aligned} \tag{1}$$

where C is the regularization parameter. In Eq. (1), $\mathbf{w}$ is the normal vector of the hyperplane, b is the offset, $\phi(\mathbf{x}_i)$ is the mapping from input space to feature space, and ξ_i are the slack variables that permit the non-separable case by allowing misclassification of training instances. In practice, the convex quadratic programming problem in Eq. (1) can be solved by optimizing the following dual function

$$\begin{aligned} &\max \ L_d = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \\ &\text{subject to } \ \boldsymbol{\alpha}^{\mathrm{T}} \mathbf{y} = 0, \ \ 0 \leq \boldsymbol{\alpha} \leq C \end{aligned} \tag{2}$$

with Lagrange multipliers α_i. The kernel matrix $K(\mathbf{x}_i,\mathbf{x}_j)$ represents the dot products of $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ in feature space. After solving the QP problem, the norm of the hyperplane $\mathbf{w}$ can be represented as a linear combination of the vectors. Thus

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i \phi(\mathbf{x}_i) \tag{3}$$

If the model is properly trained, a soft margin SVM classifier that separate samples from different classes is given by

$$f(\mathbf{x}) = \operatorname{sgn}\left(\sum_{i=1}^{n} \alpha_i K(\mathbf{x}, \mathbf{x}_i) + b \right) \tag{4}$$

A widely popular methodology for solving the SVM QP problem is Sequential Minimal Optimization (SMO) [6]. Although SMO has been shown to be effective on large data sets for SVM learning, the algorithm requires high computational cost and a large amount of memory to hold the kernel products. These drawbacks always limit the applicability of the SVMs to many real-world problems. In the next section, an incremental/decremental online SVM learning is introduced to resolve aforementioned problems.

3 Online SVM

As opposed to the batch mode in which all samples are available at once, the online learning is a learning scenario in which training samples are provided one sample at a time. A practical advantage is that it allows the existing SVM model to incorporate additional data, without re-training from scratch. An important requirement for the online SVM learning is to maintain the KKT conditions during incremental/decremental procedures [7]. The incremental learning procedure refers to add one sample to the existing SVM model. When a new sample is considered, the KKT conditions are preserved on existing data and new sample for the enlarged data set. On the other hand, the decremental unlearning procedure refers to remove an existing sample from SVM model. Its weight is forced to zero while the weights of the remaining samples are updated such that the solution is optimal for the reduced data set. Meanwhile, elements of different categories may change their states. Hence, the learning and unlearning algorithms try to keep the least variation of the hyperplane. Since the decremental unlearning procedure is the inverse of incremental learning, we only consider the variation of the optimization problem in the case of incremental learning.

Recall that the dual formulation of the nonlinear SVM model in Eq. (1), the online SVM optimization problem is formulated by incorporating equality constraint directly into the dual objective function as

$$\max_{b} \min_{0\le\alpha_i\le C,\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{y}=0} L_d(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j H_{ij} - \sum_{i=1}^{n}\alpha_i + b\sum_{i=1}^{n}\alpha_i y_i \tag{5}$$

where $H_{ij}=y_iy_jK(\mathbf{x}_i,\mathbf{x}_j)$ and b is the bias in the decision function. The idea of the incremental SVM is that updates to the state of the sample $\mathbf{x}_i$ should keep the remaining samples in their optimal state. In other words, the KKT conditions

$$\begin{aligned} g_i &= \frac{\partial L_d(\boldsymbol{\alpha})}{\partial\alpha_i} = \sum_{j=1}^{n} H_{ij}\alpha_j + y_i b - 1 = y_i f(\mathbf{x}_i) - 1 \\ h &= \frac{\partial L_d(\boldsymbol{\alpha})}{\partial b} = \sum_{j=1}^{n}\alpha_j y_j = 0 \end{aligned} \tag{6}$$

must be maintained for all the examples, except possibly for the current one. Based on Eq. (6), any sample $\mathbf{x}_i$ can be partitioned into three different categories, e.g. Support set (S set), Error set (E set), and Reserve set (R set)[1].

The goal of incremental learning of updating solutions $\{\alpha,b\}$ is to find the increments $\{\Delta\alpha,\Delta b\}$ such that $\{\alpha+\Delta\alpha,b+\Delta b\}$ can satisfy the KKT conditions. In other words, the KKT conditions are preserved by changing the parameters in response to the perturbation induced by the addition of the new sample. The increments $\{\Delta\alpha,\Delta b\}$ can result in at least two situations. Firstly, the solution changes, but no sample changes its state among the different categories. In this case, only the coefficients of the support vectors and the bias term must be updated. Secondly, there exist one or more samples that migrate among S, E, and R sets. These changes can be controlled if $\Delta\alpha$ is chosen so that the minimum number of migrations occurs. When a new sample $\{\mathbf{x}_c,y_c\}$ is present, the new values of g_i and h in Eq. (6) are

$$\begin{aligned} g_i^{\text{new}} &= \sum\nolimits_j H_{ij}(\alpha_j+\Delta\alpha_j)+y_i(b+\Delta b)-1 \\ &= \underbrace{\sum\nolimits_j H_{ij}\alpha_j + y_i b - 1}_{g_i^{\text{old}}} + \underbrace{\sum\nolimits_j H_{ij}\Delta\alpha_j + y_i\Delta b}_{\Delta g_i}, \quad \forall i \in \chi \cup \{c\} \\ h^{\text{new}} &= \sum\nolimits_i (\alpha_i+\Delta\alpha_i)y_i = \underbrace{\sum\nolimits_i \alpha_i y_i}_{h^{\text{old}}} + \underbrace{\sum\nolimits_i \Delta\alpha_i y_i}_{\Delta h} \end{aligned} \tag{7}$$

Therefore, the changes of Δg_i and Δh are expressed as

$$\begin{aligned} \Delta g_i &= H_{ic}\Delta\alpha_c + \sum\nolimits_{j\in S} H_{ij}\Delta\alpha_j + y_i\Delta b \\ \Delta h &= \Delta\alpha_c y_c + \sum_{i=1}^{n}\Delta\alpha_i y_i \end{aligned} \tag{8}$$

where $\Delta\alpha_c$ is the coefficient being incremented of $\mathbf{x}_c$. To facilitate our analysis, Δg_i and Δh are expressed in term of the symmetric Hessian matrix $\mathbf{H}$. The Hessian matrix consists of several block matrices as shown in the following formulation

$$\mathbf{H} = \left[\begin{array}{ccc|c} \mathbf{H}_{SS} & \mathbf{H}_{SE} & \mathbf{H}_{SR} & \mathbf{H}_{Sc} \\ \mathbf{H}_{ES} & \mathbf{H}_{EE} & \mathbf{H}_{ER} & \mathbf{H}_{Ec} \\ \mathbf{H}_{RS} & \mathbf{H}_{RE} & \mathbf{H}_{RR} & \mathbf{H}_{Rc} \\ \hline \mathbf{H}_{Sc}^{\mathrm{T}} & \mathbf{H}_{Ec}^{\mathrm{T}} & \mathbf{H}_{Rc}^{\mathrm{T}} & H_{cc} \end{array}\right] \tag{9}$$

The columns and rows of the matrix are arranged in the orders of S, E, and R sets. For clarity, the new sample is appended in the last row and last column of the matrix. Thus, $\mathbf{H}_{Sc}$, $\mathbf{H}_{Rc}$, and $\mathbf{H}_{Ec}$ are column vectors with dimensions n_S, n_R, and n_E, where n_S, n_R, and n_E are the cardinality of three categories, respectively. $\mathbf{H}_{SS}$, $\mathbf{H}_{RS}$, and $\mathbf{H}_{ES}$ are

[1] S set: set of support vectors that lie on the margin (g_i=0 and $0<\alpha_i<C$). E set: set of error vectors that lie in the margin ($g_i<0$ and $\alpha_i=C$). R set : set of reserve vectors ($g_i>0$ and $\alpha_i=0$).

sub-matrix of **H** with dimensions $n_S \times n_S$, $n_R \times n_S$, and $n_E \times n_S$. From Eq. (9), Δg_i and Δh are rewritten as

$$\begin{aligned} \Delta g_i &= H_{ic}\Delta\alpha_c + \mathbf{H}_{Si}^{\mathrm{T}}\Delta\boldsymbol{\alpha}_S + y_i\Delta b \\ \Delta h &= \Delta\alpha_c y_c + \mathbf{y}_S^{\mathrm{T}}\Delta\boldsymbol{\alpha}_S = 0 \end{aligned} \tag{10}$$

In Eq. (10), $\mathbf{H}_{Si}$ is the ith column vector in the Hessian matrix **H** and $\Delta\boldsymbol{\alpha}_S$ contains the variation for the existing support vectors. Eq. (10) should be applied for existing data and the new sample. Therefore, before and after an update of $\Delta\alpha_c$, we obtain the following conditions in matrix-vector form which must be satisfied after an update

$$\begin{bmatrix} \Delta g_c \\ \Delta \mathbf{g}_S \\ \Delta \mathbf{g}_R \\ \Delta \mathbf{g}_E \\ 0 \end{bmatrix} = \begin{bmatrix} y_c & \mathbf{H}_{Sc}^{\mathrm{T}} \\ \mathbf{y}_S & \mathbf{H}_{SS} \\ \mathbf{y}_R & \mathbf{H}_{RS} \\ \mathbf{y}_E & \mathbf{H}_{ES} \\ 0 & \mathbf{y}_S^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \Delta b \\ \Delta\boldsymbol{\alpha}_S \end{bmatrix} + \Delta\alpha_c \begin{bmatrix} H_{cc} \\ \mathbf{H}_{Sc} \\ \mathbf{H}_{Rc} \\ \mathbf{H}_{Ec} \\ y_c \end{bmatrix} \tag{11}$$

The matrix-vector form in Eq. (11) is further reformulated into two parts. Firstly, it follows from condition in Eq. (7) that $\Delta \boldsymbol{g}_S$=**0** for support vectors. Therefore, rows 2 and 4 of Eq. (11) is rewritten as

$$\mathbf{Q}\Delta\mathbf{s} + \Delta\alpha_c\boldsymbol{\eta}_c = \mathbf{0} \tag{12}$$

where

$$\mathbf{Q} = \begin{bmatrix} 0 & \mathbf{y}_S^{\mathrm{T}} \\ \mathbf{y}_S & \mathbf{H}_{SS} \end{bmatrix},\ \Delta\mathbf{s} = \begin{bmatrix} \Delta b \\ \Delta\boldsymbol{\alpha}_S \end{bmatrix},\ \boldsymbol{\eta}_c = \begin{bmatrix} y_c \\ \mathbf{H}_{Sc} \end{bmatrix} \tag{13}$$

It is worth noting that the vector $\Delta\mathbf{s}$ contains the increments $\{\Delta\boldsymbol{\alpha}_s, \Delta b\}$ for the Lagrange multipliers α_s and b. As long as $\Delta\alpha_c$ is known, $\Delta\mathbf{s}$ can be solved by

$$\Delta\mathbf{s} = \Delta\alpha_c\boldsymbol{\beta} \tag{14}$$

and $\boldsymbol{\beta}=-\mathbf{Q}^{-1}\eta_c$. Secondly, rows 1 and 3 of Eq. (11) can also be reformulated as

$$\Delta\mathbf{g} = \Delta\alpha_c\boldsymbol{\gamma} \tag{15}$$

where

$$\Delta\mathbf{g} = \begin{bmatrix} \Delta g_c \\ \Delta \mathbf{g}_R \\ \Delta \mathbf{g}_E \end{bmatrix},\ \boldsymbol{\gamma} = \begin{bmatrix} y_c & \mathbf{H}_{Sc}^{\mathrm{T}} \\ \mathbf{y}_R & \mathbf{H}_{RS} \\ \mathbf{y}_E & \mathbf{H}_{ES} \end{bmatrix}\boldsymbol{\beta} + \begin{bmatrix} H_{cc} \\ \mathbf{H}_{Rc} \\ \mathbf{H}_{Ec} \end{bmatrix} \tag{16}$$

In Eqs. (14) and (15), the updates of $\Delta\mathbf{s}$ of samples in S set and $\Delta\mathbf{g}$ of samples in R set and E set are controlled by the sensitivity vectors $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ with respect to $\Delta\alpha_c$. Therefore,

$$\begin{bmatrix} b \\ \boldsymbol{\alpha}_S \end{bmatrix} \leftarrow \begin{bmatrix} b \\ \boldsymbol{\alpha}_S \end{bmatrix} + \Delta\alpha_c \boldsymbol{\beta}, \quad \text{for } S \text{ set}$$
$$\begin{bmatrix} g_c \\ \mathbf{g}_R \\ \mathbf{g}_E \end{bmatrix} \leftarrow \begin{bmatrix} g_c \\ \mathbf{g}_R \\ \mathbf{g}_E \end{bmatrix} + \Delta\alpha_c \boldsymbol{\gamma}, \quad \text{for } E \text{ and } R \text{ sets and new sample } \mathbf{x}_c \tag{17}$$

Finally, the complete flow chart of the incrementing procedure is depicted in Fig. 2. Detail descriptions of the incrementing procedure can be referred to [8].

4 Experimental Results

The simulation experiment is conducted for terrain image classification of synthetic aperture radar image. SAR is an active sensor, which illuminates targets with electromagnetic waves that can penetrate through cloud coverage. It owns all-weather and day and night imaging capabilities. Usually, SAR data can be used to synthesize

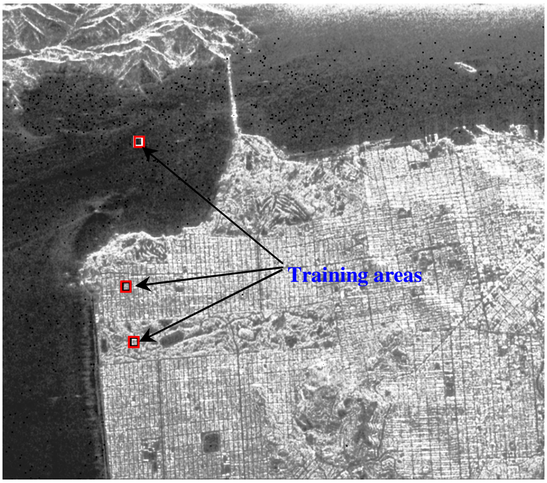


Fig. 1. SAR image of the San Francisco Bay area acquired by the NASA/JPL AIRSAR system. The red square boxes are labeled points for online SVM training.

Table 1. Classification matrix of SAR image

Class	Urban	Ocean	Park	Producer's accuracy
Urban	1408	20	93	92.57 %
Ocean	20	1486	15	97.70 %
Park	107	14	1400	92.04 %
User's accuracy	91.73 %	97.76 %	92.84 %	

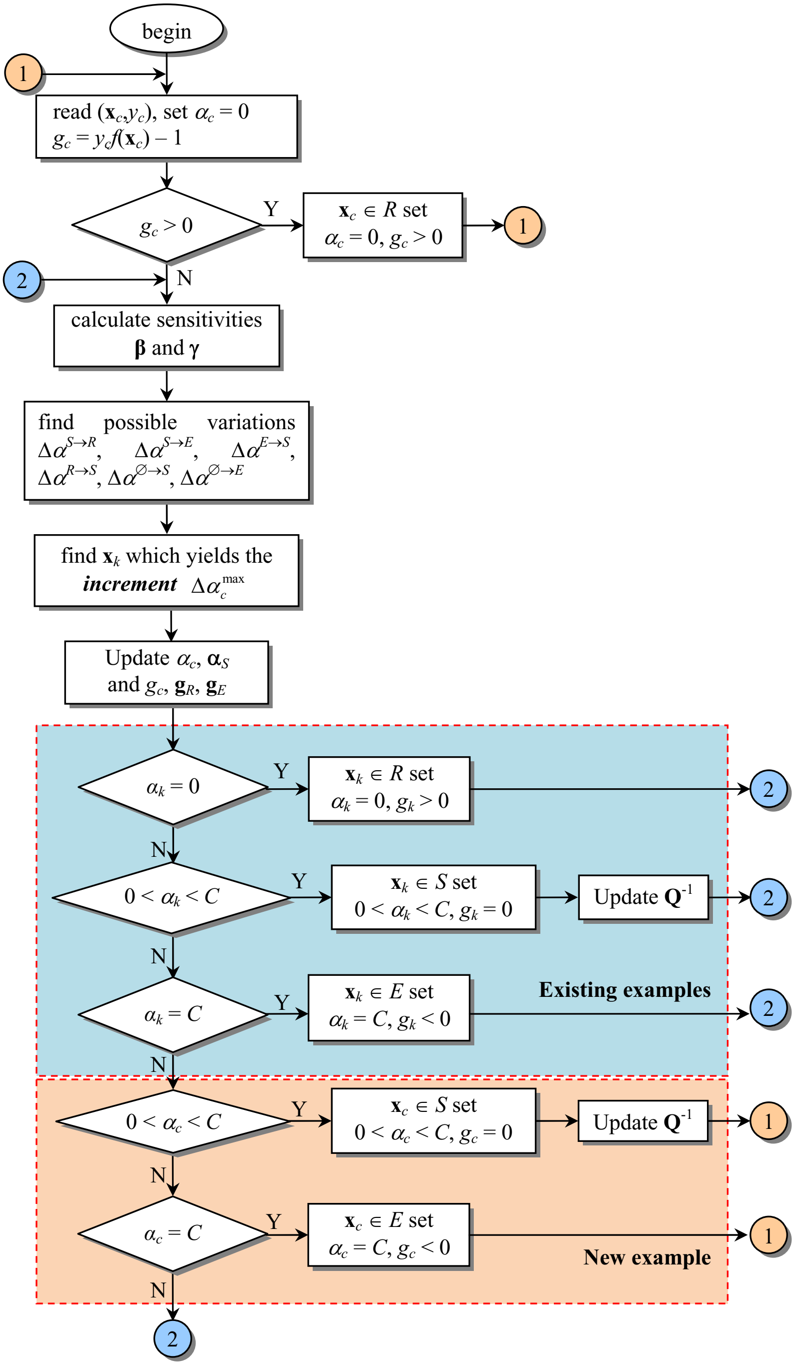

begin
1
read $(\mathbf{x}_c, y_c)$, set $\alpha_c = 0$
$g_c = y_c f(\mathbf{x}_c) - 1$
$g_c > 0$
Y
$\mathbf{x}_c \in R$ set
$\alpha_c = 0$, $g_c > 0$
1
2
N
calculate sensitivities
$\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$
find possible variations
$\Delta\alpha^{S\to R}$, $\Delta\alpha^{S\to E}$, $\Delta\alpha^{E\to S}$,
$\Delta\alpha^{R\to S}$, $\Delta\alpha^{\varnothing\to S}$, $\Delta\alpha^{\varnothing\to E}$
find $\mathbf{x}_k$ which yields the
increment $\Delta\alpha_c^{\max}$
Update α_c, $\boldsymbol{\alpha}_S$
and g_c, $\mathbf{g}_R$, $\mathbf{g}_E$
$\alpha_k = 0$
Y
$\mathbf{x}_k \in R$ set
$\alpha_k = 0$, $g_k > 0$
2
N
$0 < \alpha_k < C$
Y
$\mathbf{x}_k \in S$ set
$0 < \alpha_k < C$, $g_k = 0$
Update $\mathbf{Q}^{-1}$
2
N
$\alpha_k = C$
Y
$\mathbf{x}_k \in E$ set
$\alpha_k = C$, $g_k < 0$
Existing examples
2
N
$0 < \alpha_c < C$
Y
$\mathbf{x}_c \in S$ set
$0 < \alpha_c < C$, $g_c = 0$
Update $\mathbf{Q}^{-1}$
1
N
$\alpha_c = C$
Y
$\mathbf{x}_c \in E$ set
$\alpha_c = C$, $g_c < 0$
1
New example
N
2

Fig. 2. The incrementing procedure of the online SVM

responses from any combination of transmitting and receiving polarizations. Polarimetric SAR measures a target's reflectivity with quad-polarizations, horizontal transmitting and receiving (HH), horizontal transmitting and vertical receiving (HV), vertical transmitting and horizontal receiving (VH), and vertical transmitting and vertical receiving (VV)[2]. This capability provides information to characterize scattering mechanisms of various terrain covers. The test site selected in our study is a 4-look L-band fully polarimetric SAR image over the San Francisco Bay area acquired by the NASA/JPL AIRSAR system. The size of the image is 1024 lines x 900 pixels. Fig. 1 shows the original image.

The most obvious characteristic and representative of the class pixels are selected as training sample in the image. Based on visibility, a total of three classes are identified, i.e. ocean (open water), park area, and urban city. Both copolarized and cross-polarized signatures, HH, VV, and HV polarizations, were extracted for SVM training. In this simulation, approximately 1,500 pixels (for training samples) of each class were selected with a total of 9,000 pixels (approximately 7,500 pixels for checking results) where ground truths are available. The training areas are enclosed with red boxes as shown in Fig. 1. To classify the image, one-against-one strategy is adopted in the study mainly because many different classes are required to be treated as one class.

As can be seen from Table 1, three types of covers are clearly distinguished. The classification accuracy is slightly lower in urban regions, when compared with ocean and parks. These results are in agreement with our observations and are mainly due to the complex mixtures of streets, vegetation, and buildings. It is worth while to note that in the regions of urban areas and parks, where the radar returns shown on the image is comparable and the classification rates are similar.

5 Conclusions

Synthetic aperture radar image is an artificial form of real data which provides a useful source to improve understanding the complexity of reality. It can also be employed to emphasize the desirable information. In the terrain image classification, only a small portion of objects or ground truths is available. Furthermore, data is often collected continuously in time and may also evolve in time. In the former case, the performance of the SVM learning can only be sub-optimal. In the latter case, the SVM model needs to be retrained from scratch whenever a new sample arrives. In this paper, an online SVM learning is thus proposed and applied for classifying SAR image of the San Francisco Bay area. Experimental results show that the online SVM learning can achieve higher accuracy of classification. For future work, we plan to combine transductive SVM with online learning whenever labeled samples are scarce and expensive to generate while unlabeled samples are often readily available.

[2] This is achieved by alternately transmitting horizontal (H) and vertical (V) polarization radar pulses and receiving both H and V polarizations of reflected pulses with sufficiently high pulse repetition frequencies.

Acknowledgments. This research was supported by the National Science Council under contract numbers NSC99-2221-E-212-023 and NSC100-2221-E-212-006.

References

1. Noble, W.S.: Support vector machine applications in computational biology. In: Schoelkopf, B., Tsuda, K., Vert, J.-P. (eds.) Kernel Methods in Computational Biology, pp. 71–92. MIT Press (2004)
2. Lal, T.N., Schröder, M., Hinterberger, M.T., Weston, J., Bogdan, M., Birbaumer, N., Schölkopf, B.: Support vector channel selection in BCI. IEEE Trans. Biomedical Engineering 51(6), 1003–1010 (2004)
3. Trafalis, T.B., Ince, H.: Support vector machine for regression and applications to financial forecasting. In: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, Como, Italy, pp. 348–353 (2000)
4. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, Berlin (1995)
5. Delbos, F., Gilbert, J.C.: Global linear convergence of an augmented Lagrangian algorithm for solving convex quadratic optimization problems. Journal of Convex Analysis 12, 45–69 (2005)
6. Platt, J.C.: Fast Training of Support Vector Machines Using Sequential Minimal Optimization. In: Advances in Kernel Methods: Support Vector Learning, pp. 185–208. MIT Press (1999)
7. Cauwenberghs, G., Poggio, T.: Incremental and Decremental Support Vector Machine Learning. In: Leen, T.K., Dietterich, T.G., Tresp, V. (eds.) Advances in Neural Information Processing Systems, vol. 13, pp. 409–415. MIT Press (2001)
8. Nguyen, T.H.: Online Transductive Support Vector Machine. Mater thesis, Da-Yeh University (August 2008)

A Novel Algorithm for Kernel Optimization of Support Vector Machine

Lijie Li

NingBo City College of Vocational Technology, Xuefu Road,
Yinzhou Higher Educational Zone, NingBo, ZheJiang, P.R. China 315100
llj@graduate.shu.edu.cn

Abstract. Model optimization namely the kernel function and parameter selection is an important factor to affect the generalization ability of support vector machine (SVM). To solve model optimization problem of support vector machine classifier, a novel algorithm (GC-ABC) is proposed which integrate artificial bee colony algorithm, greedy algorithm and chaos search strategy. The simulation results show that the accuracy of SVM optimized by GC-ABC is superior to the SVM optimized by genetic algorithm and ant colony algorithm. The experiments further suggest that GC-ABC algorithm has fast convergence and strong global search ability, which improves the performance of the support vector machine.

Keywords: Kernel Optimization, Support Vector Machine, Artificial Bee Colony, Chaos Search.

1 Introduction

Support vector machine is a statistical machine learning method which exhibit many unique advantages of nonlinear and high dimensional pattern recognition, and is widely applied to the fitting function, text and image recognition problems. The support vector machine method is to establish a learning theory of VC dimension theory and structural risk minimization principle on the basis of statistical, according to the information of limited samples to find the best compromise between the complexity of model selection and learning ability [1]. A large number of studies show that, the relationship between performance and the kernel function, parameter and penalty factor of support vector machine closely. So how to choose the appropriate model and its parameters (namely model selection) to get the optimal classification results has been a hotspot in the research of support vector machine. The related research work about on the model optimization focused on cross validation method, orthogonal method, gradient descent method, ant colony optimization[2], momentum particle swarm optimization[3] and genetic algorithm[4][5].

In this paper, a novel effective algorithm (GC-ABC) is proposed based on artificial bee algorithm, greedy algorithm and chaos search strategy to solve the model optimization of SVM. The experimental results show that the GC-ABC algorithm can

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 98–105, 2013.

achieve better searching for the optimal model in the evolution as to improve the classification performance of support vector machine. The paper will be organized as follows. Firstly, it introduces the support vector machine and artificial bee colony algorithm. Then, it will be a detailed description of the model of effective selection mechanism. The experimental results will be described in the end.

2 Support Vector Machine and Artificial Bee Colony

Support vector machines are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. The basic SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the output, making it a non-probabilistic binary linear classifier. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. In addition to performing linear classification, SVM can efficiently perform non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. Given training data set $\{x_i, y_i\}_{i=1}^{m}$, x_i input vectors, $y_i \in \{-1,1\}$ classify labels. SVM aims to find the optimal hyper plane which makes the margin maximum (See Figure 1.)

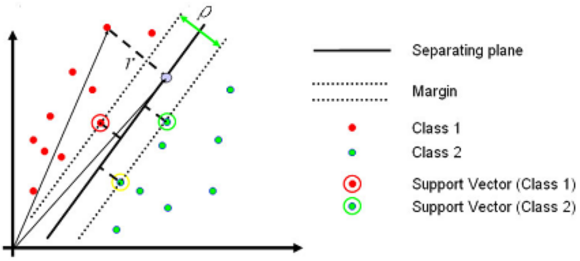


Fig. 1. Support Vector Machine

The Slack variable ε_i , $i = 1, 2, ..., m$ is introduced which measure the degree of misclassification of the data when the training data is inseparable. So the optimal problem of the classification hyper plane transfer to solve the object function (1).

$$Min(r,w,b)=\frac{1}{2}\|w\|^2+C\sum_{i=1}^{m}\varepsilon_i$$
$$s.t.\quad y_i[(w.x_i)+b]\geq 1-\varepsilon_i$$
$$\sum_{i=1}^{m}y_i\alpha_i=0 \tag{1}$$
$$0\leq\alpha_i\leq C,\varepsilon_i\geq 0,i=1,2,...,m$$

Here, variable C is called error penalty parameter or soft margin parameter which is great than zero. By using the kernel trick $K(x_i,x_j)=\phi(x_i).\phi(x_j)$, the objective function (1) is converted to function (2).

$$Q(\alpha)=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j K(x_i.x_j) \tag{2}$$

Accordingly, the corresponding classification decision function is represented as function (3).

$$f(x)=\text{sgn}((w^*.x)+b^*) \tag{3}$$

其中， $w^*=\sum_{i=1}^{m}y_i\alpha_i^* x_i$ ， $b^*=y_j-\sum_{i=1}^{m}y_i\alpha_i^*(x_i.x_j),j\in\{j\,|\,0<\alpha_j^*<C\}$ 。

So, the effectiveness of SVM depends largely on the selection of kernel, the kernel's parameters, and soft margin parameter.

The artificial bee colony algorithm (ABC) is an optimization algorithm based on the intelligent foraging behavior of honey bee swarm, proposed by Karaboga in 2005[6]. In the ABC model, the colony is divided into three groups of bees: employed bees, onlookers and scouts. It is assumed that there is only one artificial employed bee for each food source. Employed bees go to their food source and come back to hive and dance on this area. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source. Onlookers watch the dances of employed bees and choose food sources depending on dances. In ABC, a population based algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. At the first step, a randomly distributed initial population (food source positions) is generated. After initialization, the population is subjected to repeat the cycles of the search processes of the employed, onlooker, and scout bees, respectively. An employed bee produces a modification on the source position in her memory and discovers a new food source position. Provided that the nectar amount of the new one is higher than that of the previous source, the bee memorizes the new source position and forgets the old one. Otherwise she keeps the position of the one in her memory. After all employed bees complete the search process; they share the position information of the sources with the onlookers on the dance area. Each onlooker evaluates the nectar information taken from all employed bees and then chooses a food source depending on the nectar amounts of sources. As in the case of the employed bee, she produces a modification on the source position in her memory and checks its nectar amount. Providing that its nectar is higher than that of the previous one, the bee memorizes the new position and forgets the old one. The sources abandoned are determined and new sources are randomly produced to be replaced with the abandoned ones by artificial scouts. ABC algorithm has applied to many real world problems, such as unconstrained numerical optimization problems, its extended version for the constrained optimization problems, optimal multi-level threshold[7], MR brain image classification[8], cluster analysis[9], face pose estimation, and 2D protein folding.

3 The Framework of GC-ABC

To improve the performance of artificial bee colony algorithm, the paper proposes a novel effective algorithm which adopts greedy algorithm and chaos search strategy to balance enhance convergence speed and global search ability. The framework of GC-ABC is described as follows.

Step 1: initialization. Generated initial solution randomly and half of them is employed bees.
Step 2: Divide the dataset into ten sub datasets, one sub dataset for training and the rest sub datasets for test.
Step 3: Calculate the fitness value of employed bees on training sub dataset, sort according to the fitness value and then save the best fitness value (bst).
Step 4: On-looks chooses the food resource according the information the employed bee shares and search the new solution locally using greedy strategy.
Step 5: Update the food resource. If the new solution is inundated in ten times continuously, then use chaos search strategy to jump the local convergence.
Step 6: Update the best fitness value. Compare the current best fitness value of current iteration with bst and update the value of bst . The stop criteria meets, then stop the procedure, otherwise go to step 3.

To avoid the local optimal caused by greedy algorithm and make the balance between the local optimal and global optimal, the chaos search strategy is adopted. Chaos sequence will not repeat pass through all the states in a specific region by characteristics of periodicity. The chaotic sequence can help to generate the near neighbor of locally optimal in the iteration to improve the performance of GC-ABC and help to find the optimal solution quickly. The core of chaos search is to generate the chaos sequence firstly and then map chaotic variables to the value range of variables by carrier. The chaos sequence is generated by logistic function (4).

$$y_{(n+1),d} = uy_{n,d}(1 - y_{n,d}) \tag{4}$$

The variable $n \in [1, N_{\max}], d \in [1, D]$ and u is the variable to control the chaos states and equals to 4 in this paper. Further, the chaotic variables are first amplified by carrier operation and then loaded on the individual variables to search. New individual will changed by function (5) after chaos operation.

$$\omega_{n,d} = f_{i,d} + R_{i,d}(2y_{n,d} - 1) \tag{5}$$

According the basic idea of function (5), the iteration variable $w_{n,d}$ is mapped to the area with the radius of $R_{i,d}$ and center in the $f_{i,d}$. The chaos search can be described as follows.

Step 4.1: generate the chaos sequence by function (4), and map the chaos sequences to optimal space.

Step 4.2: calculate the fitness of $w_{n,d}$

Step 4.3: if the stop criteria meet, then stop the process, otherwise jump to step 4.

Step 4.4: replace the food resource of employed bee with new value.

4 Experiments Results

In order to validate the generalization performance of GC-ABC, we compare it with ACO-SVM, PSO-SVM, GA-SVM on various benchmark data sets in UCI. ACO-SVM uses the ant colony algorithm to choose the optimal model of SVM, and PSO-SVM, GA-SVM adopts particle swarm optimization algorithm, genetic algorithm correspondingly.

Table 1. Accuracy comparison by 4 algorithms on different datasets

DataSet	Kernel function	GC-ABC	ACO-SVM	PSO-SVM	GA-SVM
Shuttle	RBF	85.6	80.1	84.2	69.2
Poker Hand	RBF	78.5	57.4	50.8	54.2
Yeast	RBF	93.3	93.7	91.3	92.5
Adult	RBF	94.5	90.4	91.6	92.8

According to the table 1, we can see that the SVM's performance which is optimized by GC-ABC is improved dramatically. The figure 2 shows that with the use of chaos search, the GC-ABC enhances the global search ability to find the optimal model which improves the accuracy of the SVM. The figure 3 suggests that with out the greedy algorithm, the artificial bee colony algorithm converges slower. To further demo the power of GC-ABC, the more experiments were conducted between GC-ABC, GS-SVM, HM-SVM and KGP on more various datasets and the results were listed from figure 4-7.

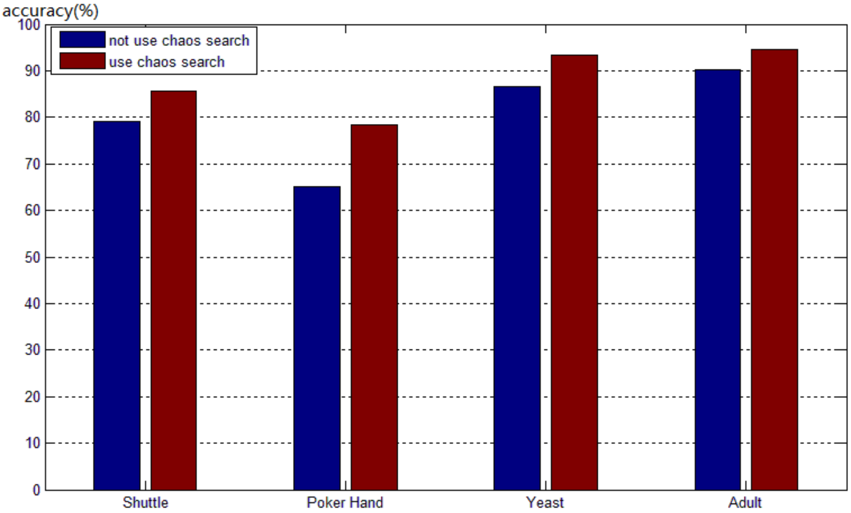


Fig. 2. The effective of chaos search

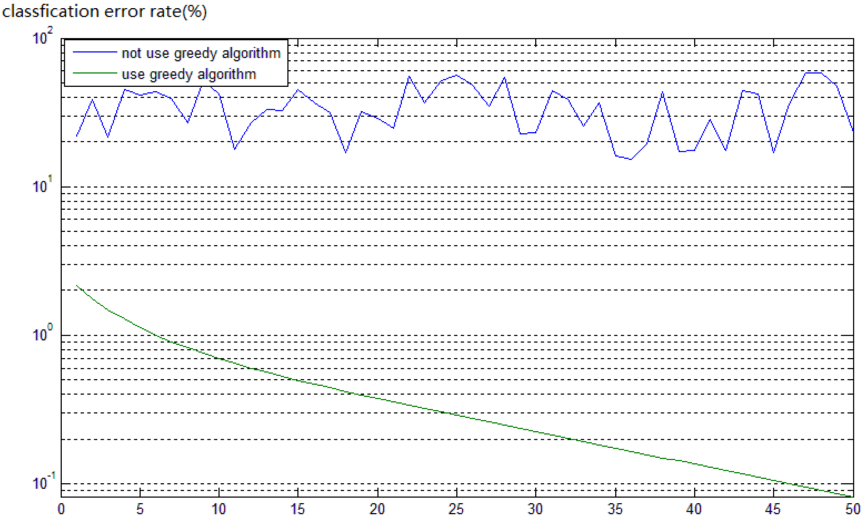


Fig. 3. The effective of greedy algorithm

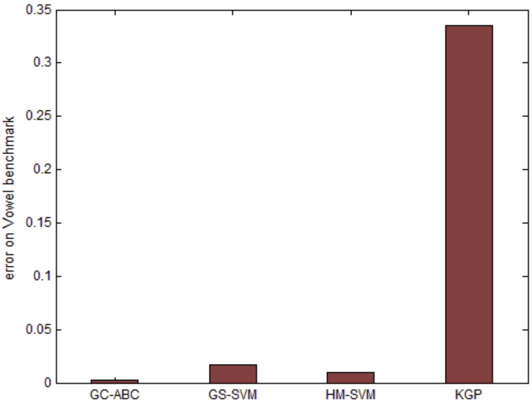


Fig. 4. Errors on Vowel benchmark

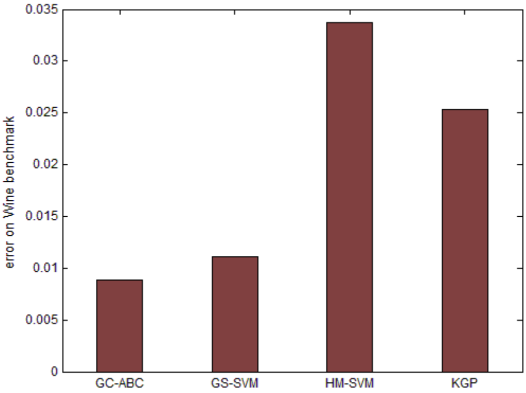


Fig. 5. Errors on Wine benchmark

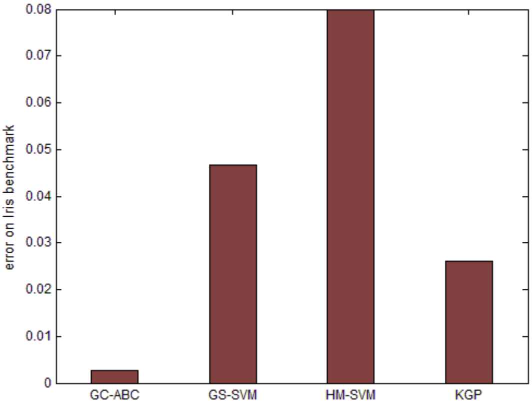


Fig. 6. Errors on Iris benchmark

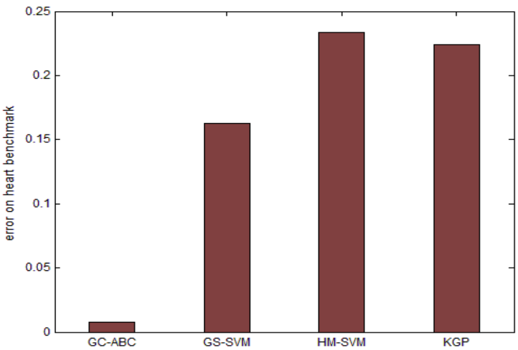


Fig. 7. Errors on Heart benchmark

5 Conclusion

This paper proposes a new novel algorithm that optimizes the model of support vector machine using chaos search strategy and greedy algorithm. The experiments were conducted between GC-ABC algorithm between ACO-SVM, PSO-SVM, GA-SVM and others on benchmark dataset of UCI. The proposed algorithm can make a well balance between global optimization and local optimization and improve the performance of SVM.

References

1. Sch-kopf, B., Smola, A.: Learning with kernels: support vector machines regulation, optimization and beyond. MIT, London (2001)
2. Yu, M., Ai, Y.-Q.: SVM parameter optimization and application based on artificial bee colony algorithm. Journal of Optoelectronics Laser 23, 374–378 (2012)
3. Wang, J., Xu, W.-H.: Parameter Optimization of mixed kernel SVM based on momentum partical swarm optimization. Journal of Computer Application 31, 501–504 (2011)
4. Li, L., Pan, F.: Parameter Optimization Algorithm for Support Vector Machine Based on the Non-dominated Sorting Genetic Algorithm II. Control Theory and Application 27, 1–4 (2008)
5. Ma, Y.-L., Pei, S.-L.: Study on Parameters Optimizaiton of Support Vector Machines Base on Improved Genetic Algorithm. Computer Simulation 27, 150–154 (2010)
6. Karaboga, D., Akay, B.: A comparative study of Artifical Bee Colony Algorithm. Applied Mathematics and Computation 214, 108–132 (2009)
7. Zhang, Y., Wu, L.: Optimal multi-level Thresholding based on Maximum Tsallis Entropy via an Artificial Bee Colony Approach. Entropy 13, 841–859 (2011)
8. Zhang, Y., Wu, L., Wang, S.: Magnetic Resonance Brain Image Classification by an Improved Artificial Bee Colony Algorithm. Progress in Electromagnetics Research 116, 65–79 (2011)
9. Zhang, Y., Wu, L., Wang, S., Huo, Y.: Chaotic Artificial Bee Colony used for Cluster Analysis. Communications in Computer and Information Science 13, 205–211 (2011)

Training Least-Square SVM by a Recurrent Neural Network Based on Fuzzy c-mean Approach*

Fengqiu Liu[1], Jianmin Wang[1], and Sitian Qin[2]

[1] Harbin University of Science and Technology,
[2] Harbin Institute of Technology, Harbin, 150080, China
liufengqiuhit@126.com

Abstract. An algorithm to solve the least square support vector machine (LSSVM) is presented. The underlying optimization problem for LSSVM follows a system of linear equations. The proposed algorithm incorporates a fuzzy c-mean (FCM) clustering approach and the application of a recurrent neural network (RNN) to solve the system of linear equations. First, a reduced training set is obtained by the FCM clustering approach and used to train LSSVM. Then a gradient system with discontinuous righthand side, interpreted as an RNN, is designed by using the corresponding system of linear equations. The fusion of FCM clustering approach and RNN overcomes the loss of spareness of LSSVM. The efficiency of the algorithm is empirically shown on a benchmark data set generated from the University of California at Irvine (UCI) machine learning database.

Keywords: least square support vector machine, neural network, fuzzy c-mean clustering.

1 Introduction

Least square support vector machine (LSSVM) is originally proposed by Suykens in Ref. [1] through replacing the inequality constraints of the optimization problem in support vector machine (SVM)[2] with equality constraints. The underlying optimization problem for LSSVM follows a system of linear equations. This modification simplifies the optimization problem and improves the training efficiency for large-scale learning tasks. But an obvious shortcoming of LSSVM is the loss of sparseness. So many pruning techniques to obtain sparse LSSVM are proposed [3,4]. In Ref. [3], sparse LSSVM classifiers are obtained by removing the training points with smaller absolute support values. In Ref. [4], a reduced training set is selected by introducing the smallest approximation error. Both pruning algorithms in [3,4] need train an LSSVM based on the original training sets. That is to say, the system of linear equations with large-scale need to

* This work was supported by Fund project of Heilongjiang Province Education Office (Grant No.1251112).

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 106–113, 2013.

be solved before the reduced training sets are obtained. In fact, many classical methods are capable of solving the linear equations, such as Gaussian elimination methods and conjugate gradient methods [5,6]. Besides, a gradient system is presented to solve an underdetermined system of linear equations. This gradient system can be interpreted as a recurrent neural network (RNN) and applied to solve LSSVM. Thus, an advantage of this system is that it can be solved using standard ODE software. Moreover, the solution of linear equations is convergent in finite time. However, the direct applications to LSSVM of the above mentioned methods can not overcome the loss of spareness.

This study aims at a simple and feasible algorithm to solve LSSVM. The proposed algorithm combines the FCM clustering approach and the application of an RNN to solve the linear equations. The origin training sets are reduced through FCM clustering approach. This procedure avoids to solve large-scale linear equations. Moreover, the efficiency of learning is improved using a RNN to solve the linear equations.

This paper is organized as follows. Section 2 briefly reviews LSSVM. An algorithm to solve LSSVM is proposed based on FCM clustering approach and in Section 3 Experiments are performed in Section 4. Conclusions are given in Section 5.

2 Least Support Vector Machine

This section recalls the basic ideas of LSSVM. For further detail on LSSVM we can refer to [3,8,9]. In LSSVM one aims at minimizing the empirical risk with the equality constraints:

$$\begin{aligned} &\min_{w,b,e} J_P(w,e) = \frac{1}{2}w^T w + \frac{1}{2}\gamma \sum_{k=1}^{m} e_k^2 \\ &s.t.\;\; y_k[w^T\phi(x_k)+b] = 1 - e_k, \;\; k = 1,2,\cdots,m. \end{aligned} \tag{1}$$

For a classifier in the primal spaces that takes the form

$$y(x) = \mathrm{sign}[w^T\phi(x_k)+b], \tag{2}$$

where $\phi(\cdot) : R^n \to R^n$ is the mapping to the high dimensional feature space as in the standard SVM case. The Lagrange for the problem is

$$L(w,b,e;\alpha) = J_P(w,e) - \sum_{k=1}^{N} \alpha_k\{y_k[w^T\phi(x_k)+b] - 1 + e_k\}, \tag{3}$$

where the α_k is the Lagrange multipliers. The conditions for optimality yield

$$\begin{aligned}
\frac{\partial L}{\partial w} &= 0, \qquad w = \sum_{k=1}^{m} \alpha_k y_k \phi(x_k), \\
\frac{\partial L}{\partial b} &= 0, \qquad \sum_{k=1}^{m} \alpha_k y_k = 0, \\
\frac{\partial L}{\partial e_k} &= 0, \qquad \alpha_k = \gamma e_k, \\
\frac{\partial L}{\partial \alpha_k} &= 0, \qquad y_k[w^T \phi(x_k) + b] - 1 + e_k = 0.
\end{aligned} \tag{4}$$

Defining $z = (\phi(x_1^T)y_1, \cdots, \phi(x_m^T)y_m)^T, y = (y_1, y_2, \cdots, y_m), \overrightarrow{1} = (1, 1, \cdots, 1)^T$, $e = (e_1, e_2, \cdots, e_m)$, $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_m)$, and eliminating w, e, one obtains the following linear Karush-Kuhn-Tucker (KKT) system [2].

$$\begin{aligned}
&\text{Solve} \quad \alpha,\ b \\
&\text{s. t.} \ \ A\alpha = b.
\end{aligned} \tag{5}$$

Here, $A = [1, y^T; y^T, H + \frac{I}{\gamma}]$. $H = (h_{k,l})_{k,l=1}^{m}$ is given by

$$h_{k,l} = y_k y_l \phi(x_k)\phi(x_l) = y_k y_l K(x_k, x_l), k, l = 1, \cdots, m. \tag{6}$$

The classifier in the dual space takes the form

$$y(x) = \text{sign}[\sum_{k=1}^{m} \alpha_k y_k K(x, x_k) + b] \tag{7}$$

3 Solve LSSVM Based on FCM Clustering Approach and Recurrent Neural Network

In the proposed algorithm, the FCM clustering approach is used to reduce the original training sets. First, the original training data are split into subsets using the FCM clustering approach [7]. A reduced training data set is obtained through deleting the training data with larger membership degree in the subsets. We formulate the LSSVM using the reduced training set. Finally, we design an RNN to solve the corresponding optimization problem.

Now the proposed algorithm is described by four steps as follows.

Step 1. Use the FCM clustering approach to split training data set $\mathcal{T}$ into k subsets. Besides, the largest membership degree $u(x_i)$ of x_i is given for the k subsets.

Step 2. Set the parameter $\beta > 0$. Denote $T_l = \{x_i | u(x_i) \geq \beta, \ \ i \in \{1, 2, \cdots, m\}\}$, where $l = 1, 2, \cdots, k$. That is to say, we obtain the reduced training set $T_r = \cup_{l=1}^{k} T_l$.

Step 3. Construct an RNN according to the system of linear equations (5).

LSSVM transforms into the problem of solving system of algebraic linear equations (5). By the results in Ref. [5], the deviation approach to solving the system (5) is to solve the following unconstrained optimization problem:

$$\text{Minimize } ||Ax - b||_1 \tag{8}$$

where $|| \cdot ||_1$ denotes the L_1 norm of the argument. The optimization problem (8) is solved by an RNN of the form

$$\dot{x} = -MA^T \text{sgn}(x) \tag{9}$$

Here $M = \text{diag}(\mu_1, \mu_2, \cdots, \mu_m),\ \mu_i > 0$ for all i, is a positive diagonal matrix, which is referred to as a learning matrix; sgn(x), for a vector $x = (x^1, x^2, \cdots, x^m)$, is defined as

$$(\text{sgn}(x^1), \text{sgn}(x^2), \cdots, \text{sgn}(x^m))$$

and

$$\text{sgn}(x^i) = \begin{cases} 1, & \text{if } x^i > 0; \\ [-1, 1], & \text{if } x^i = 0; \\ -1, & \text{if } x^i < 0. \end{cases}$$

Step 4. Apply cross-validation on the training data to choose optimal hyper-parameters. Solve the linear equations (5) by RNN (9) for the optimal choice of the parameters.

4 A Benchmark Study– Seeds Data Set

For our benchmark study, the seeds data set originates from the University of California at Irvine (UCI) machine learning database.

4.1 Seeds Data Set

Seeds data set includes seven input attributions: area, perimeter, compactness, length of kernel, width of kernel, asymmetry coefficient, which are denoted as $x^i,\ i = 1, 2, \cdots, 7$. There are $m = 210$ data in the set. That is, $\mathcal{T} = \{(x_i, y_i)|i = 1, 2, \cdots, m\}$. The output values $y_i \in \{1, 2, 3\}$. Fig. 1 depicts the data set.

4.2 Simulation Procedure

In this experiment, we use 10-folds cross-validation to evaluate the performance of the proposed method. Parameter selection is performed by training LSSVM on 9/10 of Seeds data set. The other 1/10 of that is used in the test procedure. According to **Step1**, the training data set is split into three classes by using FCM clustering approach. Fig. 2 depicts one of the clustering results and the reduced training sets for every β, respectively.

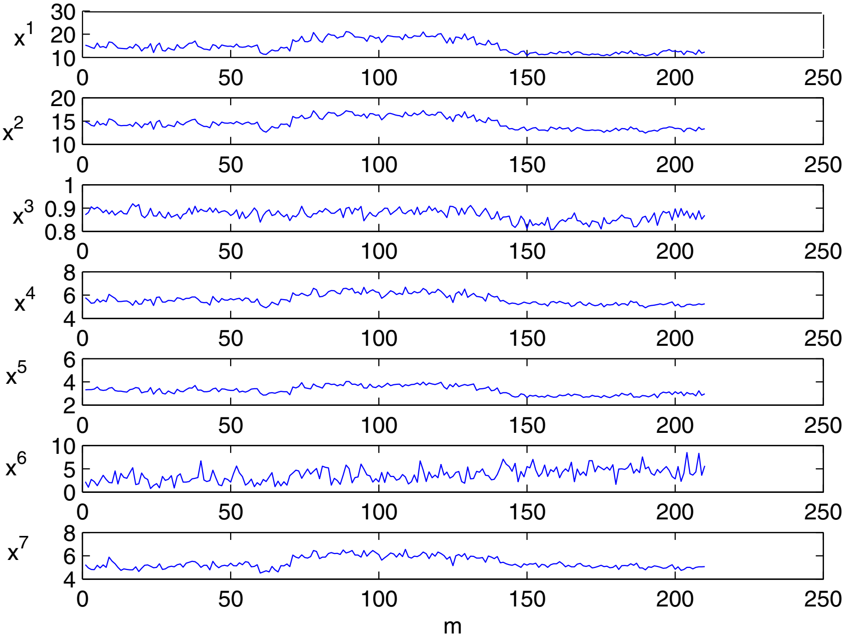


Fig. 1. Seeds data set

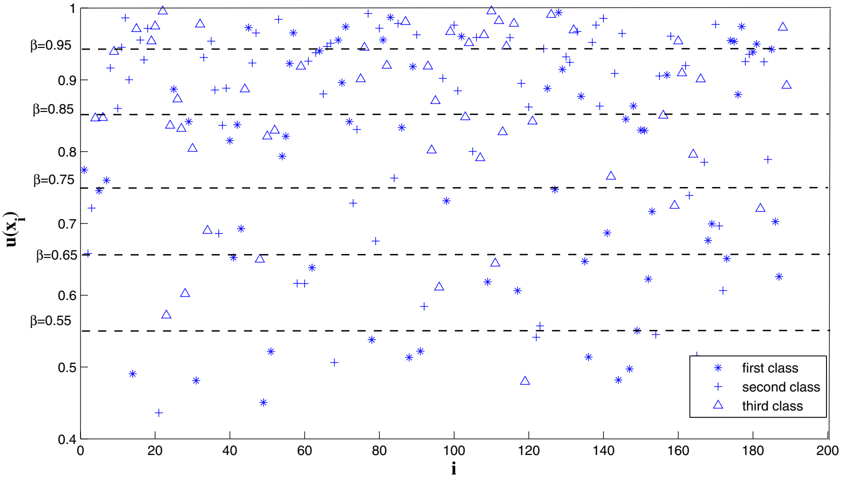


Fig. 2. One of clustering results for training data sets by FCM clustering approach

Table 1. Numerical Results for Seeds Data Set

Methods	β	The number of training data	Tes-er	SV
Proposed method	0.55	17	8	2
	0.65	31	11	5
	0.75	48	10	4
	0.85	80	6	6
	0.95	83	6	8
SVM	$--$	189	7	5

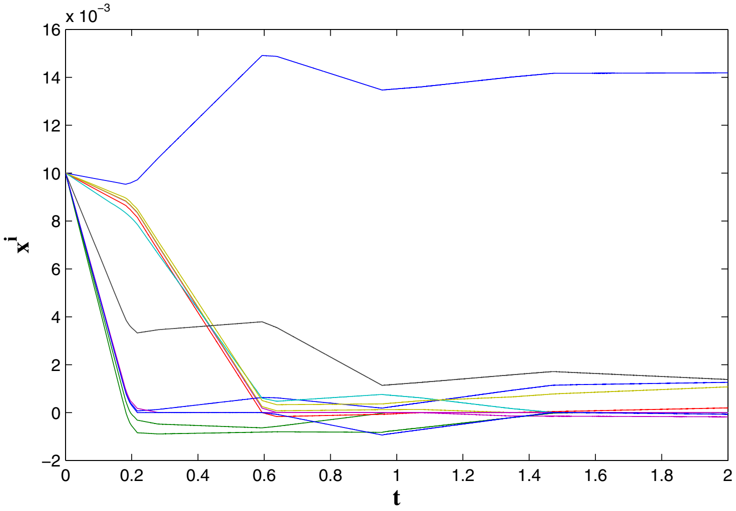


Fig. 3. Some trajectories of the recurrent neural network for $\beta = 0.55$

The following terms are used in the experiments.

- **Kernels:** We choose a positive definite kernel, i.e. Gaussian kernel

$$k^{rbf} = \exp(-\frac{||x - x_i||^2}{2\delta^2})$$

  to perform our experiments.
- **Parameters:** γ stands for the regularization factor. δ represent the parameters in k^{rbf}. The parameters are chosen using a grid search method, where $\gamma \in \{2^{-1}, 2^0, \cdots, 2^{25}\}$ and $\delta \in \{2^{-1}, \cdots, 2^2\}$. Additionally, the parameter $\beta = 0.55,\ 0.65,\ 0.75,\ 0.85,\ 0.95$ in **Step 2**, respectively.
- **Performance indices (PIs):** We select SVM to compare with the proposed algorithm. The mean number of support vectors (SVs), the test errors

(Tes-er) are used as PIs. Here, Tes-er equal the mean number of misclassified samples in the test procedures.

4.3 Simulation Results

We summarize the numerical results in Table 1.

Some trajectories of the recurrent neural network (9) are depicted in Fig. 3 for $\beta = 0.55$ and in Fig. 4 for $\beta = 0.75$, which shows finite time convergence to the solution of the system of linear equations (5).

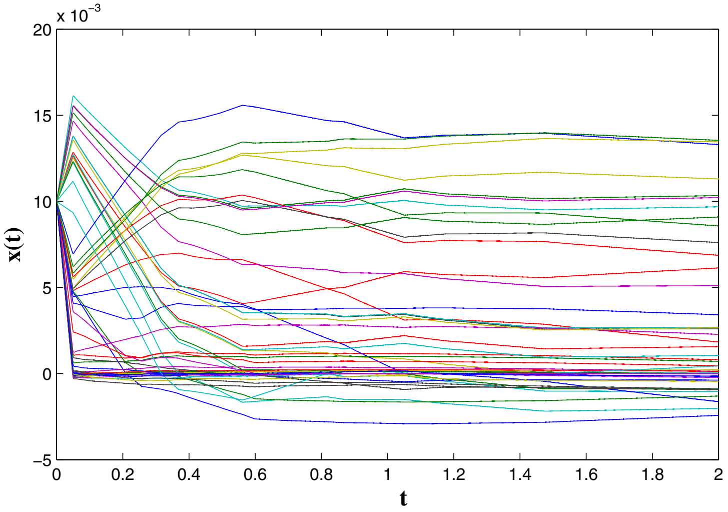


Fig. 4. Some trajectories of the recurrent neural network for $\beta = 0.75$

From the results in Table 1, we see that Tst-er of the proposed algorithm are basically same as those of SVM. Additionally, the number of training data for the proposed algorithm is less than those for SMV when the values of Tes-er and SV for these two methods equal almost.

5 Conclusions

An training algorithm for LSSVM is presented in this paper. The key idea of the proposed algorithm is the use of FCM clustering approach and the application of recurrent neural networks in solving optimization problem of LSSVM. In particular, the recurrent neural netowrk can be solved using both standard ODE software and implemented by means of integrated circuits [5]. The solution of the system of linear equations is convergent in finite time. The obtained results show that the incorporation of the two methods enhances the training and test efficiency.

References

1. Suykens, J.A.K., Vandewalle, J.: Least squares support vector machine classifiers. Neural Processing Letters 9(3), 293–300 (1999)
2. Schölkopf, B., Smola, A.J.: Learning With Kernels. MIT Press, Cambridge (2002)
3. Suykens, J.A.K., Lukas, L., Vandewalle, J.: Sparse least squares support vector machine classifiers. In: roceedings of European Symposium of Artificial Neural Networks 2000, Bruges, Belgium, pp. 37–42 (April 2000)
4. De Kruif, B.J., De Vries, T.J.A.: Pruning error minimization in least squares support vector machines. IEEE Transactions on Neural Networks 14(3), 696–702 (2003)
5. Ferreira, L.V., Kaszkurewicz, E., Bhaya, A.: Solving systems of linear equations via gradient systems with discontinuous righthand sides: application to LS-SVM. IEEE Transactions on Neural Networks 16(2), 501–505 (2005)
6. Xue, X., Bian, W.: Subgradient-based neural networks for nonsmooth convex optimization problems. IEEE Transactions on Neural Networks Circuits and Systems I: Regular Papers 55(8), 2378–2391 (2008)
7. Lin, C.T., George Lee, C.S.: Neural Fuzzy Systems. Prentice-Hall, Englewood Cliffs (1996)
8. Suykens, J.A.K., De Brabanter, J., Lukas, L., et al.: Weighted least squares support vector machines: robustness and sparse approximation. Neurocomputing 48(1), 85–105 (2002)
9. Suykens, J.A.K., Vandewalle, J.: Recurrent least squares support vector machines. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 47(7), 1109–1114 (2000)
10. KSuykens, J.A., Lukas, L., Vandewalle, J.: Sparse approximation using least squares support vector machines. In: IEEE International Symposium on Circuits and Systems, Geneva (2000)

A Locality Sensitive K-Means Clustering Method Based on Genetic Algorithms

Lei Gu[1,2]

[1] Guangxi Key Laboratory of Wireless Wideband Communication & Signal Processing,
Gulin, 541004, China
[2] School of Computer Science and Technology,
Nanjing University of Posts and Telecommunications, Nanjing, 210023, China
gulei@njupt.edu.cn

Abstract. The locality sensitive k-means clustering has been proposed recently. However, it performance depends greatly on the choice of the initial centers and only proper initial centers enable this clustering approach to produce a better accuracies. In this paper, an evolutionary locality sensitive k-means clustering method is presented. This new approach uses the genetic algorithms for finding its initial centers by minimizing the Davies Bouldin clustering validity index regarded as the fitness function. To investigate the effective of our approach, some experiments are done on several datasets. Experimental results show that the proposed method can get the clustering performance significantly compared to other clustering algorithms.

Keywords: K-means, Locality sensitive k-means, Genetic Algorithms, Initial centers, Clustering validity index.

1 Introduction

With the rapid developments of computer science, pattern recognition has played an important role in our life. In pattern recognition, a pattern can be represented by a set of *d* attributes or it also can be viewed as a *d*-dimensional feature vector. Clustering is known as one of the popular pattern recognition techniques and utilized to establish the decision boundaries that can divide the dataset into several homogeneous groups called clusters, within each of which the similarity of dissimilarity between data is larger or less than data belonging to different groups[1]. It has been used in wide variety of fields, ranging from image retrieval, image matching, image segmentation and biometric recognition to electrical engineering, mechanical engineering, remote sensing and genetics.

The unsupervised clustering partitions all unlabeled data into a certain number of groups on the basis of one chosen similarity or dissimilarity measure[2, 3]. Different measure of the similarity of dissimilarity can lead to various clustering methods such as k-means[4], fuzzy c-means[5], mountain clustering, subtractive clustering[6] and neural gas[7]. In these classical clustering algorithms, the k-means(KM), which can be easily implemented, is the best-known squared error-based clustering algorithm.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 114–119, 2013.

Recently, the locality sensitive k-means(LSKM) clustering method has been proposed in [8]. Because this algorithm can use the distance between the data points and the cluster centers for calculating the proper weight parameters, it is able to describe the neighborhood structure of the data and gain the better clustering accuracies than the KM clustering method[8].

However, the LSKM algorithm is easily affected by the random initial clustering centers. So it has to obtain the poor clustering performance sometimes. To solve this problem, the genetic algorithms is applied to the LSKM(GA-LSKM) in this paper. The GA-LSKM method can employ the genetic algorithms in finding the good initial centers by minimizing the Davies-Boulidin(DB) clustering validity index[9] viewed as the fitness function.

The remainder of this paper is organized as follows. Section 2 reports the LSKM. In Section 3, the novel GA-LSKM is presented. Experimental results are shown in Section 4, and Section5 gives our conclusions.

2 The LSKM Clustering Algorithm

Let $X = \{x_j \mid j = 1, 2, \cdots, N\}$ be a set of N unlabeled data in the d-dimensional space R^d . Given that X into c clusters and k_i ($i = 1, 2, \cdots, c$) represent cluster centers. The LSKM clustering algorithm can be outlined as follows[8]:

Step1. Initialize c cluster centers k_i ($i = 1, 2, \cdots, c$) randomly.

Step2. Let $S = 1$ and $m_i = k_i$ ($i = 1, 2, \cdots, c$).

Step3. For each data x_j, Compute the weight M_{ij} between x_j and the cluster center k_i by the following formula:

$$M_{ij} = \exp\left(\frac{-\left\| x_j - k_i \right\|^2}{h_i} \right) \tag{1}$$

In Eq.(1), h_i is a parameter and can be gained as follows:

$$h_i = \begin{cases} w_i^2 & x_j \in T_i \\ \left(\dfrac{\sum_{l=1}^{c} w_i}{c} \right)^2 & otherwise \end{cases} \tag{2}$$

where $w_i = \dfrac{\sum_{q=1}^{g} \left\| x_q - k_i \right\|^2}{g}$, g is the number of the neighbors of k_i and T_i is a set with g nearest neighbors of k_i.

Step4. For each data x_j, Compute each element u_{ij} of the matrix U with the follow-ing equation:

$$u_{ij} = \begin{cases} 1 & if \left\| x_j - k_i \right\|^2 is\ minmum, i = 1, 2, \cdots, c \\ 0 & otherwise \end{cases} \tag{3}$$

Step5. Update each cluster center k_i as follows:

$$k_i = \frac{\sum_{j=1}^{N} u_{ij} M_{ij} x_j}{\sum_{j=1}^{N} u_{ij} M_{ij}} \tag{4}$$

Step6. If $\max\limits_{i=1}^{c} \left(\left\| k_i - m_i \right\| \right) < \beta$, then goto Step8; otherwise goto Step7.

Step7. If $S \le \theta$, then let $S = S + 1$, $m_i = k_i$ ($i = 1, 2, \cdots, c$) and goto Step3; other-wise goto Step8. θ is the maximum running times of the LSKM.

Step8. End the LSKM clustering algorithm.

3 The Proposed GA-LSKM Clustering Algorithm

Let $X = \left\{ x_j \middle| j = 1, 2, \cdots, N \right\}$ be a set of N unlabeled data in the *d*-dimensional space R^d. Given that X into c clusters and k_i ($i = 1, 2, \cdots, c$) represent cluster centers.The goodness of the proposed GA-LSKM clustering algorithm can be evaluated by validity indices. The DB index is a sort of clustering validity indices and a function of the ratio of the sum of within-cluster distance to between-cluster separation. The DB index can be defined as follows in [9]:

$$I_{DB} = \frac{1}{c} \sum_{i=1}^{c} \max_{j, j \ne i} \left\{ \frac{F(E_i) + F(E_j)}{f(E_i, E_j)} \right\} \tag{5}$$

where c is the number of clusters, $1 \le i, j \le c$, $F(E_i)$ and $f(E_i, E_j)$ are defined as the within *i*th cluster scatter and the between *i*th and *j*th cluster distance respectively. Here the distance can be chosen as the Euclidean metric. Moreover, when the found clusters have minimum within-cluster scatter and maximum between-class separation, the clustering can minimizes the value of the DB index and is taken as the optimal clustering. The genetic algorithms are applied to the new GA-LSKM clustering approach and the DB clustering validity index is regarded as the fitness function of the genetic algorithms.

Main processes of the proposed GA-LSKM clustering algorithm can be described as follows:

Firstly, the LSKM clustering method is run with the random initial centers.

Secondly, the value of the DB clustering validity index can be computed according to the above Eq.(5). The less value of the DB index indicates the better clustering here if using the DB index evaluates the GA-LSKM clustering method.

Nextly, the genetic algorithms are used for generating the optimal value of $d \cdot c$ parameters δ_i^t ($i = 1, 2, \cdots, c,\ t = 1, 2, \cdots, d,\ \delta_i^t \in [0,1]$) where d is the number of the dimension for one data point and each parameter is applied to producing each dimension of each new cluster centers. Assume that one point $x_j \in X$ ($j = 1, 2, \cdots\ \ , N$) and $x_j = \left\{ \hat{x}_j^t \middle| t = 1, 2, \cdots, d \right\}$. Given $\hat{k}_i^t$ is the *t*th dimension of the *i*th cluster center again. $\hat{k}_i^t$ can be updated by the following equation:

$$\hat{k}_i^t = \left(1 - \delta_i^t\right) \cdot \min_{j=1}^{N}\left(\hat{x}_j^t\right) + \delta_i^t \cdot \max_{j=1}^{N}\left(\hat{x}_j^t\right) \tag{6}$$

After the new cluster centers are producing by Eq.(6), the LSKM clustering method is run with them and the DB index is computed again.

Finally, some operations of the genetic algorithms in the GA-LSKM will be done as follows:

Encoding: In our encoding scheme, each parameter δ_i^t is encoded using 20 bits in a chromosome. A population size is set to 20 chromosomes.

Fitness evaluation: The value of the corresponding DB index is chosen as the fitness function to be minimized.

Selection: Our selection strategy was the stochastic universal sampling routine. Assuming a population of size B, the new population contains $0.9B$ indi-viduals from the original population after selection.

Crossover: The single-point crossover routine is used here. The crossover probability used in our experiments is 0.9.

Mutation: The mutation probability used here is 0.02.

Furthermore, we set the maximum number of generations to 15 here.

4 Experimental Results

To demonstrate the effectiveness of the proposed GA-LSKM clustering algorithm, we compared it with the traditional KM clustering method and the previous LSKM clustering approach on one artificial dataset and four UCI real datasets[10], referred to as the DUNN[11], Iris, Tae, Wine and Haberman datasets. As shown in Fig.1, one artificial dataset DUNN collects 90 2-dimensional instances belonging to two classes. The Iris dataset contains 150 cases with 4-dimensional feature from three classes. The Tae dataset is 5 dimensional dataset with 151 samples of three classes. The Wine and Haberman dataset consist of 178 and 306 data points from three and classes respect-tively. All experiments are done by Matlab on WindowsXP operating system.

For the LSKM and GA-LSKM, on each dataset, we set $\theta = 300$, $\beta = 10^{-5}$ and g can be selected the better value from the set $\{5, 10, \lceil N/3 \rceil, \lceil N/2 \rceil, \lceil 3N/4 \rceil, \lceil 4N/5 \rceil\}$ (N is the number of all data points). Moreover, the average clustering accuracies of the KM, LSKM and GA-LSKM shown in Table 1 are obtained over 20 runs for each dataset. We see clearly from Table 1that the novel proposed GA-LSKM clustering algorithm can achieve the better clustering accuracies than the KM and LSKM clustering method.

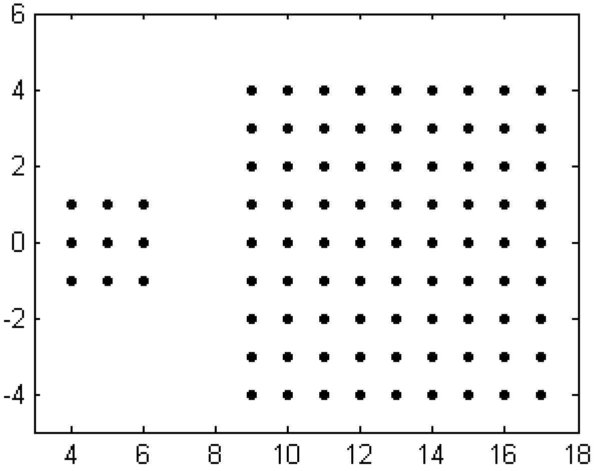


Fig. 1. The Dunn dataset

Table 1. Comparison of the average clustering performance on 20 runs

Datasets	Clustering Accuracies (%)		
	KM	LSKM	GA-LSKM
DUNN	68.50	69.00	**70.00**
Iris	80.93	86.97	**89.13**
Tae	38.21	38.58	**40.00**
Wine	68.31	69.44	**70.22**
Haberman	51.39	55.31	**59.95**

5 Conclusions

In this paper, we propose a novel locality sensitive k-means clustering method based on the genetic algorithms also called the GA-LSKM clustering method. This new approach uses genetic algorithms for generating the optimal initial centers of the GA-LSKM by minimizing the Davies-Boulidin clustering validity index viewed as the fitness function. Some experimental results demonstrate that the proposed GA-LSKM clustering approach can obtain the better clustering performance than the traditional KM and LSKM algorithms.

Acknowledgements. This research is supported by the Foundation of Guangxi Key Laboratory of Wireless Wideband Communication & Signal Processing, Guilin University of Electronic Technology, Guangxi, China (No.21107), and this work is also supported by the Scientific Research Foundation of Nanjing University of Posts and Telecommunications (No.NY210078).

References

1. Fillippone, M., Camastra, F., Masulli, F., Rovetta, S.: A survey of kernel and spectral methods for clustering. Pattern Recognition 41(1), 176–190 (2008)
2. Jain, A.K., Murty, M.N., Flyn, P.J.: Data clustering: a review. ACM Computing Surveys 32(3), 256–323 (1999)
3. Xu, R., Wunsch, D.: Survey of clustering algorithms. IEEE Transactions on Neural Networks 16(3), 645–678 (2005)
4. Tou, J.T., Gonzalez, R.C.: Pattern recognition principles. Addison-Wesley, London (1974)
5. Bezdek, J.C.: Pattern recognition with fuzzy objective function algorithms. Plenum Press, New York (1981)
6. Kim, D.W., Lee, K.Y., Lee, D., Lee, K.H.: A kernel-based subtractive clustering method. Pattern Recognition Letters 26(7), 879–891 (2005)
7. Martinetz, T.M., Berkovich, S.G., Schulten, K.J.: Neural-gas network for vector quantization and its application to time-series prediction. IEEE Transactions on Neural Networks 4(4), 558–569 (1993)
8. Huang, P.H., Zhang, D.Q.: Locality sensitive c-means clustering algorithms. Neuracomputing 73(4), 2935–2943 (2010)
9. Bezdek, J.C., Nikhil, R.P.: Some new indexes of cluster validity. IEEE Transactions on Systems Man Cybernet Part B 28(3), 301–315 (1998)
10. UCI Machine Learning Repository, `http://www.ics.uci.edu/~mlearn/MLSummary.html`
11. Dunn, J.: A fuzzy relative of the ISODATA process and its use in detecting compact well separated clusters. Journal of Cybernet. (3), 32–57 (1974)

Using Graph Clustering for Community Discovery in Web-Based Social Networks

Jackson Gomes Souza [1], Edeilson Milhomem Silva [1], Parcilene Fernandes Brito[1], José Alfredo F. Costa[2], Ana Carolina Salgado[3], and Silvio R. L. Meira[3]

[1] Department of Computing, Centro Universitário Luterano de Palmas,
Av. Theotonio Segurado, 1501 Sul, Palmas, Tocantins, Brazil
[2] Center of Technology, Federal University of Rio Grande do Norte, RN, Brazil
[3] Center of Informatics, Federal University of Pernambuco, PE, Brazil
{jgomes,milhomem}@ceulp.edu.br, parcilene@gmail.com,
alfredo@ct.ufrn.br, {acs,srlm}@cin.ufpe.br

Abstract. Knowledge discovery in social networks is not a trivial task. Often research in this context uses concepts of data mining, social network analysis, trust discovery and sentiment analysis. The connected network of people is generally represented by a directed graph (social graph), whose formulation includes representing people as nodes and their relationships as edges, which also can be labeled to describe the relationship (eg. friend, son and girlfriend). This environment of connected nodes behaves like a dynamic network, whose nodes and connections are constantly being updated. People tend to communicate or relate better with other people who have a common or similar way of thinking, which generates groups of people with common interests, the communities. This paper studies the use of a graph clustering approach, the Coring Method, originally employed in image segmentation task, in order to be applied in the context of community discovery on a social network environment.

Keywords: Graph Clustering, Community Discovery, Data Mining, Web-based Social Networks.

1 Introduction

The concept of "social network" is the basic structure of society [1]. The relationships between people creates the connections needed for the flow of information. A web-based social network (WBSN) is based in this principle. This means there is a need for the creation of new connections between members of this social network because the more connections more new paths for information exchange. This is a great motivation for any WBSN system and one very important task is to constantly motivate the members of the social network not only to produce and share information, but, also, to create new relationships.

In fact, the study of human relations in society has been made by decades and this context is subject of study for researchers in different areas such as psychology [2]

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 120–129, 2013.

and sociology [3]. The application of social networks is broad. Studying it has become the focus of mathematicians and physicists, mainly aiming to discover and to identify the properties of social networks in different contexts. Some of these properties are [4]:

- the "small world effect": the name given to the finding that the average distance between vertices in a network is short;
- the right-skewed degree distributions; and
- clustering, or network transitivity: the property that two vertices that are both neighbors of the same third vertex have a heightened probability of also being neighbors of one another.

These properties affect the approach of identifying network's structure, and, consequently, the understanding of how to contribute to stimulate the creation of new relationships between people.

This paper is in the context of application of a social network based environment for knowledge organization named *Konnen*. This environment is been developed and employed experimentally in a university where professors and students are stimulated to create content and new relations in order to augment the process of learning. The discovery of communities is a key functionality to be available to users because the collaboration in such an environment and the exchange of experience between people occurs more effectively when they are in a common environment, ie. the community.

1.1 Social Network Structure

The social network structure is commonly represented by an undirected graph $G = (V, E)$ where V is the set of nodes (vertices) that represents a set of data objects (people in the social network and related data) and E is the set of edges, which represents the relationships between data, objects. W is a symmetric matrix where w_{ij} is the weight of the edge between nodes i and j. The definition of the edges' weight depends on problem's formulation, but it can be based on a similarity or distance measure. The following sections will present in detail the formulation used in this work.

The graph clustering problem involves the partitioning of a graph in sub graphs, in such a way that nodes of these sub graphs are strongly or densely connected [6] [9].

In the context of social networks G represents the set of connections or relationships (E) between people (V), and a weight w is associated to these relationships, which is a similarity measure based on the proximity of people and their related information, such as profile information, content produced and comments on contents.

2 Related Work

Beyond [4], which focus on the discovery of the structure of social networks, [5] summarizes some of the main approaches for graph clustering and present Markov Clustering (MCL), Iterative Conductance Cutting (ICC) and Geometric MST Clustering (GMC). Other much known techniques are maximum cliques, minimum spanning tree, random walk and spectral clustering [9]. Nibble is a graph clustering algorithm that runs nearly-linear time [7]. The following properties are desirable in graph clustering context [8]:

- **Each cluster should be connected:** about the existence of paths connecting each pair of vertices within a cluster (a subgraph of G);
- **Paths should be internal to the cluster:** the subgraph induced by C (a cluster) should be connected in itself, ie. it is not sufficient for two vertices i and j in C to be connected by a path that passes through vertices in $V\backslash C$, but they also need to be connected by a path that only visits vertices included in C;
- **Edges are classified into two groups:** *internal edges* that connect node i to other vertices in C, and *external edges*, that connect node i to other vertices that are not included in C. The *degree* of node i is the sum of *internal* and *external edges*;
- **Graph cluster** is a connected component.

These properties are important to understand the formulation of the problem that will be presented in the following sections. The following section presents the formulation of the Coring method, as originally proposed by [6] and [9].

3 Coring Method for Graph Clustering

This section summarizes the Coring method for graph clustering described in [6] and [9] and describes its properties, details and applicability.

Coring method assumes that the graph to be clustered contains regions of high density, called "cluster cores", surrounded by non-core regions. Finding these regions is based on a layered clustering approach. For each node i of $H \subseteq V$ its local density is defined as:

$$d(i, H) = \frac{1}{|H|} \sum_{j \in H} w_{ij}. \tag{1}$$

Function $D(H)$ measures the local density of the weakest node of H, which is the minimum density of H:

$$D(H) = \min_{i \in H} d(i, H). \tag{2}$$

An iterative procedure removes weakest nodes from H, which causes D values to increases. Thus, the decreases of D values indicate core nodes of core clusters.

The Coring method for clustering a graph is realized in four steps, outlined in the following procedure.

The Coring method for clustering a graph

```
procedure clustering(G, alpha, beta, delta)
    compute the sequence of density variation
    identify the core nodes based on density variation
    partition the set of core nodes into clusters
    expand the cluster cores into full clusters
```

The following sections present each step in detail.

3.1 Compute the Sequence of Density Variation

First step in Coring method is to define the sequence of density variation. The procedure measures the local density at every node, finds the node with minimum density, remove it, and update local densities of its neighbors on the graph.

Procedure to find the sequence of density variation for *G*

```
procedure density_variation(G)
$H = V$
for $t = 1 \ldots |V|$
    $D_t = \min_{i \in H} \sum_{j \in H} w_{ij}$ (Eq. 2)
    $M_t = \mathrm{argmin}_{i \in H} \sum_{j \in H} w_{ij}$ (Eq. 2)
    $H = H - M_t$
end
return $D_t$ and $M_t$
```

3.2 Identify a Set of Core Nodes Based on Density Variation

Second step involves the identification of the set of core nodes based on density variation (previous step). Considering the sequence of D_t and M_t and two parameters α and β the procedure computes the local rate of decrease R_t on D_t sequence and extracts M_t as a core node if it is a set of β successive M_t whose corresponding $R_t > \alpha$. In other words, β controls the minimum size of a set of core nodes.

Procedure to identify the set of core nodes based on density variation

```
procedure corenodes($D_t$, $M_t$, $\alpha$, $\beta$)
$C = \{\}$
for $t = 1 \ldots |D_t|$
    $R_t = (D_t - D_{t+1})/D_t$
    $bset \leftarrow \{$β-successive $M_t$ with corresponding $R_t > \alpha\}$
    if $R_t > \alpha \wedge M_t \in bset$ then $C \leftarrow C \cup \{M_t\}$
end
return $C$
```

3.3 Partition the Set of Core Nodes into Clusters

In third step, weak edges are removed according a threshold θ and connected components in the remaining graph are found, each of them representing the core of a segment.

Procedure to partition the set of core nodes into clusters

```
procedure partition($C$)
$P \leftarrow \{\}$
in the graph induced by $C$:
    for each edge $w_{ij}$
            if $w_{ij} < \theta$ then remove edge $w_{ij}$
    $P \leftarrow P \cup$ {connected components}
end
return $P$
```

3.4 Expand the Clusters Cores into Full Clusters

In last step, the procedure runs over the sequence of M_t in backwards direction, going from dense to sparse regions of the graph. In this iterative process, the nodes are assigned to the most similar segment (core).

Procedure to expand the cluster cores into full clusters

```
procedure expand($G, P, C, M_t$)
$S \leftarrow \{\}$
$L \leftarrow \{\}$
for $t = |V| \ldots 1$
    if $M_t \notin P$ then
        $m_1 = \max_{C \in S} \text{average}_{i \in C, i \notin L, w_{M_t i} > 0} w_{M_t i}$
        $s = \text{argmax}_{C \in S} \text{average}_{i \in C, i \notin L, w_{M_t i} > 0} w_{M_t i}$
        $m_2 = \text{max2}_{C \in S} \text{average}_{i \in C, i \notin L, w_{M_t i} > 0} w_{M_t i}$
        if $m_2 > 0$ then
            $S \leftarrow S_s \cup \{M_t\}$
            if $m_2 > \lambda * m_1$ then $L \leftarrow L \cup \{M_t\}$
        else
            $S \leftarrow S \cup \{M_t\}$
        end
    end
end
return $S$
```

This procedure considers the graph G, a partition P (found by previous step), the core nodes C (found by step 2) and the sequence of M_t (found in first step). max2 is a function that finds the second maximum value of a set. The term $\text{average}_{i \in C, i \notin L, w_{M_t i} > 0} w_{M_t i}$ measures the similarity between a node i and a segment $C \in S$. The set L contains low-confident nodes, which is, the ones whose similarity with the second nearest segment is greater than half of its similarity with the nearest segment. At last, if a node is not connected with any segment, a new segment is created to enclose it.

This section did present the formulation of Coring method, as proposed by [6] and [9]. Following section contextualizes the use of Coring method in the task of clustering a social graph, aiming to find communities.

4 Clustering the Social Graph

The Coring method has been applied in image segmentation task [6]. This paper employs the Coring method in the task of clustering the social graph generated by an instance of a WBSN platform named *Konnen*. Information from this platform's data structure, which is relevant to the current problem formulation, is as follows:

1. *Profiles*: containing information about personal data, like full name;
2. *ProfileItems*: containing detailed information about personal interests, educational and professional experience,
3. *Contents*: containing information about users content publications, ie. posts like texts and pictures (their description); and
4. *Comments*: containing users' comments about users' content publications.

Because the text-based nature of the data and the nature of the social network environment itself, the following assumptions are important:

1. **The social graph is fully connected**. This means all nodes are connected and their associated weights represent a value that must be considered as the possibility for two nodes to be part of one community in addition to the concept of similarity between them (the measure of similarity between two users). We do not take into account the information about users' relationships.
2. **The text-based nature of the data**. Data in the social graph is mainly text-based. This means the function that generates edges weights must consider this characteristic.

We use the *Vector Space Algebraic Model* to represent user's data. For the definition of the similarity measure between two users' data, we adopt the *cosine measure*. The Coring method has some parameters, whose value define the behavior of the approach in the graph clustering task. In the next section we show experimental results.

5 Experimental Results

The dataset (the data from the social network) considering in the experiment has the following characteristics:

1. 75 users (containing person genre information)
2. 380 content publications (containing content text or description)
3. 197 profile items (119 of which contain educational experience information and 47 contain work experience information)
4. 148 comments (the users' comments about content published by other users)

The social graph generated by this dataset is present by Fig 1.

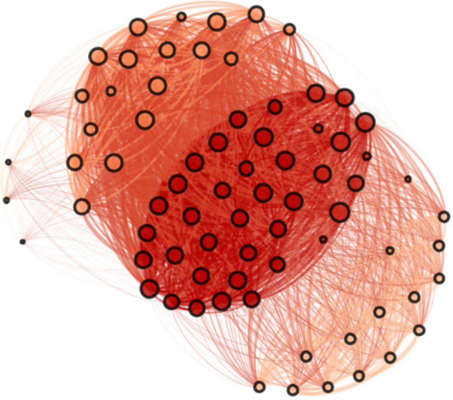


Fig. 1. Original social graph generated by data in the social network

Fig. 1 shows the social graph generated by dataset in the social network using a layout method known as *YifanHu's Multilevel* [10]. Vertices sizes are based on weighted degree. Edges weights are used to define the edges thickness. From the problem's formulation we can see that the graph is fully connected (excluding edges with weight = 0.0) and there are regions of more density.

Graph clustering task uses the Coring method present previously. Experiments on the dataset showed that:

a) The value of α influences the generation of core nodes considering the density variation. Values greater than or equal 0.1 generates core nodes segments of size = 0 or 1, which, consequently, generates no clusters or only one cluster;
b) The value of β influences the size of core nodes segments. Small value (eg. 1) tends to generate more clusters;
c) The value of θ is a threshold for cleaning up edges with small weights from the set of core nodes.

In order to analyze graph clustering results we setup experiments summarized in Table 1 with results shown by Fig. 2.

Table 1. Experiments setup, varying values for parameters

#	α	β	θ
1	0.005	1	0.1
2	0.005	1	0.9
3	0.005	4	0.1
4	0.005	4	0.9
5	0.05	1	0.1
6	0.05	1	0.9
7	0.05	4	0.1
8	0.05	4	0.9
9	0.0	1	0.9

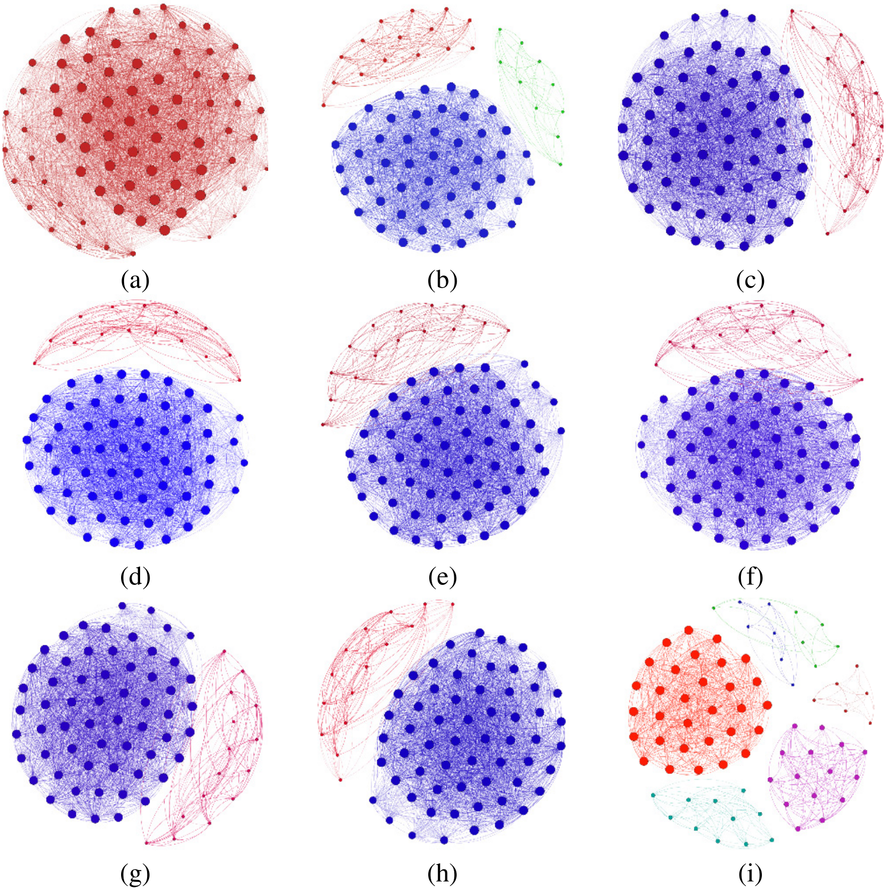


Fig. 2. Graph clustering results, with variations of parameters' values

Fig. 2 presents graph clustering results from experiments. The layout method for graph visualization is called *Fruchterman Reignold* [11]. Visual analysis shows that results from Experiments #3 to #8 are exactly the same, with small variations in rotation, color and positioning of some nodes. Result from Experiment #1 is close to Experiments #3 to #8, but the small value of θ influences in such a way that the two visible dense regions are still connected (their internal nodes). Experiment #2 is different from Experiments #3 to #8 because generates more clusters, thus we note the influence of small values for α and β and high value for β. Experiment #9 differs from others because it has six clusters. The influence is clearly the value of α tending to 0.0.

Fig. 3 helps to understand the behavior of variation of D_t sequence.

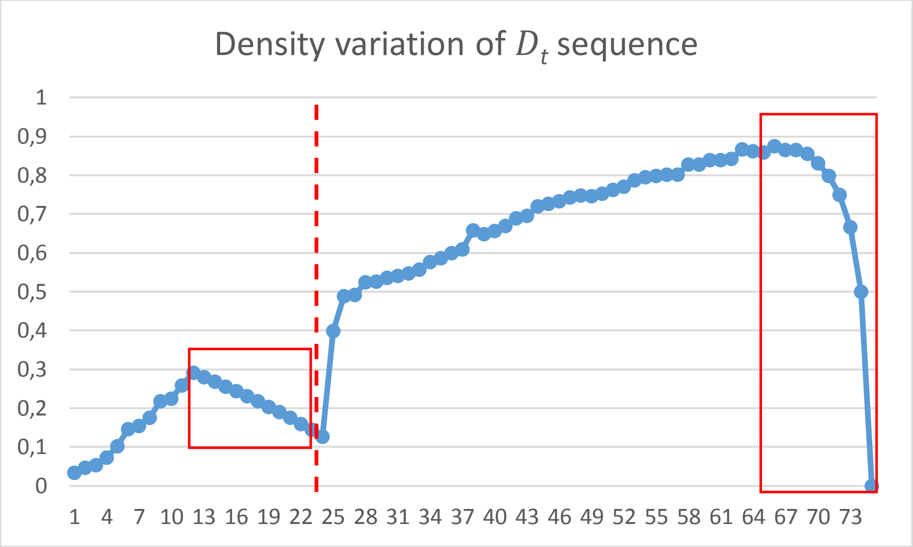


Fig. 3. Density variation for graph from social network dataset

Fig. 3 shows a graph that presents the density variation of D_t sequence for the graph generated from the social network dataset. The [red] rectangles show the regions with nodes in G that represent core segments, ie. they satisfy the condition $R_t > \alpha$. The analysis of Fig. 3 shows that the correct clustering solution should contain two clusters. This is what happens with Experiments #3 to #8. In Experiment #9, with $\alpha = 0.0$, the amount of regions extracted from the D_t sequence is higher. This explains the trend to generate more clusters.

6 Conclusions

Data clustering is a common task in data mining scenario. The high dimensionality of data is a problem for people who try to visualize it, since is common sense that humans have difficult to visualize information beyond third dimension. Web-based Social Networks are a very important context of analysis. Considering the growing of social communities in the web, and even the commercial importance of the information this context holds, it is of special interest the understanding and the extraction of information in the dataset provided by a social network. One example applicability is the discovery of communities in the social network, that takes into account the establishment of some measure of similarity between the social network members and the content they produce.

This paper employs the Coring method in the task of graph clustering aiming to discover communities in a social network environment named *Konnen*. We show the problem formulation and results from experiments considering Coring methods' parameters variations. Specifically we verify the influence of parameters α, β and θ.

Experimental results show that the most adequate combination is the setup: $\alpha = 0.5$, $\beta = 4$ and $\theta = 0.9$, which results in the finding of two communities in the dataset.

Following [8], still remains the necessity of a quantitative measure of graph clustering quality. In a future work, we pretend to understand the influence of a graph clustering quality measure and, because the natural characteristic of this problem, we plan to use an optimization method, like *Particle Swarm Optimization* and *Genetic Networks*. Finally we need also to understand the influence of other similarity measures. The one used in this paper considers the same importance for all sources of social information (ie. *Profiles*, *ProfileItems*, *Contents* and *Comments*). A weight should be assigned to these sources of information in order to understand the influence of each one of them.

References

1. Pimentel, M., Fuks, H.: Sistemas Colaborativos, p. 35. Elsevier (2001)
2. Barnes, J.A.: Social Networks. Addison-Wesley Module in Anthropology 25, 1–29 (1972)
3. Wellman, B.: Studying personal communities. In: Marsden, P. (ed.) Social Structure and Network Analysis. Beverly Hills (1982)
4. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences 12, 7821–7826 (2002)
5. Brandes, U., Gaertler, M., Wagner, D.: Experiments on Graph Clustering Algorithms. In: Di Battista, G., Zwick, U. (eds.) ESA 2003. LNCS, vol. 2832, pp. 568–579. Springer, Heidelberg (2003)
6. Le, T.V., Kulikowski, C.A., Muchnik, I.B.: A graph-based approach for image segmentation. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y.K., Rhyne, T.-M., Monroe, L. (eds.) ISVC 2008, Part I. LNCS, vol. 5358, pp. 278–287. Springer, Heidelberg (2008)
7. Spielman, D. A., Teng, S.: A Local Clustering Algorithm for Massive Graphs and its Application to Nearly-Linear Time Graph Partitioning. The Computing Research Repository (2008)
8. Schaeffer, S.E.: Survey: Graph Clustering. Comput. Sci. Rev. 1(1) (2007)
9. Le, T.V., Kulikowski, C.A., Muchnik, I.B.: Coring method for clustering a graph. In: 19th International Conference on Pattern Recognition, ICPR 2008, pp. 1–4 (2008)
10. Hu, Y.F.: Efficient and high quality force-directed graph drawing. The Mathematica Journal 10(1), 37–71 (2005)
11. Fruchterman, T.M.J., Reingold, E.M.: Graph Drawing by Force-Directed Placement. Software: Practice and Experience 21(11) (1991)

Application of Dynamic Rival Penalized Competitive Learning on the Clustering Analysis of Seismic Data

Hui Wang[1], Yan Li [2,*], and Lei Li [3]

[1] School of Banking and Finance, University of International Business and Economics,
Beijing, China
[2] School of Insurance and Economics, University of International Business and Economics,
Beijing, China
liyan@163.com
[3] BGP INC., China National Petroleum Corporation

Abstract. Rival penalized competitive learning (RPCL) has provided attractive ways to perform clustering without knowing the exact cluster number. In this paper, a new variant of the rival penalized competitive learning is proposed and it performs automatic clustering analysis of seismic data. In the proposed algorithm, a new cost function and some parameter learning methods will be introduced to effectively operate the process of clustering analysis. Simulations results are presented showing that the performance of the new RPCL algorithm is better than other traditional competitive algorithms. Finally, by clustering the seismic data, a kind of geological characteristic, underground rivers, can be extracted directly from the 3D seismic data volume.

Keywords: Rival penalized competitive learning (RPCL), Pattern recognition, Clustering analysis, Clustering analysis, Geological characteristic.

1 Introduction

Competitive learning, as an efficient way of data analysis, has been widely applied in many fields such as classification, economic prediction, image processing [1][2]. The process of competitive learning clustering aims at dividing a set of data into different groups so that members of the same group are more alike than those of different groups. However, the major problem in typical competitive learning algorithms (such as K-means algorithm [3]) is that the number of clusters should be given before clustering. Otherwise, it will lead to a poor clustering result. In addition, because of the effect of initial values, many competitive learning algorithms often fall into the local optimum solutions, that is, if a center is inappropriately activated, it may never be modified.

During the past decade, some advanced algorithms have been proposed to tackle these problems without the exact cluster number, for example: fuzzy c-means

[*] Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 130–136, 2013.

algorithm [4], the frequency competitive algorithm (FSCL) [5] [6], the rival penalized competitive learning (RPCL) algorithm and its variants [7]-[9]. These algorithms often reduce the learning rate of the frequent winners, so that all of centers have chance to win the competition. In many fields, these algorithms have been proposed for different types of samples. The fuzzy c-means and FSCL algorithm, however, need knowing the exact number of clusters. The RPCL algorithm performs exact clustering without knowing the number of clusters and it can eliminates the local optimum problem appropriately. The RPCL algorithm can find the correct centers of clusters by rewarding the winning center and penalizing the second winner, named rival center. Although the RPCL algorithm performs better than k-means algorithm and FSCL algorithm, the convergence rate of the classical RPCL algorithm is slow.

In this paper, a dynamic RPCL (DRPCL) algorithm is introduced. In the proposed algorithm, a new cost function and parameter learning rules (about the adaptive updating of the learning rate and the penalizing rate) are introduced to accelerate the convergence of the RPCL algorithm. Besides this, the Mahalanobis distance is used instead of the Euclidean distance in the experiments in order to distinguish high correlation of clusters.

In our experiments of DRPCL, synthetic and real 3D seismic data sets are used to demonstrate the performance of the proposed approach. First, a simple 2D synthetic data set is presented to compare the efficiency of DRPCL with other competitive learning approaches. Then, the approach proposed is implemented to identify the underground rivers. In this way, if the initial number of clusters N is greater than the actual number of geological characteristics, extra initial weight centers will be driven out of the data set. Therefore, more accurate information on the geological characteristics can be extracted from the multi-attribute seismic data set. The results show that DRPCL is a robust method and can be used to reveal geological relevant information in the petroleum exploration.

The remainder of this paper is organized as follows. Section 2 and 3 describe RPCL and the proposed dynamic RPCL methods, respectively. The results of simultaneous data and the seismic multi-attributes data are presented in Section 4. Then, we draw a brief conclusion in the last section.

2 The Rival Penalized Competitive Learning Algorithm

The rival penalized competitive learning (RPCL) algorithm is an unsupervised clustering algorithm proposed by Xu et al [7], which can perform clustering without knowing the clusters number and overcome the “local optimum” problem. The main idea of the RPCL algorithm is that the center of the best unit is rewarded to adapt to the input, while the weight vector of its rival (the second best) is penalized with a small penalizing rate. According to the given learning and penalizing rates, RPCL can automatically allocate the appropriate weight vectors, while driving the weight vectors of the extra units far away at the same time. In addition, the algorithm is quite simple and provides a better convergence than k-means algorithm.

Generally, input samples are represented as a d -dimensional vector set, $S = \{X^u\}_{u=1}^N$ with $X^u = [x_1^u, x_2^u, ..., x_d^u] \in \Re^d$, where N and d are the number of samples and the samples dimension, respectively. Besides this, the clustering centers $\{W^i\}_{i=1}^k$ can be expressed by a d -dimensional vector, $W^i = [w_1^i, w_2^i, ..., w_d^i]$, and k is the accurate number of data set. In the simple RPCL algorithm, for a random sample X^u, the winning center vector, W^c, with a minimum distance, is defined by the following function:

$$c = \arg\min \gamma^i \left\| X^u - W^i \right\| \quad i = 1,..,k\ , \tag{1}$$

where the relative winning frequency, γ^i, is defined as $\gamma^j = t^j \Big/ \sum_{i=1}^k t^j$ and t^j is the number of times when the center W^j was defined winner in the past. Furthermore, the second winning center (or rival center), r, is defined as:

$$r = \arg\min \gamma^i \left\| X^u - W^i \right\| \quad i = 1,..,k\ ,\ i \neq c\ . \tag{2}$$

Because of the influence of the winning frequency, γ^i, the centers which won the competition during the past, have a smaller chance in the next time.

After choosing out the winner W^c and the rival center W^r, the RPCL algorithm updates centers, $\{W^i\}_{i=1}^k$, and meanwhile adjusts the winning frequency, $\{\gamma^i\}_{i=1}^k$, with the following functions.

$$W^i = \begin{cases} W^i + a^c \partial E \big/ \partial W^i & i=c \\ W^i - a^r \partial E \big/ \partial W^i & i=r \\ 0 & i \neq c \ \ i \neq r \end{cases} \tag{3}$$

$$\gamma^i = \gamma^i + 1 \quad i = c \tag{4}$$

where $0 < a^c < 1$ is the learning rate for the winner center W^c and $0 < a^r < 1$ is the penalizing rate for the rival center W^r, which is often much smaller than the learning rate. From above functions, we can find out that the winning center W^c is rewarded and moved close to the correct center, while the rival center W^r is penalized and moved away from the correct center. In the application, all of the parameters are randomly initialized. Functions (1)-(4) are than used iteratively until the algorithm converges or the number of iterations reaches a pre-given value, respectively. The RPCL algorithm can always successfully distribute the samples into k clusters without the local optimum, even if the correct number of clusters is unknown.

However, there are many weaknesses in this algorithm. For example, the convergence of the classical RPCL algorithm is sensitive to the cost function and the learning or penalizing rate. Then, the distance between a sample and the weight vector is mainly used in the form of the Euclidean distance, which limits the clusters to other fields.

3 The Dynamic Rival Penalized Competitive Learning Algorithm

The proposed algorithm is a variant of the typical RPCL algorithm. In this paper, a special kind of cost function is introduced to dynamically control the times of the loop in the RPCL algorithm. The cost function can be expressed as:

$$E(W)=\frac{1}{2}\sum_{u}\left\|X^{u}-W^{c(u)}\right\|^{2}+\eta\sum_{u,i\neq c(u)}\left\|X^{u}-W^{c(u)}\right\|^{2} \tag{5}$$

where $c(u)$ denotes the index of the winner center for the sample X^{u}, and η is called the regulatory factor. That is, once the cost function is minimized, the RPCL will be stopped immediately.

In the typical RPCL algorithm, the two variables, the learning rate and the penalizing rate, are usually assumed to have a fixed value or simple decreasing functions $a^{c}(t)$ and $a^{r}(t)$ with the loop time t . But in the algorithm, the two key values are estimated dynamically, and can be expressed by the following function:

$$a^{c}(t,W)=\rho(t)\frac{\left\|X^{u}-W^{c}\right\|}{(\left\|X^{u}-W^{c}\right\|+\left\|X^{u}-W^{r}\right\|)}$$

$$a^{r}(t,W)=\lambda(t)\frac{\left\|X^{u}-W^{r}\right\|}{(\left\|X^{u}-W^{c}\right\|+\left\|X^{u}-W^{r}\right\|)} \tag{6}$$

where $\lambda(t)$ and $\rho(t)$ are two monotonous decreasing functions of the loop time t . The learning rate and penalizing rate are not only related to the loop time, t , but are also influenced by the current weight vector, $\{W^{i}\}_{i=1}^{k}$. Because the value of $\left\|X^{u}-W^{c}\right\|$ is always less than that of $\left\|X^{u}-W^{r}\right\|$, the greater punishment will be put to the rival center. Finally, the Mahalanobis distance norm is used instead of the Euclidean distance norm in the cost function computation in order to distinguish high correlation data set.

4 Experiments

4.1 Simulations Data

A simple 2D synthetic experiment is carried out to demonstrate the performance of the DRPCL method. The procedure aims to compare the results of the DRPCL algorithm, K-means algorithm and typical RPCL algorithm. In the section, dataset in a two dimensional space with predetermined number of clusters were generated with variances 2.0, 1.0 and 1.0, centered at (-0.5, 0.0), (0.5, 1.0) and (0.5, -1.0) (Fig.1.a.). Points were assigned to each center according to a Gaussian distribution and generated randomly. A total of 300 points were associated with each cluster. In the simultaneous data set, the strong overlap between different clusters is very obvious (Fig. 1a), that would affect the results of clustering algorithms. The results of K-means, the typical RPCL and DRPCL are presented in Fig.1.b-1.d, respectively. In order to get a clear look, the final results of the cluster are shown with different colors in each experiment. Besides this, all the centers were initialized with the same random value and the initial number of the cluster is set to 5 for all. In Fig. 1, we find that the K-means algorithm converges to the local results because of the incorrect number of clusters. The typical RPCL algorithm could find the desired centers, drive away two extra centers from the sample data. But there are some errors in the results from the RPCL algorithm. It is clear that the distance between the sample and the cluster calculated by Euclidean norm, is not accurate enough to describe the complex data set. In contrast, the DRPCL algorithm extracts accurate information from the sample data set by using the Mahalanobis norm to calculate the distance between each sample and the center.

4.2 Real Seismic Data

Original seismic data was obtained from Western China. After some signal denoising operations, physical and geometry seismic attributes are obtained from original seismic data to quantify the characters of the geological objective reservoir. Although geometry seismic attributes can be easily accepted and understood, physical seismic attributes derived from abstract mathematical algorithms may be applied more widely than geometry attribute in seismic reservoir analysis. In the past several years, many of pattern recognition techniques have been used in the field of seismic exploration [10]-[14]. The proposed method is applied to identify the geological characteristic of underground rivers. The seismic data along the target horizon (green line) is shown in Fig.2a. Along the target horizon, a 20-ms window is used to extract the seismic attributes (sample characteristics) by some mathematical algorithms. In order to identify geological characteristic, we chose four conventional seismic attributes (energy, instantaneous amplitude, phase and curvature) to construct the 4-dimension sample data set, which consists of 200*500 points. Fig.2b shows the identification results of the proposed algorithm, in which the initial number of clusters is equal to ten. Finally, seven kinds of distinct, primary geologic characteristics were determined by clustering, while two extra initial centers were driven away from the sample set. From Fig.2, we find that these areas (marked by red, orange and blue) are an important oil and gas reservoir in seismic exploration. Therefore, the experiment proves that our method is efficient and produces reliable results.

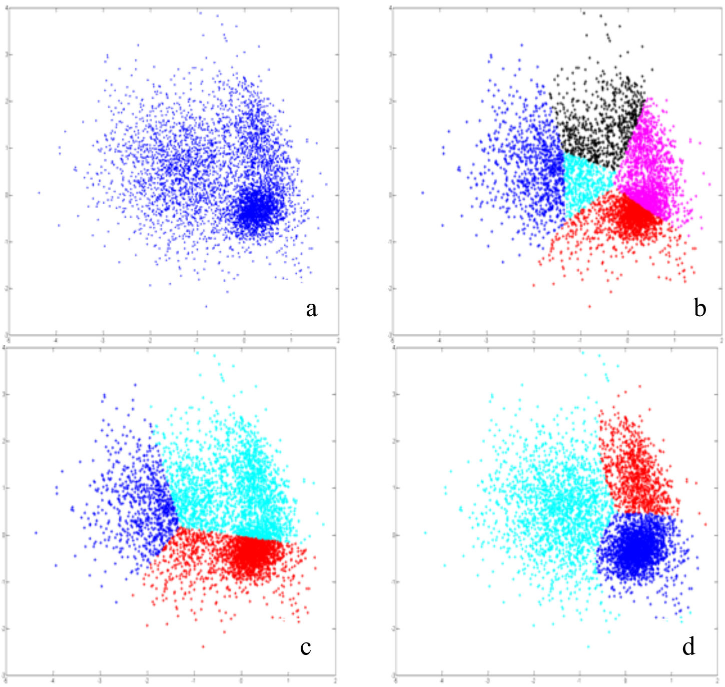


Fig. 1. The synthetic model experiment. (a) Original data set. (b) Results of K-means algorithm. (c) Results of typical RPCL algorithm. (d) Results of DRPCL algorithm.

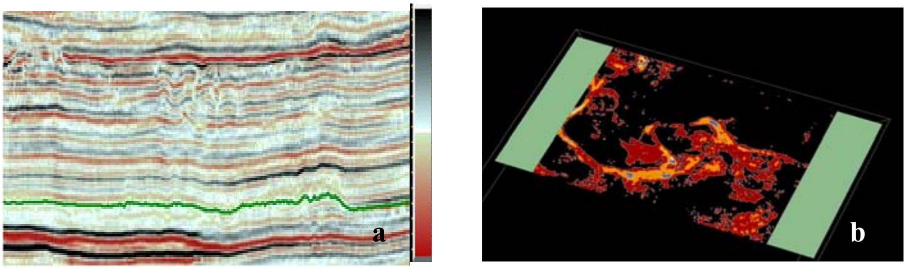


Fig. 2. (a) Seismic section along the "inline" direction. The green line is picked target horizon. Color bar is amplitude. (b) The clustering results of the underground river deposits (marked by red, orange and blue).

5 Conclusions

The proposed DRPCL algorithm achieves an automatic classification process for the sample data set, which dynamically updates the learning rate and penalizing rate. Compared to the traditional competitive algorithms, the proposed method reduces the influence of the initial values in the competitive algorithms and obtains more accurate and reasonable classification results. In addition, the proposed algorithm adaptively estimates a credible number of underground river deposits in the target zone, which not only helps the interpreter investigate the geologic structure of the target area, but also compensates for the uncertainty and subjectivity of the seismic interpreter.

References

1. Kecman, V.: Learning and Soft Computing, Support Vector Machines, Neural Networks and Fuzzy Logic Models. The MIT Press, Cambridge (2001)
2. Wang, L.P., Fu, X.J.: Data Mining with Computational Intelligence. Springer, Berlin (2005)
3. MacQueen, J.B.: Some Methods for Classification and Analysis of Multivariate Observations. In: Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297. University of California Press, Berkeley (1967)
4. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York (1981)
5. Ahalt, S.C., Krishnamurty, A.K., Chen, P., Melton, D.E.: Competitive Learning Algorithms for Vector Quantization. Neural Networks 3, 277–291 (1990)
6. Banerjee, A., Ghosh, J.: Frequency Sensitive Competitive Learning for Scalable Balanced Clustering on High-dimensional Hyperspheres. IEEE Transactions on Neural Networks 15(3), 702–719 (2004)
7. Xu, L., Krzyzak, A., Oja, E.: Rival penalized competitive learning for cluster analysis RBF net and Curve Detection. IEEE Trans. Neural Network 4(4), 636–649 (1993)
8. Ma, J.W., Wang, T.J.: A Cost-function Approach to Rival Penalized Competitive Learning (RPCL). IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics 36(4), 722–737 (2006)
9. Krzyzak, A., Linder, T., Lugosi, G.: Nonparametric Estimation and Classification using Radial Basis Function nets and Empirical Risk Minimization. IEEE Trans. Neural Network 7(2), 475–487 (1996)
10. Matos, M.C., Osorio, P.L.M., Johann, P.R.S.: Unsupervised Seismic Facies Analysis using Wavelet Transform and Self-organizing Maps. Geophysics 72(1), 9–21 (2007)
11. Saggaf, M.M., ToksÖzz, M.N., Marhoon, M.I.: Seismic Facies Classification and Identification by Competitive Neural Networks. Geophysics 68(6), 1984–1999 (2003)
12. Fernando, A.N., Harold, T.: Multi-attribute Seismic Volume Facies Classification for Predicting Fractures in Carbonate Reservoirs. The Leading Edge 25(6), 698–700 (2006)
13. Gao, D.L.: Application of Seismic Texture Model Regression to Seismic Facies Characterization and Interpretation. The Leading Edge 27(3), 394–397 (2008)
14. Marroquin, I.D., Brault, J.J., Hart, B.S.: A Visual Data-mining Methodology for Seismic Facies Analysis: Part 1-Testing and Comparison with other Unsupervised Clustering Methods. Geophysics 74(1), 1–11 (2009)

An Online Trend Analysis Method for Measuring Data Based on Historical Data Clustering

Jianfeng Qu*, Maoyun Guo, Yi Chai, Zhimin Yang, Tao Zou, Tian Lan,
and Zhenglei Liu

School of Automation at the Chongqing University, No. 174 Shazhengjie,
Shapingba, Chongqing, 400044, China
qujianfeng@cqu.edu.cn

Abstract. It is important to analyze and predict the measuring data trend in industrial measuring and controlling process. The paper introduces a method for predicting the trend of the current measuring data based on clustering the historical data. It calculates the similarities of the current trend and the bases result from the clustering. And with these similarities, the future trend of the current measuring data can be predicted , the combination of the above bases representing low frequency and a reviser representing high frequency. The simulation shows the weights of high or low frequency have effect on the precision of predict results. It is also found that the proposed method can predict more precisely than the RBFNNs method in high frequency.

Keywords: Trend analysis, predicting, historical data, clustering.

1 Introduction

In the industrial process, analyzing and predicting the trend of measuring data is meaningful. Many studies have been conducted in the fields of predicting the trend of time series. D.B.L. Bong propose a new method called K-means-Greedy Algorithm (KGA) to automate the process of finding the optimal value of the number of neurons in the hidden layer of a BP network that is used to perform prediction of a time series [1]. Yang build a forecasting model of service marketing based on cluster analysis and decision tree [2]. Johannes use K-Means Clustering to predict the vehicle velocities, which can be used for any point of time in the future. The approach is based on a two step clustering utilizing toll transaction data of the training of the model [3]. Zheng present a generalized regression neural network (GRNN) based approach to predict the amount of nitrogen oxides emitted from coal-fired boiler. A novel set of 'multiple' smoothing parameters were assigned to GRNN model and K-means clustering algorithm was developed, it was used to reduce the number of smoothing parameters [4]. Raimir present a method for traffic prediction based on multidimensional OD traffic including two

[*] Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 137–145, 2013.

components: predicting short-term traffic using Principal Components Analysis using a technique for dimensionality reduction and a local linear model based on K-means as a technique for prediction and trend analysis [5]. Yiakopoulos use K-means to automatically diagnose the defective rolling element bearings. K-means clustering is an unsupervised learning procedure, which can be directly implemented to measure vibration data. Thus, there is no need for training the method with data measured on the specific machine under defective bearing conditions [6]. Mohammed use the RBF Neural Networks based on the clustering to predict the data series, and assesses the performance of the smooth time series data prediction system [7].

In this paper, a trend predicting method for measuring data is introduced based on historical data clustering. The result is supposed to be the bases of the typical patterns of the measuring data. So for predicting online trend, the value of the future period is the linear combination of the bases in the same period and a reviser of current measuring data.

2 The Clustering of Historical Data

This section presents the clustering method of the historical data which sets the clustering centers to be the bases of the typical patterns of the historical measuring data.

2.1 The Data Structure for the K-means Algorithm

For the trend analysis, the data structures of the pattern and measuring data are defined as follows, the measuring data can be represented as a vector which has N components $M_i = (m_{i,1}, m_{i,2}, m_{i,3}, ..., m_{i,N})$, M_i is the ith measuring data which contains N components, $m_{i,1}, m_{i,2}, ..., m_{i,N}$. And $m_{i,j}$ is the value of the jth sampling point of ith measuring data, where $j = 1, ..., N$, $i = 1, 2, ..., S$. The cluster centers $C_l(t)$ are defined as follows: $C_l(t) = (c_{l,1}(t), c_{l,2}(t), c_{l,3}(t), ..., c_{l,N}(t))$, $C_l(t)$ is the lth clustering center of the K-means clustering at the moment t, where $l = 1, ..., K$.

2.2 The K-means Clustering of the Historical Data

The K-means clustering algorithm is a typical clustering algorithm to get the typical patterns of the historical measuring data.

Suppose there are historical measuring data M_i , $i = 1, 2, ..., S$ to be clustered to K groups, the clustering process include the following steps:

Step 1. Set the historical measuring data M_i as clustered data and select one piece of them as the initial centers $C_l(0)$, $l = 1, ..., K$;

Step 2. Calculate the distances between each piece of the historical measuring data and each clustering center $d_{i,l}(t) = ||M_i - C_l||$, $d_{i,l}(t)$ is the distance between the ith measuring data and the lth clustering center at moment t;

Step 3. For each clustering center, the membership of measuring data is verified according to the following rules: if the measuring data M_i is the nearest to the Lth clustering center $C_L(t)$, i.e., the distance $d_{I,L}(t)$ is smallest among the distance dataset including $d_{i,1}(t), d_{i,2}(t), , d_{i,k}(t)$, then M_i belongs to the dataset which is corresponding to the Lth clustering center $C_L(t)$ and can be labeled as $M_p(t, L)$, $p = 1, 2, ..., P_L$, for which there are P_L members in the dataset which corresponding to the Lth clustering center $C_L(t)$.

Step 4. Update the clustering centers. After verifying the membership of each measuring data, update the clustering center using equation $C_L(t+1) = \frac{1}{P_L} \sum\limits_{p=1}^{P_L} M_p(t.L)$;

Step 5. Calculate the error $\varepsilon_L(t)$ between the current centers $C_L(t+1)$ and previous centers $C_L(t)$ using equation $\varepsilon_L(t) = \|C_L(t+1) - C_L(t)\|$. If $\varepsilon_L(t) < \varepsilon$, $L = 1, ..., K(\varepsilon$ is the threshold value for ending the clustering procedure), then the clustering procedure ends and the final clustering center $C_L(t+1)$ is the K-means clustering result; otherwise, go to the Step 2 and continue.

3 The Trend Prediction Based on the Clustering

With the clustering result of the clustering centers C_i, this section introduces a method to predict the trend of current measuring data.

In the section, the ith pattern is defined as a vector

$$P_i = (p_{i,1}, p_{i,2}, p_{i,3}, ..., p_{i,N}) \tag{1}$$

which contains N components $p_{i,1}, p_{i,2}, p_{i,3}, ..., p_{i,N}$, each component corresponds with a sampling point in the historical measuring, the pattern of the measuring data can be represented as the above vector. Following the K-means algorithm of the historical data in section 2, the ith pattern P_i is the ith cluster center C_i.

3.1 The Similarity between the Measuring Data and the Historical Measuring Data Pattern

There is shortest distance exists between the measuring data and the clustering centers, which contain the features of measuring data. The final cluster center can be taken as the typical patterns. So at the current moment t_c in the current measuring procedure, there are measuring values $M_{t_c} = (m_{c,1}, m_{c,2}, ..., m_{c,t_c})$, $0 \leq t_c \leq N$. If $M_{t_c} \neq P_l$, the similarity between the current measuring data and the pattern can be defined as

$$s_l(t_c) = \frac{1}{\sum\limits_{i=1}^{t_c} a_i \left|(m_{c,i} - p_{l,i})\right|} \tag{2}$$

where $s_l(t_c)$ is the similarity of the measuring data at the moment t_c and the lth clustering center. $p_{l,i}$ is the ith component of the lth pattern P_l which is the lth clustering centers C_l in fact, and a_i is the ith weight.

3.2 The Value Representation of the Current Measuring Data in the Future Moment Based on the Typical Patterns

This section presents a method to represent the trend of the measuring data based on the similarities of the measuring data and the typical patterns.

With the similarities $s_l(t_c)$ of the current measuring data and the patterns in the Eq.(2), the value m_{c,t_f} in the future moment t_f $(t_f > t_c)$ can be predicted as

$$m_{c,t_f} = \sum_{l=1}^{K} b_l s_l(t_c) p_{l,t_f} \tag{3}$$

where b_l is the weight of the lth patterns, which is assumed to be 1 in this paper for simplicity. Since the clustering obtains the mean value of the data, it can be regarded as a filter which blocks the high frequency characteristic of the data. Therefore, there is few data items of high frequency in the m_{c,t_f} in the Eq.(3) from the clustering result due to the low frequency part of the data. And the m_{c,t_f} can be modified as

$$m_{c,t_f} = \beta_1 \sum_{l=1}^{K} b_l s_l(t_c) p_{l,t_f} + \beta_2 \Delta \tag{4}$$

where Δ is a reviser based on the differential of the current measuring data just before current moment t_c, β_1 and β_2 are the weights which can be calculated as equation

$$\Delta = (t_f - t_c) \cdot \frac{(m_{c,t_c} - m_{c,t_s})}{t_c - t_s} + m_{c,t_c} \tag{5}$$

where t_s is the past moment, and $0 < t_s < t_c$. The m_{c,t_s} is the measuring data of the moment t_s, $\frac{(m_{c,t_c} - m_{c,t_s})}{t_c - t_s}$ is the gradient of the current measuring data from the moment t_s to the current moment t_c which is substituted for the gradient from the current moment t_c to the future moment t_f.

3.3 The Online Trend Prediction Method Based on Clustering

With the definition and calculation of the similarity of the typical patterns of the historical measuring data and current measuring data, this section presents the method of predicting the measuring data trend as shown in Fig. 1.

4 The Simulation

In this section, the proposed method is simulated and the simulation results are shown in the Fig. 2-Fig. 4, where the Time unit (horizontal axis) is second and the Value unit (longitudinal axis) is voltage. The historical measuring data and the curves of the K-means clustering are shown in Fig. 2, the clustering center of the historical data (dash-dot line), the K-means clustering of the historical data (dotted line) and the groups of historical data (solid line) are shown in Fig. 3, and Fig. 4 show the results of clustering centers and the predictions of the future segment using the method discussed in section 3.

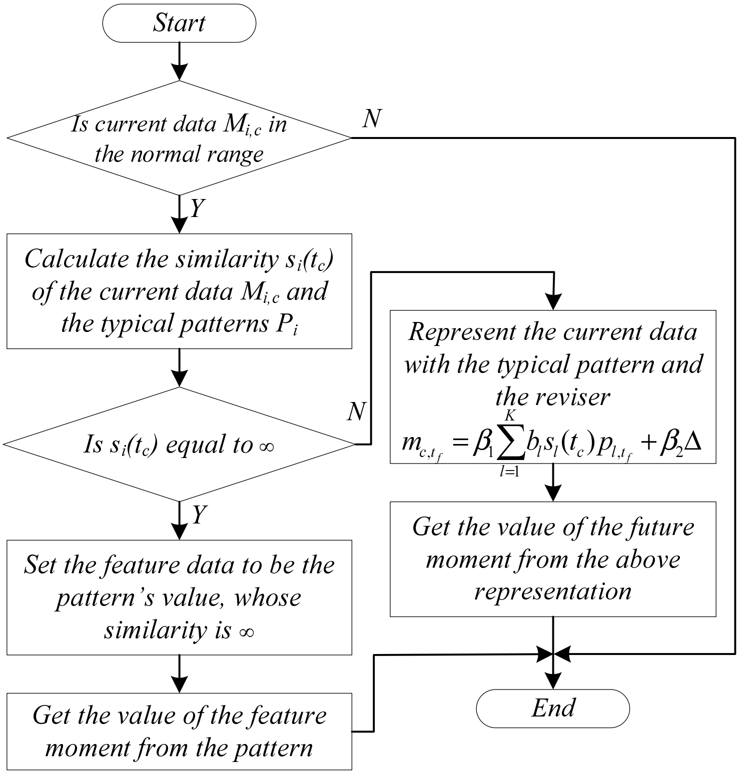

Start
Is current data $M_{i,c}$ in the normal range
N
Y
Calculate the similarity $s_i(t_c)$ of the current data $M_{i,c}$ and the typical patterns P_i
Is $s_i(t_c)$ equal to ∞
N
Y
Represent the current data with the typical pattern and the reviser
$m_{c,t_f} = \beta_1 \sum_{l=1}^{K} b_l s_l(t_c) p_{l,t_f} + \beta_2 \Delta$
Get the value of the future moment from the above representation
Set the feature data to be the pattern's value, whose similarity is ∞
Get the value of the feature moment from the pattern
End

Fig. 1. The trend prediction process

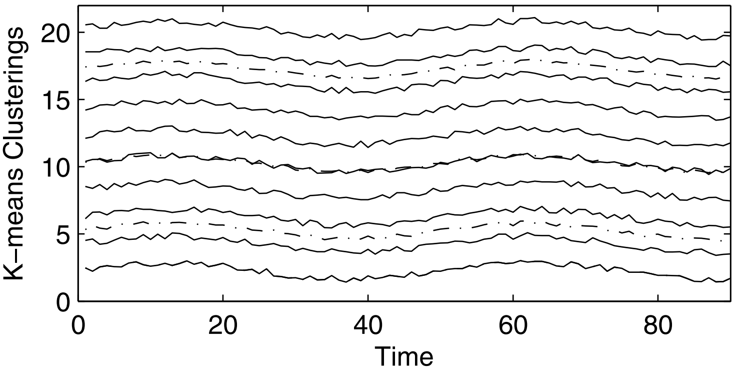


Fig. 2. The K-means clustering of the historical data (dotted line) and the groups of historical data (solid line)

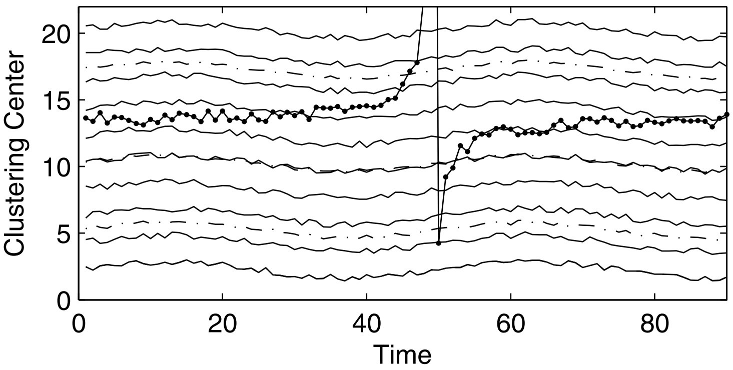


Fig. 3. The clustering center of the historical data (dash-dot line), the K-means clustering of the historical data (dotted line) and the groups of historical data (solid line)

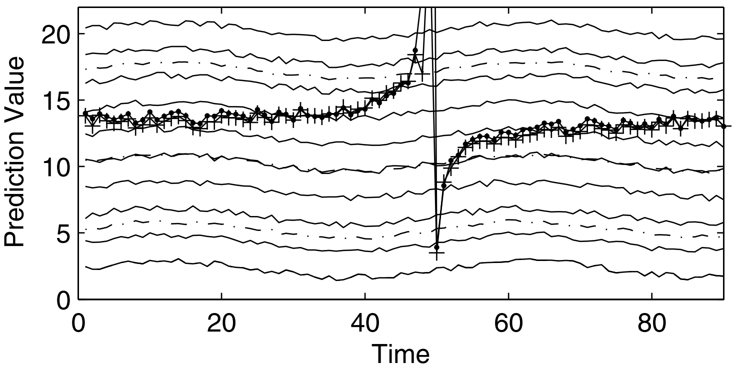


Fig. 4. The trend prediction based on the clustering center (solid line marked with plus sign), the clustering center of the historical data (dash-dot line), the K-means clustering of the historical data (dotted line) and the groups of historical data (solid line)

5 The Predicting Error Analysis

In this section, the predicting error with the method mentioned in the paper is illustrated in the Fig. 5 and Fig. 6, which show that β_1 and β_2 have affects on the errors. When $\beta_1 = 0.2$ and $\beta_2 = 0.8$, the errors from 45th to 55th second are smaller than that when $\beta_1 = 0.8$ and $\beta_2 = 0.2$, because β_1 represents the weight of the K-means clustering centers and β_2 represents the reviser Δ in predicting result. And when $\beta_1 = 0.2$ and $\beta_2 = 0.8$, in the periods from the beginning to the 45th and in the periods from 55th to the ending (90th) second, the errors are larger than that when $\beta_1 = 0.8$ and $\beta_2 = 0.2$, this is because the representation of the measuring data based on the clustering center is more suitable for the data contain low frequency.

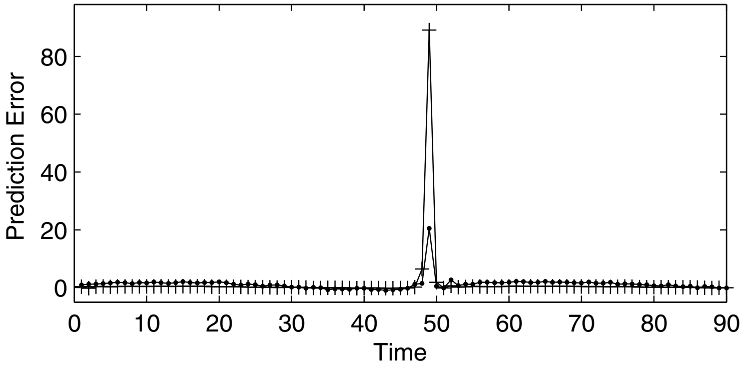


Fig. 5. The prediction error (solid line with plus sign when $\beta_1 = 0.2$, $\beta_2 = 0.8$; dash-dot line when $\beta_1 = 0.8$, $\beta_2 = 0.2$)

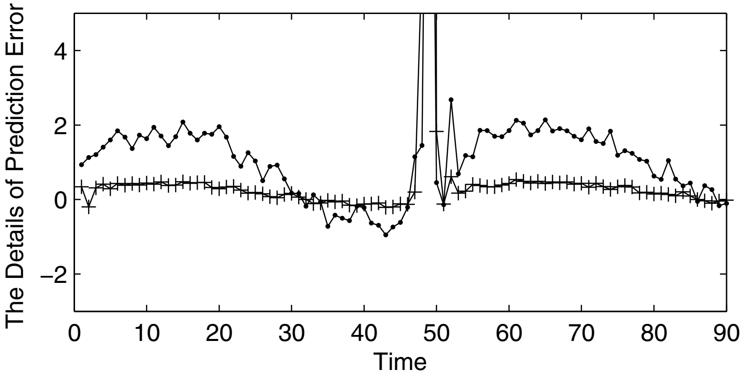


Fig. 6. The details of prediction error (solid line with plus sign when $\beta_1 = 0.2$, $\beta_2 = 0.8$; dash-dot line when $\beta_1 = 0.8$, $\beta_2 = 0.2$)

Fig. 5 indicates that the prediction errors are smaller from 0th-47th and 56th-90th second during which contain more data are of low frequency. Details of Fig. 5 are shown in Fig. 6. In Fig. 6, the dash-dot line represents the errors when $\beta_1 = 0.2$ and $\beta_2 = 0.8$, and the solid line with plus represents that when the $\beta_1 = 0.8$ and $\beta_2 = 0.2$. The dash-dot line indicates larger errors in low frequency and smaller errors in high frequency compared with the the solid line with plus. Obviously, the proposed method is more suitable for trend predicting when most data are of low frequency. In Fig. 7 and Fig. 8, the proposed method is compared with the method of the Radial Basis Function Neural Networks (RBFNNs) in the trend predicting. The difference of predicting error is not significant in the periods from 0th second to 45th second and from 56th to 90th second. And obviously, the predicting error of the proposed method is smaller than that of the RBFNNs.

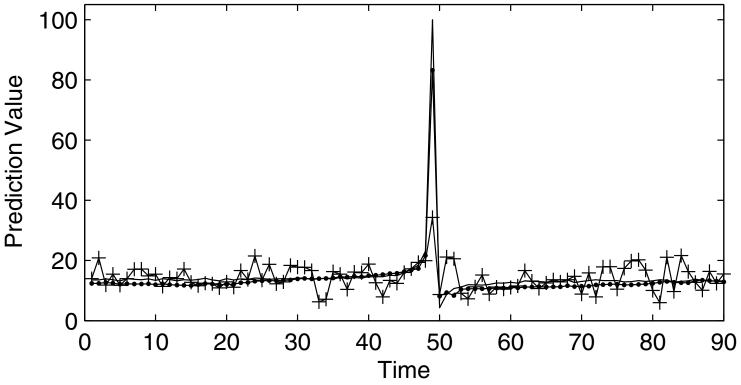


Fig. 7. Prediction value compare with RBF network algorithm

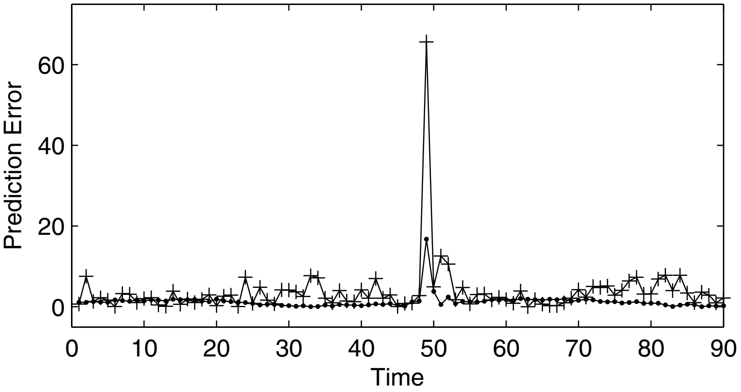


Fig. 8. Prediction error compare with RBF network algorithm

6 Conclusion

This paper introduces a trend predicting method for measuring data based on historical data K-means clustering. The simulation and analysis indicate that the method can predict the trend accurately when most current measuring data are of low frequency. And more efforts should be made to improve the trend prediction performance when more measuring data are of high frequency.

Acknowledgments. This work was supported by the Fundamental Research Funds for the Center Universities of China, Project No. 0215005202020.

References

1. Bong, D., Tan, J., Rigit, A.: Optimization of the backpropagation hidden layer by hybrid k-means-greedy algorithm for time series prediction. In: 2010 IEEE Symposium on Industrial Electronics & Applications (ISIEA), pp. 669–674. IEEE (2010)
2. Huaizhen, Y., Lei, L.: Order prediction of interactive service of cable television based on k-means cluster and decision tree. In: 2010 International Conference on Computer Application and System Modeling (ICCASM), vol. 3, pp. V3–V215. IEEE (2010)
3. Asamer, J., Din, K.: Prediction of velocities on motorways by k-means clustering. In: Seventh Mexican International Conference on Artificial Intelligence, MICAI 2008, pp. 399–403. IEEE (2008)
4. Zheng, L., Yu, M., Yu, S., Wang, W.: Improved prediction of nitrogen oxides using grnn with k-means clustering and eda. In: Fourth International Conference onNatural Computation, ICNC 2008, vol. 2, pp. 91–95. IEEE (2008)
5. Maia, J., et al.: Network traffic prediction using pca and k-means. In: 2010 IEEE Network Operations and Management Symposium (NOMS), pp. 938–941. IEEE (2010)
6. Yiakopoulos, C., Gryllias, K., Antoniadis, I.: Rolling element bearing fault detection in industrial environments based on a k-means clustering approach. Expert Systems with Applications 38(3), 2888–2911 (2011)
7. Awad, M., Pomares, H., Rojas, I., Salameh, O., Hamdon, M.: Prediction of time series using rbf neural networks: A new approach of clustering. The International Arab Journal of Information Technology 6(2) (2009)

Measuring Micro-blogging User Influence Based on User-Tweet Interaction Model

Dong Liu, Quanyuan Wu, and Weihong Han

School of Computer, National University of Defense Technology
410073 Changsha, China
nudtld@yahoo.cn, quanyuan.wu@gmail.com, hanweihong@139.com

Abstract. Measuring micro-blogging user influence is very important both in economic and social fields. In this paper, we propose a user-tweet interaction model to describe the relationships among users and tweets. Considering the time affect, *TAC*(time-effectiveness attenuation coefficient) is proposed when calculating tweet influence which consists of retweet influence and comment influence. Then we make a detail analysis on the generation of user influence which consists of post influence and follow influence based on the results of tweet influences. We also discuss the correlation between post influence and follow influence by use of Spearman's rank correlation coefficient. At last, we rank users by calculating the bias spatial distances. Taking Sina micro-blogging as background, after a series of experiments, we believe that our method is accurate and comprehensive when measuring the influences of micro-blogging users.

Keywords: micro-blogging, user influence, user-tweet interaction model, correlation.

1 Introduction

Nowadays, micro-blogging has developed into one of the most influential media platforms of Internet. By the end of November 2011, the number of Chinese micro-blogging accounts has grown to 320 million and the tweets' number has reached over 150 million per day [1]. Therefore, many researchers began to do data mining and analysis on micro-blogging, and mining micro-blogging user influence has become a hot research topic.

User influence generally means the interaction among users. Some researchers use the concept of *social influence* which indicates people modifying their behaviors to bring them closer to the behaviors of their friends [2]. Measuring user influence is very important in both economic and social fields. In business recommendations, others would receive the advertisements much more quirkily and efficiently if we recommend the goods and services to the influential users. In addition, the joining of influential users brings the explosive propagation of topics. Therefore, the analysis of user influence is much useful and essential to mine the propagation rules of topics in micro-blogging and acquire the behavior features of the key users, and it has broad application prospect in public sentiment monitoring and analyzing.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 146–153, 2013.

In this paper, a user-tweet interaction model is proposed to describe the relationships of micro-blogging users and tweets. The user influence is analyzed detailedly based on the model. Taking Sina micro-blogging which is the largest micro-blogging platform in China as background, a series of experiments are done to prove the effectiveness of our method.

2 Relate Work

Hyperlink analyses, learning from PageRank [3] algorithm, are widely used when analyzing the micro-blogging user influence. Tunkelang [4] proposed TunkRank which measured the influences of users in Twitter based on a user graph constructed according to following relationships. Similarly, Weng et al. [5] proposed the algorithm TwitterRank which measures the Twitter user's influence considering the link structure of follow relationships and the number of tweets. Liu [6] proposed UserRank to measure user influence in social network based on PageRank. These methods mainly consider the following relationships among micro-blogging users.

The users' actions play an important role in the analysis of user influence as well, therefore some researchers made comprehensive analyses considering these factors. Kong [7] proposed a tweet-centric approach to rank topic-specific influential authors in micro-blogging. Yuto Yamaguchi [8] focused on the post/posted, follow/followed, retweet/retweeted actions of micro-blogging users, and proposed TURank which evaluate users' authority scores in Twitter by use of user-tweet graph based on ObjectRank [9]. Meeyoung [10] focused on three actions: follow, retweet and mention, and then analyzed the influences represented by the three actions. At last, he sorted the influence scores by use of Spearman's Rank Correlation Coefficient [11].

Most of these former researches take no account of time affect which may cause inaccurate results, since a tweet's influence may change as time goes by. Therefore a user-tweet interaction model considering time affect is proposed. Based this model, a more comprehensive analysis of user influence is made in this paper.

3 A User-Tweet Interaction Model

To construct the user-tweet interaction model, firstly we define a user-tweet schema graph as shown in Fig. 1 similar to the graph in [8].

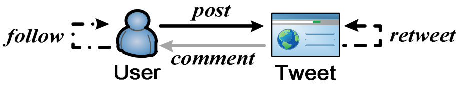


Fig. 1. A user-tweet schema graph

A user-tweet schema graph is a directed graph $G = (V, E)$. V is the set of nodes consisting of user nodes and tweet nodes. E is the set of edges consisting of *follow*, *post*, *retweet* and *comment* edges. A follow edge is from a user u to another user who follows *u*. A post edge is from a user *u* to a tweet *t* posted by *u*. A retweet edge is from a tweet *t* to another tweet which retweets *t*. A comment edge is from a tweet *t* to a user *u* who comments *t*. The direction of arrow represents the direction of influence flow. For example, a micro-blogging user is followed by others means that he influence his followers; a tweet is retweeted by another tweet means it influences that tweet.

A user-tweet interaction model is show in Fig 2. In the case of user-tweet interaction model, it consists of user nodes, tweets nodes and four kinds of edges. The *time range* is the range of posted time of tweets. The tweets whose time is out of range are not considered in influence generation.

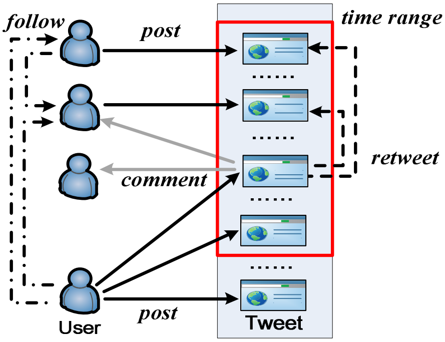


Fig. 2. The user-tweet interaction model

The influence of tweet node *t* called *tweet influence* is calculated from its out-degree which is the number of comment edges and retweet edges directed from *t*, therefore the tweet influence consists of *retweet influence* and *comment influence* which are separately calculated from the two kinds of edges. Considering the time affect, the tweet influence of a tweet reduced as time goes by.

The influence of user node *u* called *user influence* consists of *follow influence* and *post influence*. The follow influence is calculated from the number of follow edges directed from *u*. The post influence represents the influences of tweets posted by *u*, and it is calculated from the sum of tweet influences.

4 Influence Generation Based on User-Tweet Interaction Model

4.1 Generation of Tweet Influence

As described in section 3.2, the tweet influence is calculated from retweet influence and comment influence.

Retweet influence of tweet node *t* is calculated from the number of retweet edges directed from *t*. It is defined as follow.

$$retweet\_influence(t)=\alpha\times OutDegree\_retweet(t)\ . \tag{1}$$

OutDegree_retweet(*t*) is the number of retweet edges directed from the tweet node *t*. $\alpha \in (0, 1]$. It is adjustable and indicates the weight of retweet edge.

The same is comment influence. It is defined as follow.

$$comment\_influence(t)=\beta\times OutDegree\_comment(t)\ . \tag{2}$$

OutDegree_comment(*t*) is the number of comment edges directed from the tweet node *t*. $\beta \in (0, 1]$ and it indicates the weight of comment edge. According to the experience, β is bigger than α, and it means that the users who comment on tweet *t* are more interested in it than others who only retweet it.

It is obviously that micro-blogging users mainly focus on the current tweets. Therefore, the recent tweets have bigger influences than the former ones. In view of this, *TAC*(time-effectiveness attenuation coefficient) is proposed to represent the attenuation degree of tweet influence as time goes by. It is defined as follow.

$$TAC=\frac{1}{1+2^{(post\_time(t)-t_{now})}}\ . \tag{3}$$

post_time(*t*) is the posted time of tweet *t*, and t_{now} is the current time. The two time variables could be the time granularity such as 1 day which is pre-defined before calculation. The nearer to the current time, the bigger influence a tweet has.

Finally, the tweet influence of tweet node *t* is defined as follow:

$$tweet\_influence(t)=TAC\times(retweet\_influence(t)+comment\_influence(t))\ . \tag{4}$$

4.2 Generation of User Influence

As described in section 3.2, the user influence consists of follow influence and post influence.

The follow influence is defined as follow:

$$follow\_influence(u)=\gamma\times OutDegree\_follow(u)\ . \tag{5}$$

OutDegree_follow(*u*) is the number of follow edges directed from user node *u*. $\gamma\in(0, 1]$ and it indicates the weight of follow edge.

The post edge is directed from user node to tweet node, and it represents the authorship of the tweets. Obviously, a user who posts lots of influential tweets has big post influence. The post influence is defined as follow. V_T is the node set of tweets posted by *u* during the time range.

$$post\_influence(u)=\sum_{t\in V_T} tweet\_influence(t)\,. \tag{6}$$

Finally, the user influence is defined to be a binary vector as follow.

$$user\_influence(u)=(\ post\_influence(u),\ \ follow\_influence(\mathrm{u})) \tag{7}$$

In addition, we sometimes need to mine the influence reflected by single tweet. It means that the post influence sometime need to be normalized, the same is true of user influence. The normalized user influence is defined as follow. *tweet_num* is the total number of tweets posted by user *u* during the time range.

$$normalized\_user\_influence(u) = (normalized\_post\_influence(u),\ follow\_influence(u))$$
$$= (\frac{post\_influence(u)}{tweet\_num}, follow\_influence(u)). \quad (8)$$

4.3 Measuring Correlation between Post Influence and Follow Influence

In micro-blogging, the tweets of a user who has more followers always draw more attentions, so there evidently exists correlation between the post influence and follow influence. Here, we try to mine the correlation between the two kinds of influences.

Rather than use the values of post influence and follow influence directly, we use the relative order of influence's rank as a measure. In order to do this, we sorted users by each kinds of influence, so that the rank of 1 indicates the most influential user and increasing rank indicates a less influential user. Users with the same influence value receive the same rank.

Assume that there are n micro-blogging users, we generate the post influence and follow influence of each u_i ($i \in [1, n]$) at first. $post\_influence(u_i)$ represents the post influence of u_i, the same are true of $normalized\_post\_influence(u_i)$ and $follow\_influence(u_i)$. Then, we sort users by the three kinds of influences separately. The raw scores $post\_influence(u_i)$, $normalized\_post\_influence(u_i)$, $follow\_influence(u_i)$ are finally represent as ranks $Rank\_p(u_i)$, $Rank\_n(u_i)$, $Rank\_f(u_i)$.

Since the sample space is n, we finally get three n-dimensional vectors: $Rank\_post$=($Rank\_p(u_1)$,$Rank\_p(u_2)$,…, $Rank\_p(u_n)$), $Rank\_normalized$=($Rank\_n(u_1)$, $Rank\_n(u_2)$,…,$Rank\_n(u_n)$) and $Rank\_follow$=($Rank\_f(u_1)$, $Rank\_f(u_2)$,…, $Rank\_f(u_n)$).

We used Spearman's rank correlation coefficient

$$\rho = 1 - \frac{6 \times \sum (x_i - y_i)^2}{n(n^2 - 1)}. \quad (9)$$

as a measure of the strength of the association between vectors $Rank\_post$/ $Rank\_normalized$ and $Rank\_follow$, where x_i is $Rank\_p(u_i)$ or $Rank\_n(u_i)$ and y_i is $Rank\_f(u_i)$. Spearman's rank correlation coefficient is a nonparametric measure of statistical dependence between two variables. It assesses how well the relationship between two variables can be described using a monotonic function. If there are no repeated data values, the closer ρ is to +1 or -1, the stronger the likely correlation. A perfect positive correlation is +1 and a perfect negative correlation is -1.

4.4 Ranking of User Influence

Sometimes, we need to sort micro-blogging users by their influences. The ranking only according to post influence or follow influence would be biased. Therefore, considering the correlation between post influence and follow influence, the bias spatial distance is adopted to measure the distance between user influences and sort the users compositively.

Developed from the previous section, user influence and normalized user influence are finally represented by the binary vectors as follows:

$user\_influence\_rank(u_i)$=($Rank\_p(u_i)$, $Rank\_f(u_i)$) .

$normalized\_user\_influence\_rank(u_i)$=($Rank\_n(u_i)$, $Rank\_f(u_i)$) .

The bias spatial distance between vectors x_i=(x_{i1},x_{i2},…,x_{im}) and x_j=(x_{j1},x_{j2},…,x_{jm}) is calculated as follow. r_{kl} is the correlation coefficient between x'_k =(x_{1k},…,x_{ik}, x_{jk},…,x_{nk}) and x'_l =(x_{1l},…,x_{il}, x_{jl},…,x_{nl}), n is the sample size. Different from Euclidean distance, bias spatial distance considers the correlations among the variables and it is also scale-invariant.

$$d_{ij} = \left[\frac{1}{m^2} \sum_{k=1}^{m} \sum_{l=1}^{m} \left(x_{ik} - x_{jk} \right) \left(x_{il} - x_{jl} \right) r_{kl} \right]^{1/2} . \tag{10}$$

We first define *Optimum_rank*=(1,1), and then the bias spatial distances between *user_influence_rank*(u_i)/*normalized_user_influence_rank*(u_i) and *Optimum_rank* is calculated. The smaller the distance, the bigger the influence. In the calculation, x_i is *Optimum_rank*, x_j is *user_influence_rank*(u_i) or *normalized_user_influence_ rank*(u_i), and r_{kl} is the Spearman's rank correlation coefficient ρ between *Rank_post*, *Rank_normalized* and *Rank_follow*.

5 Experiments

In order to ensure that the analyzed users are representative, we randomly choose 100 friends from Kai-fu Lee who is very popular in Sina micro-blogging. Firstly, by use of OpenAPI [12], we collected their follower information as well as all their tweet data from October to December in 2011 including the contents, comments and retweet information. After generating and analyzing their tweet influences and user influences, we rank the 100 users by their composite influences at last.

5.1 Analysis of User Influence

First of all, the parameters are defined: α←0.8, β←1, γ←1 and the time granularity of *post_time*(t) is defined to be 1 day. Then the user influence is generated by use of the formula 1 to 8, as described in section 4. Limited by the space, the result of 5 users is shown in Table 1.

Table 1. The influence of 5 users

User	*post_influence*(u)	*normalized_post _influence*(u)	*follow_influence*(u)
Chenggang Rui	200589.73	818.73	6098048
Caijing	2062100.49	425.53	4785643
Daqing Mao	6415.86	16.75	114887
Ciqun Bei	790.14	0.87	22158
DFdaily	48566.11	42.12	916454

Take "Chenggang Rui/a television presenter" and "Caijing/a finance website" for example. The post influence of Caijing is much bigger than that of Chenggang Rui, because there are more users who retweet or comment on Caijing's tweets. However, its normalized post influence is smaller. The reason is that Caijing posted 4846 tweets in the three months but Chenggang Rui only post 245 tweets. So the influence of Chenggang Rui is much bigger for singer tweet.

5.2 Correlation between Post Influence and Follow Influence

Firstly we sort the users according to their three kinds of influences. Limited by the space, the result of 5 users is shown in Table 2.

Table 2. Ranks of influences of 5 users

User	*Rank_p(u)*	*Rank_n(u)*	*Rank_f(u)*
Chenggang Rui	16	5	5
Caijing	1	8	9
Daqing Mao	69	61	66
Ciqun Bei	81	83	79
DFdaily	37	39	31

The formula 9 is used to calculate the correlation, and the final results are shown as follows.

$$\rho\_total = 0.7608, \rho\_normalized = 0.8060.$$

ρ_total is the correlation coefficient between post influence and follow influence and $\rho\_normalized$ is the correlation coefficient between normalized post influence and follow influence. ρ_total and $\rho\_normalized$ both $\in [0,1]$ which means that the post influence is positive correlated with the follow influence. Once the number of followers increases, the tweets draw more people's attention than before, and vice versa.

$\rho\_total < \rho\_normalized$ means that the follow influence influences normalized post influence more than post influence, and it also indicates that the new posted interesting tweet rather than all the tweets may attract other users' attentions.

In summary, if a user wants to improve his influence, he should post attractive tweets or increase the number of followers in all ways.

5.3 Ranking of User Influence

After the calculation as described in section 4.4, we finally get a ranking of the 100 users and the ranks of the 5 users are shown in Table 3. *distance_t* indicates the bias spatial distance between *user_influence_rank*(u_i) and *Optimum_rank*, *distance_n* indicates the distance between *normalized_user_influence_rank*(u_i) and *Optimum_rank.* According to the characteristics of bias spatial distance, the ranking considers both post influence and follow influence as well as the correlation between them, and therefore it could reflects the real influences of micro-blogging users.

Table 3. Ranks of 5 users

User	*distance_t*	*rank*	*distance_n*	*rank*
Chenggang Rui	8.2861	9	3.5911	2
Caijing	3.4889	2	6.7334	8
Daqing Mao	58.0029	68	56.1115	66
Ciqun Bei	68.9058	81	71.8227	82
DFdaily	28.7834	33	30.5246	31

6 Conclusion

A user-tweet interaction model is proposed to describe the relationships among users and tweets. Considering the time affect, *TAC*(time-effectiveness attenuation coefficient) is proposed when calculating tweet influence and user influence. We also discuss the correlation between post influence and follow influence by use of Spearman's rank correlation coefficient. At last, we rank users by calculating the bias spatial distances which proved to be comprehensive by experiments.

The method of measuring user influence proposed in this paper is for general. Depending on different applications, such as mining the influences of users who have same topics, the method needs to be modified.

Acknowledgments. The authors are grateful to the anonymous referees for a careful checking and for helpful comments that improved this paper. This work was supported by "863" Program (No. 2010AA012505, 2011AA010702, 2012AA01A401、2012AA01A402) and NSFC (No. 60933005).

References

1. The reduction of micro-blogging users' activity and stability, http://whb.news365.com.cn/jkw/201112/t20111221_3209369.htm
2. Yu, L., et al.: What Trends in Chinese Social Media. In: 5th SNA-KDD Workshop 2011, San Diego, pp. 2–4 (2011)
3. Lawrence, P., Sergey, B., et al.: The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Stanford Digital Library Technologies Project (1998)
4. Tunkelang, D.: A twitter analog to pagerank, http://thenoisychannel.com/2009/01/13/
5. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, pp. 261–270. ACM (2010)
6. Liu, Y.: Research on social network structure. ZheJiang University (2008)
7. Kong, S., Feng, L.: Tweet-Centric Approach for Topic-Specific Author Ranking in Micro-Blog. In: Tang, J., King, I., Chen, L., Wang, J. (eds.) ADMA 2011, Part I. LNCS, vol. 7120, pp. 138–151. Springer, Heidelberg (2011)
8. Yamaguchi, Y., Takahashi, T., Amagasa, T., Kitagawa, H.: TURank: Twitter User Ranking Based on User-Tweet Graph Analysis. In: Chen, L., Triantafillou, P., Suel, T. (eds.) WISE 2010. LNCS, vol. 6488, pp. 240–253. Springer, Heidelberg (2010)
9. Balmin, A., Hristidis, V., Papakonstantinou, Y.: Objectrank: Authority-based keyword search in databases. In: 30th International Conference on Very Large Data Bases, vol. 30, pp. 564–575. Morgan Kaufmann (2004)
10. Cha, M., et al.: Measuring User Influence in Twitter: The Million Follower Fallacy. In: 4th international AAAI Conference on Weblogs and Social, Washington, DC, pp. 11–13 (2010)
11. Zar, J.H.: Significance Testing of the Spearman Rank Correlation Coefficient. Journal of the American Statistical 30, 578–581 (1972)
12. OpenAPI of Sina micro-blog, http://open.weibo.com/

Discover Community Leader in Social Network with PageRank

Rui Wang, Weilai Zhang, Han Deng, Nanli Wang, Qing Miao, and Xinchao Zhao

School of Science,
Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract. Community leaders are individuals who have huge influence on social network communities. Discovering community leaders in social networks is of great significance for research on the structures of the social networks and for commercial application. Based on the core idea of the PageRank algorithm, this paper firstly processes data selected from *Sina* microblog, and extracts three key indicators, comprising the number of followers, the number of comments and the number of reposts; then based on their mutual relationship, that is following or followed, it obtains the weight of influence for each individual user; and then after a finite number of iterations, this paper identifies the community leader in *Sina* microblog, by which its comprehensive influence on its community are reflected.

Keywords: social network, PageRank algorithm, community leader.

1 Introduction

In modern society, social networking, via social media, proliferates in a quickly changing world [1]. Social networking has been entrenched in people's daily lives for a long time by various approaches, such as social networking websites, mobile phone client, etc [2]. When more and more people prefer to obtain information through the social network, some very active community leaders who have huge impacts on the users in the community become evident in social networks [3]. Marketing influence evaluation of microblog is not only limited within a particular blog [4]. Through assessment between individual social networks, assessment on the integral effect of microblog marketing is then obtained.

Therefore, the problem of how to appropriately spot the community leader in a vigorous community among the huge users group becomes an important issue. In the simplified part of social network community structure as shown in Figure 1, each node represents an individual, and the social network has crowd segmentation [5]. Some users are central in the community, some users are on the edges, establishing fewer contracts with other users, and hence have lower influence.

In this paper, we present a new community discovery approach based on PageRank algorithm to find these "important" users in a social network. Different from previous methods, we exploit prior knowledge of the given network, such as the mutual

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 154–162, 2013.

relationship between top 100 users in a community and the extent to which users are related in the data. In this paper three indicators are required to balance each other in order to obtain the algorithm for community leader.

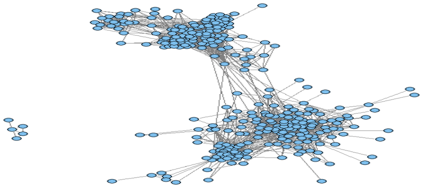


Fig. 1. Virtual Community Structure

In order to spot the ranking of a website without knowledge of its initial value, the PageRank algorithm changes this question into a matrix multiplication and assumes that the initial PR values of all websites is the same [6]. Firstly, calculate the first iterative ranking of each website based on the initial PR value. And then calculate the second rank according to the first iteration. After experiments, the PageRank estimator converges to its practical value is theoretically proved no matter what the initial value is. The whole processing is implemented without manual intervention.

As far as social network is concerned, this paper derives an algorithm from the PageRank algorithm in that the mutual relationship in the social networking can be structured as links to microblogs [7]; users can also be regarded as websites in the PageRank algorithm. Slightly and relevantly, this paper makes a bit change to the PageRank algorithm, and then calculates users' connection in the specific social network, Sina microblog, at last exploits its community leader.

2 Experiment and Data Analysis

2.1 Obtain User Data in Social Network

This paper conducts an experiment to obtain user data of Sina microblog. As social networks open up their APIs in mid-2007, it is then necessary to invoke the opened API interface for grasping data on the internet. After applying for setting application up on the application platform in Sina microblog, each authorization can invoke an API when the application is approved, and API can then be sent, applying for grabbing data required by this paper. By inserting grabbed data to API source code and compiling new program, data is successfully obtained by functioning application. In addition, by searching, screening and collecting Internet information, data obtaining or grabbing is a process which depends on the insert code; the data obtained directly comes from web page, which can be seen by accessing users' microblogs, thus there is no private information involved throughout the whole experiment.

In the experiment, the target people are top 100 users of cultural field whose activity is rather high in Sina microblog. The data grabbed is their number of followers, mutual relationship, e.g. follow or followed, which is as briefly shown in Table 1, 200 micro blogs before 22:09 on October 10, 2012(in which some of microblogs were deleted and some are less than 200), number of reposts and comments, which is as briefly shown in Table 2. To strengthen the precision of our experiment, more indicators will be added in later studies, such as number of collections, keyword querying and so on.

Table 1. User-Follower

User ID	Follower ID
2240194360	1816011541
1195883527	1672272373
1225419417	1601563722
....	…………
1672634524	1188552450
1645826702	1665372775
1601563722	1665372775

Table 2. The number of reports and comments of users

user ID	numbe r of reposts	number of comments	time
1816011541	3700	1275	2012/10/13 8:58
1816011541	12860	3335	2012/10/12 9:05
1816011541	35952	6794	2012/10/11 8:46
1816011541	2820	881	2012/10/11 8:03
1816011541	9188	2583	2012/10/10 8:55
1816011541	4001	1152	2012/10/10 8:32
1816011541	176	375	2012/10/10 8:03

2.2 Data Processing and Algorithm Optimization

After accessing the number of comments, reposts and followers of each user, next step is to process the data by means of the PageRank algorithm and then calculate each user's PR value. Three metrics (precision, recall and F-measure) are used in our experiments for evaluation.

2.2.1 Data Processing: Number of Followers

According to the situation in the real social network relationship, this paper categorizes relationships into two following cases: When the number of followers increases by 1, the user was deemed to be linked once; and after following 1 another user, the user will also be deemed to link once with another user. So the mutual relationship matrix, i.e. follow or followed, in the target social network community, Sina microblog, can be produced as briefly shown in Table 3.

Parameters in adjacent matrix a_{ij} are defined as:

$$\begin{cases} \mathrm{a_{ij}} = 1(\text{if the ith user follows the jth user}) \; . \\ \mathrm{a_{ij}} = 0(\text{if the ith user does not follow the jth user}) \; . \end{cases} \tag{1}$$

If N is used to express user number, then the adjacent matrix is with N rows and N columns. Transpose the adjacent ranks, and then divide the column vector by the number of their respective number of followers. That the sum of each column vector turns out 1 means this user follows all other users in the top 100 of the specific social network community. Thus the resulting matrix becomes transition probability matrix, by which we can analyze the extent where the user is relevant to its community.

Mathematically, the PageRank algorithm is to resolve the eigenvector of the transition probability matrix when it reaches the maximum of eigenvalue. This eigenvalue is PR value of the user. By the following two commands in Matlab software, we can respectively get the probability transposed matrix of M and the unitized M matrix. And the matrix of eigenvalue of each user is as shown in Table 4.

$$M = A * inv\left(diag\left(sum\left(A'\right)\right)\right) . \tag{2}$$

$$\left[V , lam\right] = eig\left(M\right) . \tag{3}$$

p.s. "%PageRank" is the value after unitizing $V(:,1)$, which is used when display matrix M, the matrix of eigenvalue.

Table 3. Adjacent matrix, A

	1	2	3	4	5	...	99	100
1	0	0	1	0	1	...	0	0
2	0	0	0	0	0	...	0	0
3	0	1	0	0	1	...	0	0
4	0	0	1	0	0	...	0	1
5	0	0	1	0	0	...	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
98	0	0	0	0	0	...	0	0
99	0	0	0	1	0	...	0	0
100	0	0	0	0	0	...	0	0

Table 4. Matrix of eigenvalue, M

No	User ID	Importance
1	1816011541	0.0063
2	1672272373	0.0059
3	1601563722	0.0195
4	1188552450	0.0135
5	1665372775	0.0114
.	.	.
.	.	.
99	1444865141	0.0292
100	1198367585	0.0025

The results reflect that PR value is hardly related to positive-directional following. However, the number of followers does fundamentally decide users' PR values in the social network community. As mentioned, the PageRank algorithm reflects the degree to which a website (user) is linked (followed), which means when the number of followers is the only factor to be taken into consideration of calculation, PR value directly reflects the user's influence in this community.

2.2.2 Algorithm Optimization

In the adjacency matrix, 1 is used to present that the user is followed by one user. However, users with different weights can have different influence on other users' PR value when following others in social network. Users with higher weight have higher influence, while users with lower weight get lower influence. In order to deal with the distinct influences users with different weights can make, this paper adopts the secondary, even multiple calculations to optimize the PageRank algorithm.

To optimize the current algorithm, our first thing is to substitute the first eigenvalue of each user in adjacency matrix, in which the first eigenvalue indicates each user's weight in the specific social network community. As far as the second adjacency matrix is concerned, assume user A follows user B, then a_{ax} indicates the PR value for user A after the first calculation, and a_{bx} is the first PR value of user B. Based on the calculation done in data processing, the second adjacency matrix, the transition probability matrix, and the second eigenvalue can be computed one by one. Then substitute the second eigenvalue for user weight and calculate the third eigenvalue; iterate the loop of calculation until the eigenvalue finally tends to be constant; and the final constant value is importance ranking of the first indicator, number of followers. Thus, the influence of weights of these different users upon our importance ranking is also under settlement.

2.3 Weight Distribution of Different Indicators

As far as each user is concerned, this paper sets three types of variable, which are number of followers, comments and reposts of one's recent 200 microblogs, and numbers users grabbed as x, which corresponds to their *Sina* ID, then assumes the set of average number of reposts is:

$$R(\text{Reposts}) = R\left\{R_1, R_2, \dots, R_n\right\} . \tag{4}$$

Set of average number of comments is:

$$C(\text{Comments}) = C\left\{C_1, C_2, \dots, C_n\right\} . \tag{5}$$

Set of average number of followers is:

$$F(\text{Followers}) = F\left\{F_1, F_2, \dots, F_n\right\} . \tag{6}$$

And the set every user's PR value is:

$$P = P\left\{P_1, P_2, \dots, P_n\right\} . \tag{7}$$

p.s. The temporary test number of users is 100, so $n = 100$.

The higher importance of a user who forwards a microblog, the larger scope and effect of importance one microblog will gain. Even though the number of users who have seen one microblog cannot be added up linearly, it can be substantiated that more followers of users do have browsed this microblog, which also means its influence will grow higher. Therefore, by multiplying user's PR value with their average of number of comments, $E\left(effect\right)$ can be obtained as follows, in which E to some degree quantifies the effect of one microblog:

$$E\left(x\right) = P\left(x\right) * A\left(x\right) . \tag{8}$$

But a microblog labeled with high effect does not suggest that it has high quality, neither does it suggests that the user who sends microblog of high effect is the community leader this paper is researching for; the content of the microblog should be taken into consideration. Instead of defining the quality of a microblog in a simple approach, we combine number of comments, reposts and followers to reflect the quality of a microblog, since the three factors can suggest this microblog has aroused resonance and interest among other users to a certain degree. Assume the quality function $Q\ (x)$ is a linear function associated with the number of followers, number of comments and number of reposts, thus:

$$Q\ (x) = h * R\ (x) + jC\ (x) + k * F\ (x)\ . \tag{9}$$

In order to facilitate further operation, after surveying the relationship among all kinds of magnitudes, all variables are modified to reach the same order of magnitude as shown in Table 5.

Table 5. Order of magnitude for number of repost and comments & for number of followers

PR value	Number of reposts and comments
10	>10000
9	>8000
8	>5000
7	>3000
6	>1000
5	>800
4	>500
3	>300
2	>100
1	>50

PR value	Fans number
10	>20000000
9	>10000000
8	>8000000
7	>5000000
6	>3000000
5	>1000000
4	>500000
3	>100000
2	>50000
1	>10000

Unknown coefficients h, j, k can be determined through questionnaire survey and simulated experiment of virtual community. During questionnaire survey, 1500 people were sampled to answer the question: which can better reflect the content quality of microblog, the number of comments, reposts or followers? Survey results turns out 682 people supports number of comments, 463 people are in favor of “the number of reposts” and 355 people are with “the number of followers”, which is shown in Figure 2. Combine the data above with the importance level about different users set up in virtual community, corresponding adjustment to the distribution of weight of three indicators is then achieved, which are 0.469, 0.306 and 0.225 respectively for h, j and k.

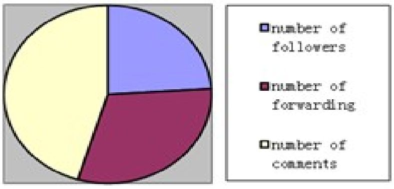


Fig. 2. Distribution of researched weights between three indicators

Since there are large differences between the order of magnitude of $R(x)$, $C(x)$, $F(x)$, then divide $R(x)$, $C(x)$ into 10 grades (L) according to uniform distributed function and evaluate $R(x)$, $C(x)$ and $F(x)$ according to current rank form, then substitute the grade value L into the function and obtain :

$$Q(x) = 0.469 * L * R(x) + 0.306 * L * C(x) + 0.255 * L * F(x) \quad (10)$$

Having defined effect function $E(x)$ and quality function $Q(x)$ for micro blogs, then comprehensive value function is obtained:

$$V(x) = E(x) * Q(x) \, . \quad (11)$$

Then this paper calculates the degree of user activity in Sina microblog, also the user's frequency of posting a microblog, which is $200/t$ in this test. At the very end, by combining the degree of user activity with comprehensive value function, we can spot community leader in this social network:

$$T(x) = E(x) * Q(x) * \left(\frac{200}{t} \right) \, . \quad (12)$$

And the data processing is as briefly shown in Table 6.

Table 6. Table of the overall value

User ID	Weight	E(x)	Q(X)	V(x)	t	T(x)
1601563722	0.0195	109.45	7.16	783.97	34.67	22.61
1191258123	0.002	165.01	9.55	1575.82	191.99	8.21
1182415487	0.0359	132.59	6.47	857.72	138.44	6.20
1454884585	0.0169	17.66	4.63	81.81	14.42	5.67
1195031270	0.0135	9.70	4.37	42.39	10.49	4.04
1665372775	0.0114	31.01	5.24	162.60	50.54	3.22
1188552450	0.0135	93.52	7.92	740.57	256.49	2.89
1444865141	0.0292	17.92	3.39	60.72	26.42	2.30
1182419921	0.027	22.09	14.02	309.67	291.63	1.06
1217330363	0.0208	4.04	2.14	8.66	10.34	0.84
1299532580	0.0068	13.80	5.31	73.24	184.02	0.40
1096536995	0.0119	10.97	3.78	41.41	105.68	0.39

2.4 Experimental Result Analysis

Analysis of the experiment is deeply discussed as followed:

- Users which have great comprehensive influence in microblogs may not gain importance in a specific community. This research focuses on a specific community rather than the overall situation.
- Users which have a large number of followers may not have a great influence. Hype phenomenon occurs here and there in today's social network, and zombie fans strongly influence people's audio-visual effect. Thus the number of followers might not be objective. This algorithm rules out this interference by multiple times of calculation and balancing these three indicators related to the importance of a user.
- Any analysis model and method has its own limitation [8]. On one hand, since differences exist in statistical data between what this paper processes and the real-time data, which is data in a day, a month, or in a few months, differences between experimental results and official data can be allowed in margins of error. On the other hand, when it comes to algorithm processing and algorithm research, there are also differences between this research and official research in Sina microblog, thus slight differences among experimental results and official results are also allowed.

3 Conclusion

This paper proposes a new algorithm for discovering the community leader in social network community. This algorithm uses Page Rank algorithm to deal user data acquiring from the social network, then optimizing data, and finally rank the user importance in a specific community. What's more, this algorithm ruled out the interference of zombie fans and "online water army", based on the fact that the PR value of these extremely low-influence users is almost zero during calculation.

In addition, under the dramatic development of social network and Internet technology, commercial value can become unimaginable when these resources such as community leader are well utilized. This algorithm for discovering the community leader is of great commercial value, which should be well investigated and excavated.

Acknowledgements. This research is partially supported by Research Innovation Fund for College Students of BUPT and National Natural Science Foundation of China (61105127).

References

1. Zhang, K., Zubcsek, P.: Community Leaders or Entertainment Works Incentivizing Content Generation in Social Media. In: Working Paper of the Business School for the World (2011)
2. Scott, J.: Social Network Analysis: A Handbook. Sage Publications, London (2000)

3. Khorasgani, R.R., Chen, J., Zaïane, O.R.: Top Leaders Community Detection Approach in Information Networks. In: KDD 2010, pp. 1–9 (2010)
4. Wu, J.: Data analysis application in microblog. Marketing (2012)
5. Chen, Y.B.: Social network analysis: Exploring RenRen friend recommend system (2011)
6. Langville, A.N., Meyer, C.D.: Google's PageRank and Beyond: The Science of Search Engine Rankings. Princeton University Press (2006)
7. Abbassi, Z., Minokni, V.S.: A recommender system based on local random walks and spectral methods. In: 9th Web KDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis, pp. 102–108. ACM, New York (2007)
8. Wiegand, T., Sullivan, G., Reichel, J., et al.: Scalable Video Coding-joint Draft 6, ISO/IEC JTCI/SC29/WG11&ITU-T SG16 Q.6, JVT-S201 (2006)

Forecasting Chinese GDP with Mixed Frequency Data Set: A Generalized Lasso Granger Method

Zhe Gao, Jianjun Yang, and Shaohua Tan

Department of Intelligence Science, Center for Information Science, Room 2314,
Science Building 2, Peking University, Beijing 100871, China
truman.gz@gmail.com

Abstract. In this paper, we introduce an effective machine learning method which can capture the temporal causal structures between irregular time series to forecast China GDP growth rate with Mixed Frequency data set. The introduced method first generalized the inner product operator via kernels so that regression-based temporal casual models can be applicable to irregular time series, then the temporal casual relationships among the irregular time series are studied by Generalized Lasso Granger (GLG) graphical models. The main advantage of this approach is that it does not directly estimate the values of missing data of low frequency time series or has restricted assumptions about the generation process of the time series. By applying this method to a 17 macroeconomic indicators GLG model, the forecasting accuracy is better than the autoregressive (AR) benchmark model and a widely used mixed-data sampling (MIDAS) model.

Keywords: Forecast GDP growth, Mixed frequency data, Generalized Lasso Granger.

1 Introduction

As a key indicator of real economic activities, gross domestic product (GDP) growth rate can help policy-makers and analysts continually assessing the state of the economy. However, in most countries, only quarterly GDP data are available. What is worse, the significant delay of quarterly GDP data weakens their role in short-term policy decision making. Therefore, forecasting current quarter GDP, the so-called nowcast, or making forecasts at longer horizons become a very important question in Macroeconomic area. One significant drawback of many traditional forecasting models, e.g. AR, VAR, is the strong assumption that all variables must be of the same frequency. It means all indictors must be quarterly. In fact, the exploration of a large number of indicators at varied monthly frequency may product fruitful results.

Previously, economists deal with the mixed forecast model by sampling all the variables in the model at the same frequency. There are two ways. On one hand, people can convert monthly available data to the quarterly frequency. For instance, in [1], the data of monthly indictors are converted into quarterly data

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 163–172, 2013.

by averaging the months. In [2], it used the data of last month in the quarter, and information on the first month (or first two months) of the quarter being forecast is discarded. Obviously, these averaging or alternative methods may ignore some important information which is implied in the high-frequency data, and also obliterate the high-frequency fluctuations in the monthly data. On the other hand, people can convert the quarterly data to monthly frequency by using the interpolation method. This method has been a standalone research topic for many years [3] [4], However, it is rarely used in the practical prediction of GDP growth rate, as it is generally agreed that the effect of the high-frequency data constructed by the interpolation method is hard to quantify.

Later, the main stream efforts shifted towards to the mixed-data sampling (MIDAS) approach proposed by [5]. MIDAS is initially used for financial applications (e.g., [6]). It has also been employed for forecasting macroeconomic indicators, especially for the quarterly GDP forecast with monthly indicators(e.g., [7] [8]). The key idea of MIDAS is to use a function with parsimonious parameterization to control the weight of the lagged coefficient, such that the regressand and regressors can be sampled at different frequencies. However, this method also needs to evaluate the data value of different frequency and sum them together. Thus, MIDAS suffers from sampling uncertainty compared with the estimation of unrestricted lags. In practical forecast, this model can only do better in short term forecast than long term forecast. Moreover, due to the computational complexity of the MIDAS model, it cannot afford too many monthly indicators to predict GDP growth rate.

In this paper, we introduce a novel machine learning approach namely Generalized Lasso-Granger (GLG) framework that can learn the temporal causal structures between irregular time series ([9]). In [9], the irregular time series are defined as time series whose observations are not sampled at equally-spaced time stamps. This method in [9] first defines a generalization of the inner product for irregular time series based on non-parametric kernel functions. Then, a Generalized Lasso-Granger (GLG) framework for causality analysis can be expanded to deal with irregular time series. One advantage for this kernel-based irregular time series processing approach is that it needs no directly estimation on the values of missing data in low frequency time series, or restricted assumptions about the generation process. Moreover, as justified by the experiment results shown in [9], this method has an excellent convergence behavior. It is a tremendous advantage for the macroeconomic indicators forecasting jobs which usually have insufficient data samples. Better performance of predict accuracy can be achieved by applying our method to forecast quarterly GDP with mixed frequency data set.

Empirically, we chose 17 macroeconomic indicators to build a GLG model. The indictors are chosen from selected areas such as consumption capacity, monetary, CPI inflation and equity market. As justified by the comparison experiments with AR and MIDAS model, the GLG model has a better forecasting accuracy than the AR model in most of the time. The GLG model also has a better performance than the MIDAS model when it comes to the long-term forecasting job.

The rest of the paper is organized as follows. In Section 2, we formulate the problem of GDP forecast based on mixed frequency macroeconomic indictors data set. In Section 3, we briefly introduce the AR model, MIDAS model and the notion of Granger causality. In Section 4, we elaborate our solution of taking the advantage of both kernel function method and GLG framework to find the temporal causal structures among macroeconomic factors. Empirical results based on mixed frequency time-series data of a set of 17 China major macroeconomic factors are presented in Section 5. Finally, some conclusions and future work are drawn in Section 6.

2 Problem Formulation

Conceptually, the problem of forecast GDP growth rate with mixed frequency data can be formulated as a 3-step process. First, select those macroeconomic indicators that can be used in the forecast model. For instance, indicators are often selected along the lines of consumption capacity, investment, export and import, monetary transmission and equity market. Second, choose an effective way to deal with the mixed frequency data set. So far, we can choose the average method, interpolation method, or some special mixed data frequency processing model. Third, apply an appropriate forecast model which can capture the temporal causal structures among the selected factors and use the collected data samples of these factors to predict the GDP growth rate. The set of the data samples can either be monthly or quarterly.

Although there are some non-linear forecast models such as neural network have been introduced to the macroeconomic area, the main stream thinking of the relationships among macroeconomic indicators is still linear assumption. This is mainly because a linear model not only can lead to simpler and more effective computation with more accurate results, but also have interpretable results among all the macroeconomic indicators. Therefore, we follow the same line of thought in approaching the GDP forecast among a set of macroeconomic factors using the linear modeling methodology. We use the same assumption which we have already used in [10]:

Assumption. Each macroeconomic indicator is affected by a limited number of other indicators, either endogenous or exogenous factors. The relationship can be described by linear equations.

In order to deal with the mixed frequency data set of macroeconomic indicators, we consider the time series in different frequency as irregular time series, the length of them is different, but the time stamp of them is strictly increasing. We can use the definition in [9]:

Definition. An irregular time series x of length N is denoted by $x = \{(t_n, x_n)\}_{n=1}^{N}$ where time-stamp sequence $\{t_n\}$ are strictly increasing, and x_n are the value of the time series at the corresponding time stamps.

With all the above considerations, the problem of forecast GDP growth rate with mixed frequency data can be put as follows:

Given a set of m macroeconomic indicators which conclude the GDP growth rate, $\mathbf{I}$=$\{I_1, I_2, ..., I_m\}$, we try to find linear equations representing the temporal casual relationships in $\mathbf{I}$. Let us solve the value of GDP growth rate from the linear equations. Then, we can use the history data samples of the macroeconomic indicators at the right side of the equal sign to forecast the GDP growth rate.

In particular, data samples S =$\{\{(t_{n1}, x_{n1})\}_{n1=1}^{N1}, \{(t_{n2}, x_{n2})\}_{n2=1}^{N2}, ...,$ $\{(t_{nm}, x_{nm})\}_{nm=1}^{Nm}\}$is a set of irregular time series.

3 Related Works

In this section, we briefly review some exist work on GDP growth rate forecast and the basic notion of Granger Causality.

3.1 The AR Benchmark Model

Formally, the simple autoregressive (AR (p)) model is given by:

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t \tag{1}$$

where $\varphi_1, ..., \varphi_p$ are the parameters of the model, c is a constant, and ε_t is white noise. p is the lag length which can be determined by the Schwarz criterion.

The AR model only can use the quarterly data to do the forecast, we can use it as a benchmark to examine whether the other model which involved the mixed frequency data can improve the forecast accuracy or not.

3.2 MIDAS Model

The MIDAS models are closely related to distributed lag models ([11]). Formally, the distributed lag model is given by:

$$y_t = \beta_0 + B(L)x_t + \varepsilon_t \tag{2}$$

Where $B(L)$ is a lag polynomial operator which can be either finite or infinite.

The key idea of the MIDAS model is replace the lag polynomial operator in Eq. 2 by a weighting function of the higher frequency explanatory variables.

The basic MIDAS model for a single explanatory variable, and h-step ahead forecasting, is given by:

$$y_t = \beta_0 + \beta_1 B(L^{1/m}, \theta) x_{t-h}^{(m)} + \varepsilon_t \tag{3}$$

Where $B(L^{1/m}, \theta) = \sum_{k=1}^{K} b(k, \theta) L^{(k-1)/m}$, and $L^{s/m} x_{t-1}^{(m)} = x_{t-1-s/m}^{(m)}$. Here t is the low frequency time unit (in this case, quarters), and m is the higher sampling frequency (if $m = 3$, it means x is monthly and y is quarterly).

For instance, if y_t represent the data in quarter 1 of the year 2013, then $x_t^{(3)}$ represent the data in March of the year 2013, and $x_{t-1/3}^{(3)}$ represent the data in February of the year 2013, $x_{t-1}^{(3)}$ represent the data in December of the year 2012.

The key step of MIDAS model is to estimate the parameters k and θ in the weighting function $b(k, \theta)$. One of the most widely used weighting function called Exponential Almon Lag function [12] is given by:

$$b(k,\theta) = \frac{exp(\theta_0 + \theta_1 k + \theta_2 k^2 + ... + \theta_p k^p)}{\sum_{k=1}^{K} exp(\theta_0 + \theta_1 k + \theta_2 k^2 + ... + \theta_p k^p)} \tag{4}$$

With this weighting function, and the sample data, the MIDAS model can incorporate within-quarter monthly observations on the indicator variable in a simple fashion.

There is lots of researchers use the MIDAS to forecast the quarterly GDP growth rate. However, the results show that the MIDAS model only can performance well in short term. This is mainly because it is very sensitive to the estimation of the parameters. Moreover, due to the computational complexity of the MIDAS model, it cant involve too many monthly indicators to predict GDP growth rate.

3.3 Granger Causality

The Granger causality test ([13]) is a statistical hypothesis test for determining whether one time series is useful in forecasting another. In other words, among the following two regressions:

$$x_t = \sum_{l=1}^{L} a_l x_{t-l} \tag{5}$$

$$x_t = \sum_{l=1}^{L} a_l x_{t-l} + \sum_{l=1}^{L} b_l y_t - l \tag{6}$$

Where L is the time lag, if Eq. 6 is a significantly better model than Eq. 5, we determine that time series y Granger causes time series x. Most existing algorithms for detecting Granger causality are based on a statistical significance test such as t-statistics or F-test.

This method has gained tremendous success due to its simplicity and robustness. The only drawback of this method is that it is very sensitive to the number of observations in the autoregression. In the next section, we will introduce a Generalized Lasso-Granger (GLG) framework to solve this problem. Moreover with the redefine of the inner product, the GLG framework can be easily used to deal with the mixed frequency data set.

4 Use the GLG Method to Forecast GDP Growth Rate with Mixed Frequency Data

In [14], it proposed a GLG framework which takes advantage of the Lasso method ([15]).The basic idea of the GLG is to perform variable selection with the L1-penalty. To be specifically, for each time series $x^{(i)}$, we can obtain a sparse solution of the coefficients a by solving the following Lasso problem among $\{a_{(i,j)}\}$:

$$min \sum_{t=L+1}^{T} \|x_t^{(i)} - \sum_{j=1}^{P} a_{i,j}^T x_{t,Lagged}^{(j)}\|_2^2 + \lambda \|a_i\|_1 \tag{7}$$

Where $x_{t,Lagged}^{(j)}$ is the concatenated vector of lagged observations, $a_{i,j}$ is the j-th vector of coefficients. a_i modeling the effect of the time series j on time series i, λ is the penalty parameter, which determines the sparseness of a_i.

The Lasso-Granger method not only significantly reduces the computational complexity compared with pairwise significant tests, but also achieves superior performance in terms of accuracy and scalability.

When it comes to build the temporal causal networks for irregular time series, the major challenge is how to capture the temporal dependence without directly estimating the values of missing data of low frequency time series or making restricted assumptions about the generation process of the time series. The proposal in [9] uses a kernel-based definition of inner product to measure the temporal dependence between time series. Lets go back to Eq. 7, if we treat $a_{i,j}$ as a time series, the $a_{i,j}x_{t,Lagged}^{(j)}$ can be consider as its inner product with another time series $x^{(j)}$. Since the definition of the inner product can be used in irregular time series, so the GLG framework can be extended to handle irregular time series.

For example, in [9], it defined the inner product as follows:

$$x \odot y = \sum_{n=1}^{N_x} \frac{\sum_{m=1}^{N_y} x_n y_m w(t_n^x, t_m^y)}{\sum_{m=1}^{N_y} w(t_n^x, t_m^y)} \tag{8}$$

Where w is the kernel function. A widely used kernel function is Gaussian kernel which can be given as follows:

$$w_{l,\Delta t}(t_1, t_2) = exp(-\frac{(t_2 - t_1 - l\Delta t)^2}{\sigma^2}) \tag{9}$$

Where σ is the kernel bandwidth, and Δt is the average length of the sampling intervals for the first time series. With this definition, we can construct a pseudo time series:

$$a_{i,j}^{'}(t) = (t_l, a_{i,j,l} | l = 1, ..., L, t_l = t - l\Delta t) \tag{10}$$

Which means for different value of t, $a_{i,j}^{'}(t)$ share the same observation vectors i.e, $\{a_{i,j}\}$, but the time stamp vectors vary according to the value of t.

Given the above product operator and the pseudo time series, we can now extend the regression in Eq. 7 to obtain the desired optimization problem for irregular time series. We can perform the temporal dependence analysis by GLG method through solve the following optimization problem among $\{a_{(i,j)}\}$:

$$min \sum_{n=l_0}^{N_i} \|x_n^{(i)} - \sum_{j=1}^{P} a_{i,j}^{'}(t_n^{(i)}) \odot x^{(j)}\|_2^2 + \lambda \|a_i\|_1 \tag{11}$$

where l_0 is the smallest value of n that satisfies $t_n^{(i)} \geq L\Delta t$.

With this method, we can forecast the quarterly GDP growth rate by mixed frequency data set. We dont have to directly estimating the values of missing data of low frequency time series like the interpolation method do, we also do not have to make restricted assumptions about the generation process of the time series like MIDAS model do.

4.1 Experimental Result

17 Chinese macroeconomic indicators are chosen from selected areas such as consumption capacity, monetary, CPI inflation, equity market. The indicators and the publication lag of them are shown in Table . 1.

We use GLG model do the forecast work. First, we normalize all the data to the same numeric level. Then we organized the data as the form of $< Timesstamps, Values >$, the start point of the training set is $1994 : 12$, and the test set is from $2003 : 12 - 2012 : 12$. We update the training set every month. For example, if we want to forecast the quarterly GDP growth rate of September 2003, we use the monthly data from $1994 : 12 - 2003 : 8$ and quarterly data from $1994 : 12 - 2003 : 6$. Next, we use a Gaussian kernel and with the Parameter $\lambda = 0.01$ and $\sigma = 2$[1] to applied the GLG framework. Then we obtain the coefficient matrix of each lagged time series and use it to forecast the quarterly GDP growth rate. The real GDP growth rate and the forecasted GDP growth rate using GLG model is shown in Fig. 1.

We evaluate forecasting accuracy by examine the mean square forecast errors (MSFE). We use the ratio between the GLG forecast models MSFE and other models MSFE to evaluate which model is better. For example, $rMSFE_{AR} = MSFE_{GLG}/MSFE_{AR}$. If $rMSFE_{AR} < 1$, it means the GLG model is better than AR. The result of the comparison of AR for each year between $2003 - 2012$ is shown in Table 2.

In [16], it applied a MIDAS(3,K,h) model which conclude three monthly indictors[2] to forecast the quarterly GDP growth rate. It given the result of the $rMSFE_{AR}$ of MIDAS from $2007 : 09 - 2010 : 09$, we give the result of the comparison between MIDAS and GLG in Table 3.

[1] The value of λ is the optimal value among $[0.00001, 1]$ after iterative experiments. The value of σ is a widely used bandwidth of Gaussian kernel.

[2] The three monthly indictors are consumer indictor, investment indictor and export indictor.

Table 1. List of Chinese Macroeconomic indictors

Indicators	Data source(frequency)	Publication lag
GDP	NBS(quarterly)	21 days
Retail Sales of Consumer Goods	NBS(monthly)	11 days
Consumer Confidence Index	NBS(monthly)	24 days
Consumer Price Index	NBS(monthly)	11 days
Producer Price Index	NBS(monthly)	13 days
Money Supply M0	PBC(monthly)	19 days
Money Supply M2	PBC(monthly)	19 days
Shanghai stock Exchange trading volume	CSRC(monthly)	0 days
Shanghai Stock Exchange Composite Index	CSRC(monthly)	0 days
Shenzhen Stock Exchange Composite Index	CSRC(monthly)	0 days
Shenzhen stock Exchange trading volume	CSRC(monthly)	0 days
Industrial Sales	NBS(monthly)	13 days
Floor Space Started	NBS(monthly)	24 days
Steel Production	NBS(monthly)	13 days
Auto Production	NBS(monthly)	17 days
Sales-Output Ratio	NBS(monthly)	11 days
Tex Revenue	Mof(monthly)	19 days

Note 1. NBS:National Bureau of Statistics of China; PBC:The Peoples Bank of China; CSRC:China Securities Regulatory Commission; MoF: Ministry of Finance.

Table 2. Contrast Experiment between GLG and AR

Year	$MSFE_{AR}$	$MSFE_{GLG}$	$rMSFE_{AR}$
2003	0.464	0.443	0.954
2004	0.201	0.183	0.909
2005	0.38	0.321	0.846
2006	0.518	0.455	0.878
2007	0.535	0.501	0.934
2008	2.337	2.426	1.038
2009	2.889	2.834	0.981
2010	2.493	2.395	0.96
2011	0.068	0.114	1.667
2012	0.319	0.391	1.223
All samples	1.034	1.021	0.986

From Table 2, we can learn that the GLG model has a higher forecasting accuracy than AR most of the time, only in the years when financial crisis is coming, the GLG model can not performance well, this is may be because we use the equity indictors which have strong fluctuations in these years.

From Table 3, we can learn that the GLG model has a higher performance in long-term forecasting than the MIDAS models. Although the MIDAS model performance well in short-term forecasting, but as it was said in [16]: when $h < 3$, it means the model need the latest GDP value, which can not obtain due to the

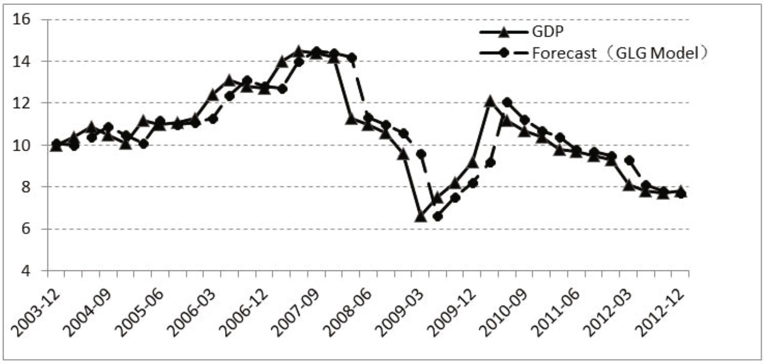


Fig. 1. Chinese real GDP growth rate and GLG forecast model

Table 3. Contrast Experiment between MIDAS(3,K,h) and GLG

h	K=12($rMSFE_{GLG}$)	K=24($rMSFE_{GLG}$)	K=36($rMSFE_{GLG}$)
1	0.694	0.780	0.707
2	0.820	0.713	0.857
3	0.802	0.828	0.826
4	1.146	1.114	1.370
5	1.154	1.159	1.211
6	1.194	1.070	1.192
7	1.232	1.266	1.167
8	2.582	1.704	5.361
9	1.318	1.577	4.946
10	1.657	3.135	7.845
11	1.802	3.814	9.081
12	1.692	2.187	6.713

Note 2. the value in the last three columns represent different MIDAS models $rMSFE_{GLG}$=$MSFE_{MIDAS}/MSFE_{GLG}$ (From 2007:09-2010:09)

publication lag. So the test is actually an in-sample test which can not practically do the short-term forecast job.

5 Conclusion and Future Work

In this paper we have introduced an approach which can capture the temporal causal structures between irregular time series. It took advantage of both kernel function method and Lasso method. Empirically, we use a 17 macroeconomic indicators GLG model to forecast Chinese GDP growth rate with Mixed Frequency data set. The accuracy of the forecasting is better than the AR model in most of the time. The long-term forecasting accuracy is also much better than the MIDAS model.

In the contrast experiment between GLG and AR, we can see that the forecasting accuracy of GLG is not as good as before after year 2008. The reason may be is we involved too many information in the past. So, how to build a mechanism to decide the start point of the GLG training set is worth to research in the future.

References

1. Silvestrini, A., Veredas, D.: Temporal aggregation of univariate and multivariate time series models: A survey. Journal of Economic Surveys 22(3), 458–497 (2008)
2. Silvestrini, A., Moulin, L., Salto, M., Veredas, D.: Monitoring and forecasting annual public deficit every month: The case of France. Empirical Economics 34(3), 493–524 (2008)
3. Chow, G., Lin, A.: Best linear unbiased interpolation, distribution, and extrapolation of time series by related series. The Review of Economics and Statistics 53(4), 372–375 (1971)
4. Mittnik, S., Zadrozny, P.A.: Forecasting German GDP at monthly frequency using monthly IFO business conditions data. In: Sturm, J.-E., Wollmershauser, T. (eds.) Ifo Survey Data in Business Cycle and Monetary Policy Analysis, pp. 19–48 (2005)
5. Ghysels, E., Santa-Clara, P., Valkanov, R.: The MIDAS touch: Mixed data sampling regressions. In: Mimeo, vol. 131. Chapel Hill N.C (2004)
6. Ghysels, E., Santa-Clara, P., Valkanov, R.: Predicting volatility: Getting the most out of return data sampled at different frequencies. Journal of Econometrics 131, 59–95 (2006)
7. Marcellino, M., Schumacher, C.: Factor nowcasting of German GDP with ragged-edge data. A model comparison using MIDAS projections. In: Bundes bank Discussion Paper, Series 1 No. 34 (2007)
8. Kuzin, J., Marcellino, M., Schumacher, C.: MIDAS vs. mixed-frequency VAR: Nowcasting GDP in the euro area. International Journal of Forecasting 27(2), 529–542 (2011)
9. Bahadori, Taha, M., Yan, L.: Granger Causality Analysis in Irregular Time Series. In: 9th VLDB Workshop on Secure Data Management, SDM 2012 (2012)
10. Zhe, G., Wang, Z., Wang, L., Tan, S.: Linear non-Gaussian causal discovery from a composite set of major US macroeconomic factors. Expert Systems with Applications 39(12), 10867–10872 (2012)
11. Pesaran, M.H., Shin, Y.: An autoregressive distributed-lag modeling approach to cointegration analysis. Econometric Society Monographs 31, 371–413 (1998)
12. Ghysels, E., Sinko, A., Valkanov, R.: MIDAS regressions: Further results and new directions. Econometric Reviews 26, 53–90 (2006)
13. Granger, C.W.: Investigating causal relations by econometric models and cross-spectral methods. Econometrica 37(3), 424–438 (1969)
14. Arnold, A., Liu, Y., Abe, N.: Temporal causal modeling with graphical granger methods. In: KDD 2007, p. 66. ACM Press, New York (2007)
15. Zou, H.: The adaptive lasso and its oracle properties. Journal of the American Statistical Association 101, 1418–1429 (2006)
16. Liu, H., Liu, J.: Nowcasting and Short-term Forecasting of Chinese Macroeconomic Aggregates: Based on the Empirical Study of MIDAS Model. Economic Research Journal (3), 4–17 (2011)

Poison Identification Based on Bayesian Network: A Novel Improvement on K2 Algorithm via Markov Blanket

Jinke Jiang[1], Juyun Wang[2], Hua Yu[1], and Huijuan Xu[1]

[1] College of Engineering & Information Technology,
University of Chinese Academy of Sciences, 19A Yu Quan Lu, Beijing, China
{jiangjinke10,xuhuijuan09}@mails.ucas.ac.cn, yuh@ucas.ac.cn
[2] The School of Science, Communication University of China, Beijing, China
wangjuyun@cuc.edu.cn

Abstract. The purpose of this paper was to provide help for poison identification via the Bayesian network according to the observed preliminary symptoms of the poisoning people. We proposed a novel improvement on K2 algorithm to solve the problem of the lack of data under the special context. Determining initial node sequence of K2 algorithm via Markov blanket, we improved greatly Bayesian network structure learning with small datasets. Bootstrap data expansion and Gibbs data correction combining with maximum weight spanning tree (MWST) were used to expand the original small data set to further improve the performance and reliability of the structure learning. Then we applied this kind of combination scheme into a real data set to verify its validity and reliability. Finally we were able to quickly distinguish between a variety of biochemical reagents with this method, and the result of the inference can be used to guide emergency rescue after certain biochemical terrorism attack.

Keywords: Bayesian network, Markov blanket, K2 algorithm, Bootstrap data expansion, Gibbs data correction, Biochemical terrorism attack, Poison identification.

1 Introduction

In recent years, terrorism attacks have become increasingly rampant. In addition to the forms of explosion etc., the terrorism attacks using biological and chemical reagents need to attract people's attention. Biochemical reagents will quickly cause significant harm to humans, even death in the short term. Both the Tokyo subway gas attack in 1995 and American anthrax attack in 2001 caused serious casualties and property losses and led to the serious social panic[1][2]. Therefore, after biochemical terrorist attacks, fast poison recognition will be helpful to decision makers for emergency rescue and play a crucial role.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 173–182, 2013.

Presently the research about poison identification is rare. Diane and Michael put forward the detection methods such as biosensors to detect the anthrax and smallpox attack [3]. Stimpfl etc. presented a computer-assisted identification of poisons in biological materials according to poison quality spectrum [4]. Jon-Erik and Rebecca introduced timely diagnosis and proper treatment based on the spectrum of clinical symptoms and signs associated during the 2001 US anthrax attacks, reducing the poisoning death rate [5]. Riza etc. put forward an expert system based on the rules used for poisoning diagnosis and management [6]. However, we cannot quickly identify the poison by recognition technology of instrumental analysis and chemical alarms, which often take quite a long time. Often it is very difficult and sometimes even impossible to construct the expert systems. Daniel et al proposed a two-step expectation-based scan statistic for monitoring disease outbreaks, to improve the timeliness, accuracy, and spatial resolution of the detection[7]. Yanna et al illustrated the method using a Bayesian outbreak detection algorithm called PANDA to augment traditional clinician outbreak detection[8]. Silvia et al proposed using Bayesian networks to model this kind of system ,used different algorithms for learning Bayesian networks, and applied them to the specific case [9]. In view of the Bayesian network model suitable for uncertain knowledge discovery, we chose the method of learning Bayesian network from the data to fast identify the poison.

In allusion to the lack of data in the special background such as biochemical terrorist attacks, this paper improved respectively in two aspects, structure learning algorithms and data sets: (1) the initial node sequence optimization of K2 algorithm via Markov blanket of the class variable; (2) Bootstrap data expansion and Gibbs data correction combining with the oriented maximum likelihood tree to expand the original small data set. Finally we find the optimal combination of learning Bayesian network structure for our small data set with a variety of toxins.

2 Related Work

In this paper, the data that we use is a small group of case data containing three kinds of poisons, gathered from a large number of case reports. Due to the characteristics of small data sets, we chose the search-and-score method instead of constraint-based dependency analysis method. Because the former is more suitable for less variables and dense networks, while the latter is more suitable for large-scale data sets. We selected the K2 algorithm [10] and Hill-climbing algorithm [11] from the search-and-score method for Bayesian network structure learning. Due to the greedy search strategy of the Hill-climbing algorithm, it is easy to fall into local optimum. However, because of the relatively small data set, it is still unable to restore the real network structure even though it is comparatively better to use the K2 algorithm. The small data set contains insufficient information so that those attribute nodes highly correlated with the class node cannot be closely related to the class node. The characteristics of Markov blanket of the class variable are not only closely related to the class node but also provide the basic information for classification [12]. So we optimized the initial node order of K2 algorithm by combining Markov blanket of the class variable and putting the attribute nodes highly correlated with the class node as closely as possible to the class node. Thereby, we obtained a better recovery of the real structure and improved the accuracy of the classification. Due to the characteristics of small data set, we chose the method of directly learning Markov blanket of class variables instead of the method of building up Bayesian network structure to determine Markov blanket of class variables, which needs a large number of reliable cases. We chose IAMB (incremental association Markov blanket) algorithm [13], a representatively improved algorithm on GS (grow-shrink) algorithm, and KS (Koller-Sahami) algorithm [14], which returns a heuristic approximation to the Markov Blanket of a target variable.

However, when we learn Bayesian networks from small data sets, a large number of sufficient statistical factors are zero, which makes the reliability far from guaranteed. Then we used Bootstrap data expansion to expand the small data set, while this repeatable sampling didn't bring in additional information so that sufficient statistical factors fail to get substantial improvement. Therefore, the effect is not very satisfactory. Owing to that maximum likelihood tree is a tree structure fitting with best Bayesian network and often constitutes the main skeleton of Bayesian network structure, data correction based on the maximum likelihood tree can make the expanded data set change substantively within the record and improve the sufficient statistical factors. Hence, we combined the maximum likelihood tree with Gibbs Sampling to correct the expanded data set [15], and eventually used the expanded large data set for Bayesian network structure learning, further improving the effect of structure learning.

In this paper, we chose Asia network, generated a small data set containing 100 pieces of data from it through sampling and conducted comparative experiments on this small data set. Because our data set contains multiple toxicants and they are each other counter-examples, and these characteristics are similar to the characteristics of node1. Node1, containing positive and negative cases data, is chose as class node, and the other seven nodes as attribute nodes. The evaluation method is the number of right edges in the learned Bayesian network structure compared with the real network structure. And then, we use the Spect heart data set from UCI machine learning repository[16] to verify the optimal combination. Finally we apply the optimal combination to a variety of toxins data set, achieving the purpose of rapid and accurate identification of toxicants and providing help for poison identification in biochemical terrorist attacks.

For our experiments, we use Matlab with the Bayes Net Toolbox [17], the BNT Structure Learning Package [18] and Causal Explorer Package [19].

3 Background

3.1 Oriented MWST

We establish a maximum weight spanning tree (MWST) with the mutual information, orient the edges of the tree, and set the oriented tree as the initial search graph of the Hill-climbing algorithm.

The classical algorithm of establishing MWST is Chow and Liu method[20]: set the mutual information between two points as the edge weight between two points, calculate the weight of all edges, and sort the edges according to descending order; follow the principle of not producing loop, select edges from the queue to build the MWST, until (Nodes-1) edges. The mutual information of discrete random variable is defined as:

$$I(X_i, X_j) = I(X_j, X_i) = \sum_{x_i,x_j} p(x_i, x_j) \log \frac{p(x_i,x_j)}{p(x_i)p(x_j)} . \tag{1}$$

The orientation of edges is essentially causality test between variables. We use the conditional relative average entropy (CRAE) to orient the edges of the MWST [21]. And CRAE is defined as:

$$\mathrm{CRAE}(X_j \rightarrow X_i) = \frac{H(X_i|X_j)}{H(X_i) \cdot |X_i|} . \tag{2}$$

where $|X_i|$ is the number of values of variable X_i, and $H(X_i) = -\sum_{x_i} p(x_i) \log p(x_i)$ is the entropy of discrete random variable X_i, $H(X_i|X_j) = \sum_{x_i,x_j} p(x_i, x_j) \log p(x_i|x_j)$ is the conditional entropy of discrete random variable X_i .If CARE($X_i \rightarrow X_j$) $\geq$ CARE($X_j \rightarrow X_i$), the orientation should be i →j.

3.2 The Initial Node Sequence of K2 Algorithm

We first sort the node blocks of the oriented MWST, then sort the nodes in each node block and finally gain the sequence of all nodes[21][22].

Node Block Sequence. In the oriented MWST composed of attribute nodes, select the nodes with no parent nodes to constitute the first node block, delete the selected nodes and the connected edges, then select the nodes with no parent nodes to constitute the next node block, repeat the previous operations until there are no remaining nodes, and finally we get the unique node block sequence.

The Sequence of Nodes in Each Node Block. In order to sort the nodes in each node block, we first build a complete undirected graph using the nodes in the node block, and then orient the complete graph according to the conditional relative average entropy. The oriented complete graph does not have loop and the topological sort of the oriented complete graph is unique.

3.3 Bootstrap Data Expansion and Gibbs Data Correction

Bootstrap data expansion generates a certain number of samples to form an expanded data set by re-sampling from the original data set. In order to conduct Gibbs data correction [23] in the expanded data set, we need to produce a certain percentage of correction locations uniformly in the expanded data set and initiate these locations randomly. Gibbs data correction is an iterative process, and an iteration completes after all the correction locations in the expanded data set are corrected once.

In an iteration, the process of correcting one data location X_i is defined as follows:

(1) Compute $w = \frac{\prod_{i=1}^{n} p(x_i \mid \pi(x_i))}{\sum_{j=1}^{r_i}\prod_{i=1}^{n} p(x_i^j \mid \pi(x_i))}$, based on the joint probability decomposition of MWST. Among them, n is the number of nodes, x_i is the first possible value of current correction node X_i or the current value of node X_i, $\pi(x_i)$ is the corresponding value of parent nodes, r_i is the number of values of node X_i, here, r_i=2, and we just need to compute one boundary value. As the number of node value increases, the number of boundary value that we need to compute also increases.

(2) Generate a random number λ, and the correction rule of the value of the variable X_i is,

$$\widehat{x}_i = \begin{cases} x_i^1, & \lambda \geq w \\ x_i^2, & \lambda < w \end{cases} . \tag{3}$$

(3) Adjust the node's related parameters after correcting one data, namely the parameters of the node and its child nodes, using the current data set.

(4) Go to step (1) until all the correction locations are corrected. When we encounter that some statistical factors are zero, we could use Laplace correction as follows:

(1) If $p(\pi(x_i))=0$, $p(\pi(x_i))$ uses uniform distribution, and its value is the reciprocal of the number of possible values of parent nodes of node X_i;

(2)If $p(x_i|\pi(x_i))=0$ and $p(\pi(x_i))\neq 0$ then

$$p(x_i|\pi(x_i)) = \frac{1/N^*}{(N^*(\pi(x_i))+N^*(x_i)\cdot(\frac{1}{N^*}))} . \tag{4}$$

where N^* is the number of data in the large data set, $N^*(x_i)$ is the number of value x_i of node X_i and $N^*(\pi(x_i))$ is the number of value $\pi(x_i)$ of parent nodes of node X_i in the large data set.

4 Improved K2 Algorithm via Markov Blanket

We propose a new train of thought determining the initial node sequence of K2 algorithm via Markov blanket of the class variable based on section 3.2. We adjust the features in Markov blanket structure of the class variable to being as adjacent as possible to the class variable, improving the initial node sequence of K2 algorithm.

We choose two Markov blanket structure algorithms, namely IAMB and KS, directly learning Markov blanket of the class variable. The process in which we improve the initial node sequence of K2 algorithm combining with Markov Blanket is as follows:

Step1. We choose IAMB algorithm to use the training sample data, the index of the target variable , domain size of each variable, statistical test method and the corresponding threshold as inputs, and the output is Markov blanket of the target variable, namely IAMB_MB $=x_1, ..., x_n$.

Step2. We choose KS algorithm to use the training sample data, the index of the target variable, domain size of each variable, the number of removed features and size of the Markov blanket estimator as inputs, and the output is removed features and order of removed features, namely KS_MB and KS_ORDER..

Step3. We place the features from IAMB_MB obtained in Step1 directly adjacent to the class node 1, i.e., 1, x_1, ..., x_n, and then select the features from KS_MB obtained in Step2 according to the reverse order of KS_ORDER until we have selected all features from IAMB_MB. In this process, the same features as those in IAMB_MB are still unchanged in accordance with the C, X_1 , ... , X_n, and the other elected features, Y_1, ..., Y_m, corresponding sequence is y_1, ..., y_m, are arranged in a subsequent of C, X_1 , ... , X_n, i.e. C, X_1 , ... , X_n, Y_1, ..., Y_m.

Step4. As to the remaining nodes, we sort them using the method of section 3.2, namely Z_1, ..., Z_t, and the corresponding node sequence is z_1, ..., z_t. Finally, we find the initial node sequence of K2 algorithm via Markov blanket of the class variable and the best initial node sequence is 1,x_1, ..., x_n,y_1, ...,y_m, z_1, ..., z_t.

In summary, we choose the directly learning Markov blanket of the class variable method suitable for small data sets, combining IAMB and KS algorithm. We place the features in Markov blanket structure of the class variable to being as adjacent as possible to the class variable, improving the effect of structure learning and the accuracy of classification. Especially in the case of small data sets, even though the information is not sufficient, we can still gain good effect with the method mentioned in this section.

5 Experimental Results

We use a small data set containing 100 data sampled from Asia network to conduct comparative experiments on this small data set. Firstly, We establish a maximum weight spanning tree based on section 3.1 and gain the initial graph shown in Fig.1(a). Secondly, we first sort the node blocks of the oriented MWST, then sort the nodes in each node block and finally gain the sequence of all nodes, [1 4 3 5 6 8 7 2]. We respectively learn Bayesian networks using Hill-climbing algorithm and K2 algorithm, as is shown in Fig.1 below.

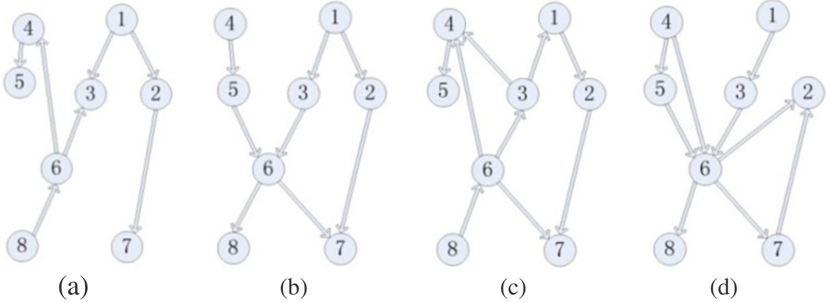


Fig. 1. The initial graph of Hill-climbing algorithm(a),the real structure of Asia network (b), the learned Bayesian network structure using Hill-climbing algorithm (c) and the learned structure using the K2 algorithm and the unique initial node sequence (d).

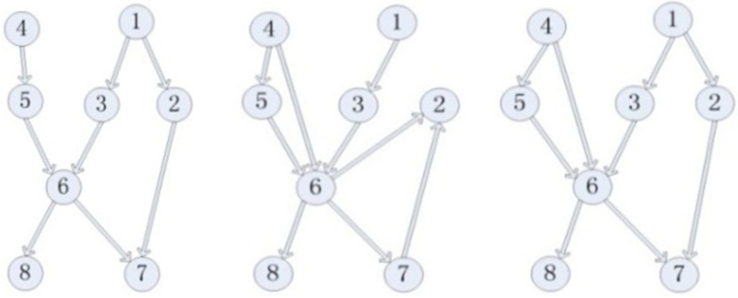
Fig. 1(c) shows that Hill-climbing algorithm with the initial graph has learned a Bayesian network structure with 4 correct sides, 3 reverse sides and 2 extra sides compared with the real structure. In contrast, Fig. 1(d) shows that K2 algorithm with the initial node sequence has learned a Bayesian network structure with 6 correct sides, 1 reverse sides and 2 extra sides compared with the real structure. As you can see, K2 algorithm with initial node sequence has learned Bayesian network structure more close to the standard network structure with the small data set, relative to Hill-climbing algorithm. This is because the greedy search strategy of Hill-climbing algorithm makes the structure learning easy to fall into local optimum, while K2 algorithm given the initial node sequence has learned a better structure.

Next, we improve the initial node sequence of K2 algorithm based on section 4, the results are as follows:

```
IAMB_MB = 2
KS: features = 1   0   0   0   0   0   0   0
order   =   0   7   4   1   2   3   6   5
```

We can see that Markov blanket of the class node ① using IAMB algorithm is node ②. At the same time, we can find node ② is the most relevant to the class node ① using KS algorithm.

So we put the attribute node ② next to the class node ① and the improved initial node sequence of K2 algorithm is [1 2 4 3 5 6 8 7].



(a) (b) (c)

Fig. 2. The real structure of Asia network (a), the learned Bayesian network structure using the K2 algorithm and the unique initial node sequence (b) and the learned Bayesian network structure using the K2 algorithm and the improved unique initial node sequence via Markov blanket (c).

Fig.2(c) shows that K2 algorithm with the improved initial node sequence via Markov blanket of the class variable has learned 8 correct sides, only 1 extra side, greatly improving the effect of structure learning with small data sets and recovering the real network structure. Compared with Hill-climbing algorithm, K2 algorithm has learned a better network structure with small data sets. Further, we have improved the initial node sequence of K2 algorithm via Markov blanket, learning the structure closer to the standard network.

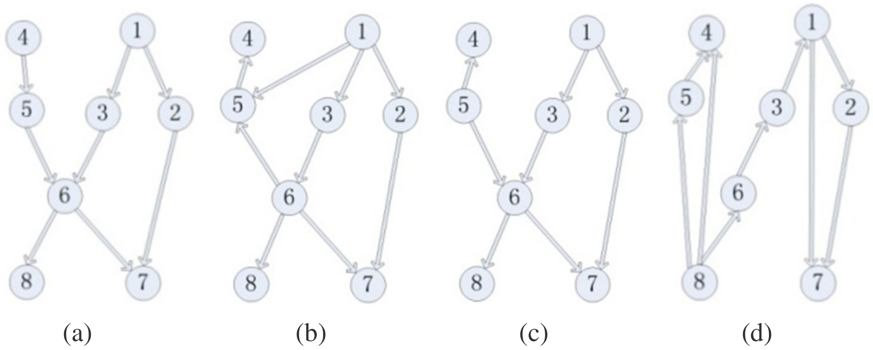


Fig. 3. The real structure of Asia network (a), the learned Bayesian network structure using the K2 algorithm and the unique initial node sequence (b), the learned Bayesian network structure using the K2 algorithm and the improved unique initial node sequence via Markov blanket (c) and the learned Bayesian network structure using the BNPC algorithm (d), using the large data set after Gibbs data correction.

Next, we expand and correct the small data set composed of 100 data sampled from the Asia network with the method mentioned in section 3.3. We expand the small data set to an expanded data set containing 5000 data, and correct 40% of the expanded data. The original data set need not to be corrected. The original small data set and the expanded data set after Gibbs data correction constitute a relatively large data set.

Fig.3 shows that the Hill-climbing algorithm using the large data has learned six correct sides, two reverse sides, an extra side, which is a significant improvement compared to that in the small data set; the improved K2 algorithm has learned eight sides, entirely consistent with the standard network, just one reverse edge. However, we get a more simple structure. Meantime, the BNPC algorithm has learned nine edges, only two correct edges, four reverse edges and three extra edges, learning a worse structure compared to the two search-scoring methods.

From above, we improve the reliability of the Bayesian network learning using Bootstrap data expansion and Gibbs data correction. Compared to the small data set, the Hill-climbing has learned a network structure closer to the real network using the corrected large data set. Likewise, K2 algorithm with the initial node sequence via Markov blanket of the class variable has learned a more simple structure, more accurately recovering the real network structure.

6 The Verification and Application in Real Data Sets

We use a common testing set in the UCI, namely SPECT heart data, to further verify the method mentioned above in this paper. SPECT is a good data set for testing Machine Learning algorithms, which has 267 instances described by 23 binary attributes. And it contains 55 counter examples, 212 positive examples. We take 40 positive examples and 40 counter-examples as training data and the remaining 187 as testing data.

The improved K2 algorithm via Markov blanket has achieved 81.28% accuracy, higher than the accuracy rate 68.98% of the common K2 algorithm. It is clear that the method we propose in this paper has greatly improved the classification performance.

We further apply this method in the real data set containing phosgene, mustard gas and benzene. The data set has 103 instances described by 67 binary attributes and a class node containing three values (respectively represent three poisons). We expand the small data set of 103 data to a large data set containing 5103 data and correct 40% of the expanded data. Then we use the method proposed in this paper to optimize the unique initial node and obtain a trained network to identify the three poisons quickly and accurately by the signs and symptoms of poisoning people. After testing, the accuracy rate is 98.1%. From the learned Bayesian network structure, we find the main symptoms appear directly in the three poisoning symptom list of Toxicology Data Network (TOXNET) [24].

7 Conclusion

This paper proposes a novel improvement on K2 algorithm via Markov blanket to solve the problem of lack of data in some special contexts. We have done a series of experiments based on Asia network and found out that the effect of structure learning by the common K2 algorithm is better than that by Hill-climbing algorithm using small data sets. And then we have further improved the common K2 algorithm combined with IAMB algorithm and KS algorithm and optimized the initial node sequence of K2 algorithm by means of Markov blanket of the class variable, acquiring better results. We have used Bootstrap data expansion and Gibbs data correction to correct the small data set to an expanded large data set, further improving the effect of structure learning. Eventually, we have found a good combination of the improved K2 algorithm via Markov blanket and Bootstrap data expansion and Gibbs data correction, more suitable for Bayesian network structure learning using small data sets. We have used the SPECT in the UCI datasets to verify the reliability and accuracy of the method mentioned in this paper.

The research result can be applied to the on-site identification after the terrorism attacks and accidental chemical leakage. Base on the learned network structure, we can compute the parameters of the nodes of the Bayesian network from the data, enter the attribute node observations, infer the CPT of the class node using the junction tree algorithm, and finally estimate the value of the class node according to the principle of most probable explanation (MPE). Finally we can identify the poison according to the probability to guide the on-site rescue effectively.

Acknowledgments. This work is supported by National Basic Research Program of China (973 program) with Grant No.2011CB706900, National Natural Science Foundation of China (Grant No. 70971128).

References

1. Okumura, T., Takasu, N., Ishimatsu, S., Miyanoki, S., Mitesuhashi, A., Kumada, K., Tanaka, K., Hinohara, S.: Report on 640 victims of the Tokyo subway sarin attack. Ann. Emerg. Med. 28, 129–135 (1996)
2. Public Health Response to Biological and Chemical Weapons-WHO Guidance, World Health Organization, Geneva, vol. 4, pp.103 (2004)
3. Jamrog, D.C., Shatz, M.P., Smith, C.: Modeling Responses to Anthrax and Smallpox Attacks. Lincoln Laboratory Journal 17(1), 115–129 (2007)

4. Stimpfl, T., Demuth, W., Varmuza, K., Vycudilik, W.: Systematic toxicological analysis: computer-assisted identification of poisons in biological materials. Journal of Chromatography B 789, 3–7 (2003)
5. Holty, J.-E.C., Kim, R.Y., Bravata, D.M.: Anthrax: A Systematic Review of Atypical Presentations. Ann. Emerg. Med. 48, 200–211 (2006)
6. Batista-Navarro, R.T.B., Naval Jr., P.C.: ESP: An Expert System for Poisoning Diagnosis and Management 35(2), 53–63 (2010)
7. Neill, D.B.: Expectation-based scan statistics for monitoring spatial time series data. International Journal of Forecasting 25, 498–517 (2009)
8. Shen, Y., Adamou, C., Dowling, J.N., Cooper, G.F.: Estimating the joint disease outbreak-detection time when an automated biosurveillance system is augmenting traditional clinical case finding. Journal of Biomedical Informatics 41, 224–231 (2008)
9. Acid, S., de Campos, L.M.: A comparison of learning algorithms for Bayesian networks: a case study based on data from an emergency medical service. Artificial Intelligence in Medicine 30, 215–232 (2004)
10. Cooper, G.F., Herskovits, E.: A Bayesian Method for the Induction of Probabilistic networks from data. J. Mach. Learn. 9, 308 (1992)
11. Chickering, D.M.: Optimal Structure Identification with Greedy Search. J. Mach. Learn. Res. 3, 507 (2002)
12. Wang, S.C., Leng, C.P., Du, R.J.: Finding optimal feature subset by learning the Markov blanket of class variable. In: ICNC 2009-FSKD 2009, pp. 184–187 (2009)
13. Tsamardinos, I., Aliferis, C.F.: Towards principled feature selection: Relevancy, filters and wrappers. In: Proceedings of the ninth international workshop on Artificial Intelligence and Statistics, Morgan Kaufmann Publishers, Key West, FL,USA (2003)
14. Koller, D., Sahami, M.: Toward Optimal Feature Selection. Technical Report. Stanford Info Lab (1996)
15. Geman, S., Geman, D.: Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration of Images. IEEE Trans. Patt. Anal. Mach. Int. 6(06), 721 (1984)
16. UCI Machine Learning Repository: SPECT Heart Data Set, `http://archive.ics.uci.edu/ml/datasets/SPECTF+Heart`
17. Murphy, K.P.: The Bayes Net Toolbox for Matlab, `http://code.google.com/p/bnt/`
18. Leray, P.: The BNT Structure Learning Package, `http://bnt.insa-rouen.fr/index.html`
19. Aliferis, C.F., Tsamardinos, I., Statnikov, A.R., Brown, L.E.: Causal Explorer: A Causal Probabilistic Network Learning Toolkit for Biomedical Discovery, `http://people.cs.ubc.ca/~murphyk/Software/bnsoft.html`
20. Chow, C.K., Liu, C.N.: Approximating Discrete Probability Distributions with Dependence Trees. IEEE Trans. Inf. Theory. 14(3), 462 (1968)
21. Wang, S.: Bayesian Network Learning, Reasoning and Application, pp. 88–96. Lixin Accounting Publishing House, Shanghai (2010) (in Chinese)
22. Bouckaert, R.R.: Optimizing Causal Orderings for Generating DAGs from Data. In: Proceedings of the Eighth Conference on Uncertainty in Artificial Intelligence, San Francisco, CA, USA (1992)
23. Geman, S., Geman, D.: Stochastic Relaxation, Gibbs Distribution and the Bayesian Restoration of Images. IEEE Trans. Patt. Anal. Mach. Int. 6(06), 721 (1984)
24. United States, National Library of Medicine (NLM), Toxicology Data Network (TOXNET), Occupational Exposure to Hazardous Agents (Haz-Map), `http://hazmap.nlm.nih.gov/cgibin/hazmap_generic?tbl=TblDiseases&id=181`

A New Efficient Text Clustering Ensemble Algorithm Based on Semantic Sequences

Zhonghui Feng, Junpeng Bao, and Kaikai Liu

Institute of Computer Software, Xi'an Jiaotong University, Xi'an 710049, China
{fzh,baojp}@mail.xjtu.edu.cn, xjtuliukk@yeah.net

Abstract. The idea of cluster ensemble is combining the multiple clustering of a data set into a consensus clustering for improving the quality and robustness of results. In this paper, a new text clustering ensemble (TCE) algorithm is proposed. First, text clustering results of applying k-means and semantic sequence algorithms are produced. Then in order to generate co-association matrix between semantic sequences, the clustering results are combined based on the overlap coefficient similarity concept. Finally, the ultimate clusters are obtained by merging documents corresponding to similar semantic sequence on this matrix. Experiment results of proposed method on real data sets are compared with other clustering results produced by individual clustering algorithms. It is showed that TCE is efficient especially on long documents set.

Keywords: text clustering ensemble, overlap coefficient similarity, semantic sequence co-association matrix.

1 Introduction

Text clustering is an important but extremely difficult issue in text mining. The aim of text clustering is grouping documents into classes or clusters so that documents within a cluster have high similarity in comparison to one another, but are very dissimilar to documents in other clusters. In applications, the document is always represented by Vector Space Model (VSM) in which each document is represented as a vector and each unique term is one dimension of this vector. Then, documents are clustered by calculating distance or similarity based on VSM. In large-scale document corpus, the large size of the corpus and high dimensionality of the vector space made the distance or similarity is not able to calculate exactly under some situations. Therefore, the classic algorithms have high time complexity and may create unexpected results.

Cluster ensembles combine multiple clustering results of a set of objects into a single consolidated clustering, often referred to as the consensus solution. Each base clustering refers to a grouping of the same set of objects or its transformed (or perturbed) version using a suitable clustering algorithm. There are many reasons for using a cluster ensemble, such as improved quality of solution, robust clustering, multi-view clustering, etc[1]. Three efficient heuristics are introduced to solve the cluster ensemble problem. All these algorithms approach the problem by first transforming the set of clusterings into a hypergraph representation[2]. In Cluster-based Similarity Partitioning Algorithm (CSPA), a clustering signifies a relationship

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 183–190, 2013.

between objects in the same cluster and can thus be used to establish a measure of pairwise similarity. This induced similarity measure is then used to recluster the objects, yielding a combined clustering. In HyperGraph Partitioning Algorithm (HGPA), the maximum mutual information objective with a constrained minimum cut objective is approximated. Essentially, the cluster ensemble problem is posed as a partitioning problem of a suitably defined hypergraph where hyperedges represent clusters. In Meta-CLustering Algorithm (MCLA), the objective of integration is viewed as a cluster correspondence problem. Essentially, groups of clusters (meta-clusters) have to be identified and consolidated. Later, Fern and Brodley proposed the Hybrid Bipartite Graph Formulation (HBGF) algorithm that is based on bipartite graph partitioning [3]. All these approaches use the efficient graph partitioning algorithm METIS to partition graphs induced by the cluster ensemble and find the consensus clustering [4]. Note that there is implicitly an additional constraint in these solutions, namely that the consensus clusters obtained should be of comparable size [5]. Fred and Jain introduce the concept of evidence accumulation clustering that maps the individual data partitions in a clustering ensemble into a new similarity measure between patterns, summarizing interpattern structure perceived from these clusterings. The final data partition is obtained by applying the single-link method to this new similarity matrix [6].

The idea of cluster ensemble is combining the multiple clustering results for improving the quality and robustness. It is proposed as an important extension of the classical clustering. Existing individual algorithms suffer from different problems in our previous works. K-means algorithms cannot find clusters of arbitrary shape and may create unexpected biased results[7],whereas semantic sequence text clustering(SSTC) algorithm produce too many small-scale clusters[8]. By combining the clustering results of the two algorithms based on the overlap coefficient similar concept, we propose a text clustering ensemble algorithm that can cluster documents efficiently in this paper. The rest of the paper is organized as follows. Section 2 introduces the basic idea of text clustering ensemble. After that, the experimental results are reported in section 3. Finally, Section 4 concludes this paper.

2 Text Clustering Ensemble

The basic idea of text clustering ensemble (TCE) method involves a number of new definitions. We intuitively present these definitions, and then turn to describe the detailed algorithm. Let *D*={ d_1,..., d_k ,..., d_n }(1≤k≤n)be a set of documents, as d_k is the k th document in D. *s* is a sequence of words, i.e. $s = w_1 w_2 \ldots w_m$, as w_i is the *i*th word in *s*. C_K ={ c_k^1 ,..., c_k^i ,..., c_k^m }(1≤i≤m) be the set of k-means algorithm result clusters, as c_k^i is the i th cluster. C_S ={ c_s^1,..., c_s^j ,..., c_s^r } (1≤j≤r) be the set of semantic sequence text clustering(SSTC) algorithm result clusters, as c_s^j is the j th cluster.

2.1 Definitions

A semantic sequence is a word sequence satisfied some conditions within a document, and a document can be regarded as a set of semantic sequences occurring in that document at least once. The key idea is not to cluster the high dimensional vector space, but to consider only the low-dimensional semantic sequences sets.

Definition 1 (Word Distance). Given *s* and a word w_i, the word distance of position *i*(1≤*i*≤*n*) is denoted by $\sigma(i)$, is the number of words between w_i and its preceding occurrence w_h:

$$\sigma(i) = i - h \tag{1}$$

Where $w_i = w_h$ and $w_i \neq w_k$ (1≤*h*<*k*<*i*≤*n*), that is, w_h and w_i are the same word and no other words is the same to w_i between w_h and w_i. If w_h doesn't exist, i.e. w_i occurs for the first time, then $\sigma(i) = \infty$.

Definition 2 (Semantic Density). Given *s*, the semantic density of position *i*(1≤*i*≤*n*) is denoted by $\rho(i)$, is the reciprocal of $\sigma(i)$:

$$\rho(i) = 1/\sigma(i) \tag{2}$$

The semantic density is a kind of word density, but we believe that a sequence of word should imply certain semantic information. In fact, $\sigma(i)$ is the distance of w_i to its preceding occurrence in the sequence *s*, and $\rho(i)$ reflects its local frequency.

Definition 3 (Semantic Sequence). Given *s*, a semantic sequence of *s* is a subsequences of *s*, denoted by $s[i] = w_{i_1} w_{i_2} \ \ldots \ w_{i_3}$ with $i = [i_1, i_2, \cdots, i_r]$ ($1 < i_1 < i_2 < \cdots < i_r \leq m$), which satisfies the following conditions:

① $|s[i]| > 1$

② $0 < i_{k+1} - i_k \leq \varepsilon, 1 \leq k < r$

③ $\rho(i_k) \geq \delta, 1 \leq k < r$

④ $(0 < i_1 - i_h \leq \varepsilon) \rightarrow \rho(i_h) < \delta$

⑤ $(0 < i_j - i_r \leq \varepsilon) \rightarrow \rho(i_j) < \delta$

⑥ $(\forall w_{i_k}, w_{i_l} \in s, i_k \neq i_l) \rightarrow w_{i_k} \neq w_{i_l}$
$(1 \leq k \leq r, 1 \leq l \leq r)$

Where $|s[i]|$ is the number of words in $s[i]$, δ and ε are user defined parameters.

In fact, a semantic sequence in *s* is a continual word sequence after the low density words in *s* are omitted.

Definition 4 (Overlap Coefficient Similarity). For a pair of clusters c_k^i , c_s^j , $sim\,(c_k^i, c_s^j)$ denotes the similarity between the cluster results. This similarity is computed using overlap coefficient concept which is defined as follows:

$$sim\,(c_k^i, c_s^j) = \frac{\left|c_k^i \cap c_s^j\right|}{\min(\left|c_k^i\right|, \left|c_s^j\right|)} \tag{3}$$

Where $|c|$ is the number of documents in the cluster c. If $sim\,(c_k^i, c_s^j)$ is greater than a threshold, we think these clusters are similar enough.

Definition 5 (Semantic Sequences Similarity). Given two semantic sequences $s[j]$ and $s[l]$, c_s^j and c_s^l are corresponding clusters in C_S . The semantic sequences similarity of $s[j]$ and $s[l]$ is denoted by $sim(s[j], s[l])$ which is defined as follows:

$$sim(s[j], s[l]) = \max_i \left(\frac{sim(c_k^i, c_s^j) \times \left|c_s^j\right| + sim(c_k^i, c_s^j) \times \left|c_s^l\right|}{\left|c_s^j\right| + \left|c_s^l\right|} \right) \tag{4}$$

We calculate the similarity of two semantic sequences by using their corresponding clusters overlap coefficient similarity in the sense of maximizing the function in the bracket.

Definition 6 (Semantic Sequence Co-association Matrix). Taken the similarity of pairs of semantic sequences $s[j]$ and $s[l]$ for their association, the semantic sequences are mapped into a co-association matrix:

$$C(j,l) = sim(s[j], s[l]) \tag{5}$$

Where $C(j,l)$ is an entry on the matrix.

2.2 Algorithm

In this section, we will discuss the different steps of the algorithm. The detailed algorithm is described as follows:

Input: A document set *D*, user defined parameters

Output: Set of the result document clustering C_r

Step 1. The main task is to obtain the clusters C_K by using k-means algorithm on *D*.

Step 2. The main task is to obtain the clusters C_S by using SSTC algorithm on *D*.

Step 3. In this phase, for each cluster c_k^i (1≤i≤m) in C_K and each cluster c_s^j (1≤j≤r) in C_S , we compute $sim\,(c_k^i, c_s^j)$ between the pair of clusters using overlap coefficient similarity. The process is repeated until there are no remaining clusters.

Step 4. Based on overlap coefficient similarity computed above, a new semantic sequence co-association matrix which entries denote the semantic sequence similarity are generated.

Step 5. The final clusters C_r in D are obtained by merging documents corresponding to similar semantic sequence on this matrix.

3 Experimental Results

In this section, several experiments were performed to evaluate the effectiveness of our proposed algorithm.

3.1 Experimental Setup

In order to evaluate our method, we used two different sets of documents. One set is News Group 20 (NG20) [9], the other set is papers from proceedings of 2002 International Conference in Machine Learning and Cybernetics (ICMLC02).

Table 1. Twenty News Groups from NG20 Used in Our Experiments

news group	num of doc	news group	num of doc
talk.politics.mideast	252	rec.sport.baseball	82
soc.religion.christian	221	comp.graphics	67
talk.politics.misc	140	comp.windows.x	51
talk.politics.guns	140	comp.os.ms-windows.misc	46
talk.religion.misc	120	rec.autos	41
sci.crypt	117	sci.electronics	40
alt.atheism	112	misc.forsale	37
rec.sport.hockey	107	rec.motorcycles	32
sci. med	104	comp.sys.ibm.pc.hardware	31
sci. space	102	comp.sys.mac.hardware	20

Table 2. Eight Classes from ICMLC02 Used in Our Experiments

class	num of doc	class	num of doc
pattern recognition	115	computing and algorithm	54
machine learning	88	system control	52
data mining	81	fuzzy theory and application	51
network	76	knowledge management and processing	33

NG20 has approximately 20 000 pieces of news which belong to 20 news groups respectively. In our experiments, we choose documents which capacity is larger than 3k bytes in NG20. It left us with 1,862 documents in 30 categories as described in Table 1. Besides, we collect 482 literature papers from the proceedings of ICMLC02 and some documents appears in multiple classes, which covers 8 classes as described in Table 2. Since documents in Reuters21578 are too short to obtain semantic sequences, we use NG20 to replace Reuters21478. All texts in corpus were preprocessed by word extraction, stop words removal, and stemming. So the texts to be clustered are converted into plain text files.

3.2 Performance Analysis

In our experiments, we compared the performance of our algorithm against the performance achieved by two common text clustering algorithms: K-Means, Semantic Sequences Text Clustering (SSTC). The effectiveness of algorithm was evaluated using Macro-averaging and Micro-averaging metric to measure precision and recall of algorithm. Figure.1 and Figure.2 shows performance of the three algorithms on two document set.

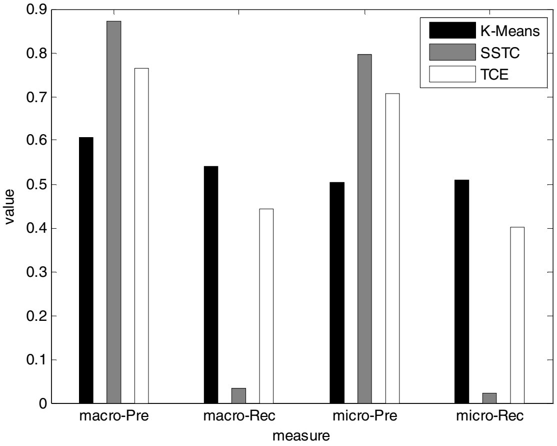


Fig. 1. Performance of three different algorithms on NG20

Experiment results showed that TCE is efficient especially on long documents set. By merging individually text cluster results, the precision and recall are improved. As above figures demonstrate, whatever Macro or Micro measure is selected, we see that the precision values of TCE are always greater than K-Means algorithms but less than SSTC algorithm. As we mentioned above, SSTC is a semantic sequence approach, which selects some words sequence from a document to reflect local features. That leads to the high precision of SSTC especially on long documents. On the contrary, k-means algorithms mentioned here are all based on VSM, which exploits the similar frequency distribution words to reflect global features. Thus, the inner structures of the document cannot response apparently, so it is difficult to improve precision further.

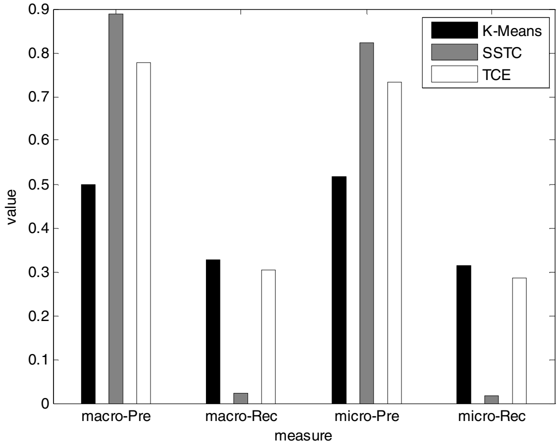


Fig. 2. Performance of three different algorithms on ICMLC02

Furthermore, the macro or micro recall values of TCE are always greater than SSTC algorithms but less than K-Means algorithm. In natural language, people often use different word or phrase to represent similar meanings, and this makes different documents may have similar content. These documents with similar content can not be detected by measuring semantic sequences, so SSTC produces a lot of small-scale clusters. That leads to the low recall of SSTC. By contrast, the high dimensional vector of VSM may contain more similar content, and this decreases the influence of synonym and polysemy. Therefore, the other two algorithms have higher recall than SSTC.

4 Conclusions

Text clustering is a research issue in text mining. The idea of cluster ensemble is combining the multiple clustering results for improving the quality and robustness. It is proposed as an important extension of the classical clustering. We propose a new text clustering ensemble (TCE) algorithm in this paper. First, text clustering results of applying k-means and semantic sequence algorithms are produced. Then in order to generate co-association matrix between semantic sequences, the clustering results are combined based on the overlap coefficient similarity concept. Finally, the ultimate clusters are obtained by merging documents corresponding to similar semantic sequence on this matrix. The experiment results demonstrate the feasibility of using TCE in real applications involving the well-known NG20 data. It is also showed that TCE is efficient especially on long documents set.

Acknowledgments. The work was supported by the National Natural Science Foundation of China under Grant No. 60903123.

References

1. Ghosh, J., Acharya, A.: Cluster ensembles. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 1(4), 305–315 (2011)
2. Alexander, S., Ghosh, J.: Cluster ensembles—a knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research 3, 583–617 (2002)
3. Fern, X.Z., Brodley, C.E.: Solving cluster ensemble problems by bipartite graph partitioning. In: Proceedings of the Twenty-first International Conference on Machine Learning. ACM Press (2004)
4. Karypis, G., Kumar, V.: A fast and high quality multilevel scheme for partitioning irregular graphs. Journal on Scientific Computing 20(1), 359–392 (1998)
5. Punera, K., Ghosh, J.: Soft cluster ensembles. Advances in Fuzzy Clustering and its Applications. John Wiley & Sons, Ltd. (2007)
6. Fred, A.L.N., Jain, A.K.: Combining multiple clusterings using evidence accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(6), 835–850 (2005)
7. Han, J., Kamber, M.: Data Mining: Concepts and Techniques, 2nd edn. Morgan Kaufmann, San Francisco (2006)
8. Feng, Z.-H., Shen, J.-Y., Bao, J.-P.: An Incremental Algorithm of Text Clustering Based on Semantic Sequences. WuHan University Journal of Natural Sciences 11(5), 1340–1344 (2006)
9. Slonim, N., Tishby, N.: Document clustering using word clusters via the information bottleneck method. In: Proceedings of the 21th ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 208–215. ACM Press, Athens (2000)

Credit Scoring Analysis Using B-Cell Algorithm and K-Nearest Neighbor Classifiers*

Cheng-An Li

College of Economics and Commerce
South China University of Technology, Guangzhou, 510006, P.R. China
chanli@scut.edu.cn

Abstract. This paper applies B-Cell algorithm (BCA) for credit scoring analysis problems. The proposed BCA-based method is combined with k-nearest neighbor (kNN) classifiers. In the algorithm, BCA is introduced to select the optimal feature subsets and kNNs are used to classify the investors in different groups representing different levels of credit in the classification phase. Experiments employing the benchmark data sets from UCI databases will be used to measure the performance of the algorithm. Its comparison with genetic algorithm, particle swarm optimization and ant colony optimization will be shown.

Keywords: Classification, Credit Scoring, K-Nearest Neighbor Classifiers, B-Cell Algorithm.

1 Introduction

Credit scoring is a major issue for financial institutions, for firms that grant credit to their customers, as well as for institutional and individual investors. One of the key decisions financial institutions have to make is to decide whether or not to grant a loan to a customer. The decision basically boils down to a binary classification problem so as to distinguish customers with low credit risk from customers with high credit risk [1], [2].

Selecting the right sets of features for classification is one of the most important problems in designing a good classifier [1]. The objective of feature selection is to search through the space of feature subsets to identify the optimal or near-optimal one with respect to a selected performance measure. In the literature, many feature selection algorithms have been proposed [2]. These algorithms can be classified into two categories, that is, filter methods that select variables by ranking them with correlation coefficients and wrapper methods that assess subsets of variables according to their usefulness to a given predictor.

Recently, randomized metaheuristic search algorithms, including simulated annealing, genetic algorithms [3], particle swarm optimizations (PSO) [4]and ant

* This paper is supported by the Guangdong Planning Office of Philosophy and Social Science (Grant No. GD10CGL11).

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 191–199, 2013.

colony optimizations (ACO) [5] are of great interest because they often yield high accuracy and are much faster, and have been used to find the optimal feature subsets in many real-world cases, including credit scoring analysis. However, the complexity of the methods in implementation made them difficult to solve some real problems.

The B-Cell algorithm (BCA) [6] is one of artificial immune algorithms, which is built on the clonal selection principle and uses an alternative model called somatic contiguous hypermutation for the mutation probability in the affinity mutation process. It is very important that BCA is very simple and has a small number of parameters [7]. Thus, in this paper, we propose a novel approach to solve the feature subset selection problem using B-Cell algorithm. In the classification phase of the proposal algorithm, the nearest neighbor classification methods are used. The proposed BCA-based algorithm is tested using the data from UCI benchmark databases in order to classify the investors in different groups representing different levels of credit risk. In addition, the other algorithms are compared with the proposed algorithms in order to validate their efficiency.

This paper is organized as follows. Section 2 summarizes the principle of BCA briefly. Section 3 addresses the BCA based classification method. Section 4 shows its application and experimental results. Finally, Section 5 concludes the paper and discusses briefly.

2 B-Cell Algorithm

2.1 B-Cell Algorithm

The B-Cell algorithm (BCA) [6] is inspired by the clonal selection process. It implements non-deterministic iterated process of exploring multidimensional search space and works with a population of tentative solutions called B-cells.

An important feature of BCA is its use of a unique mutation operator, known as contiguous somatic hypermutation. Here, a contiguous region of the point representation is randomly selected and each position is mutated with a certain probability. The general principles of the BCA method are outlined as follows.

In BCA, each cell is a candidate solution that has an associated performance index that allows it to be compared with the other cells. Each cell, represented as $x_i = (x_{i1}, x_{i2}, \cdots x_{iN})$, is composed of the binary variable and decides on {1} or {0}, where N is dimension number. Computationally, the implementation of the algorithm is quite simple and could be performed using the following steps:

--First, generate an initially randomized population of individuals P over the search space.

--Second, evaluate each individual. Clone and place in clonal pool C.

--Third, for each clone, apply the contiguous somatic hypermutation operator.

--Fourth, evaluate each clone; update the individual if a clone has higher affinity than its parent B-cell.

Stopping Criterion:

BCA is usually terminated with a maximal number of iterations or the entire population cannot be improved further after a sufficiently large number of iterations.

2.2 The Contiguous Somatic Hypermutation Operators

The immune-inspired mutation operator is one of the main characteristic of the B-Cell algorithm. The operator is called the somatic contiguous hypermutaion operator [8]. It decides randomly about a contiguous region of the bit string and flips each bit within this region with a given probability $r \in [0,1]$. In this paper, the somatic contiguous hypermutaion which has wrapping around has been used in the BCA shown in the following.

Definition 1 Somatic Contiguous Hypermutation (CSM) Wrapping Around [9]). *CHM mutate $x \in \{0,1\}^n$, given a parameter $r \in [0,1]$.*

1. *Select $p \in \{0, 1, \ldots, n-1\}$uniformly at random.*
2. *Select $l \in \{0, 1, \ldots, n\}$uniformly at random.*
3. *For $i=p$ to $l-1$ do*
4. *With probability r set $x[(p+1) \bmod n]=1- x[(p+1) \bmod n]$.*

2.3 B-Cell Algorithm for Classification

In this paper, we introduce BCA into classification, in particular, credit scoring problem. To the best of our knowledge, no related work about BCA for classification problems has been reported in the previous literatures. A new algorithm is proposed by utilizing the k-Nearest Neighbor Classifiers and BCA for credit scoring problems. In the proposed method, BCA is employed to find the optimal feature subset and an independent classifier to evaluate the quality of the subset.

3 The Proposed B -Cell Algorithm

3.1 The Method

In this paper, an algorithm for the solution to the feature selection problem based on the B-Cell Algorithm is presented. We apply B-Cell Algorithm to select optimal feature subsets. This algorithm is combined with k-Nearest Neighbor classifiers which are used to evaluate the quality of the selected feature subsets. A pseudo-code of the proposed BCA -based method is presented in Table 1.

Table 1. B-Cell Algorithm

```
Begin
  Randomly Initialize the population P
  Do until the maximum number of solutions has been reached:
    Select randomly a number of features to activate
Enddo
Selection of the maximum number of generations
While (number of iterations, or the termination criterion is not met)
    Classify given data based each B-cell in the population
    For each $x_i \in P$
      Evaluate  $x_i$ with a fitness function $f(x)$
      Clone $x_i$, and place in clonal pool C
      Randomly select a clone $c \in C$, randomise the vector
      Each $c \in C$, apply the contiguous somatic hypermutation operator
      Evaluate each clone c with the fitness function $f(x)$
    If a clone has higher fitness than its parent $x_i$ then
      $x_i = c$
    endif
  next $x_i$
  Update the optimal cell with new population P
next generation until termination criterion
  Return the best cell ( best solution)
End
```

In each iteration, the algorithm records and updates the global optimal solution.

3.2 K-Nearest Neighbor Classifiers

In this paper, the classic k-Nearest Neighbor (*k*NN) methods are used for classification after a number of features were selected in each iteration. For each sample of the test set, the Euclidean Distance from each sample in the training set is calculated. The Euclidean Distance is calculated as follows:

$$D_{ij} = \sqrt{\sum_{l=1}^{d} | x_{il} - x_{jl} |^2} \tag{1}$$

where D_{ij} is the distance between the test sample $i=1,\ldots, M_{test}$ (M_{test} is the number of test samples) and the training sample $j=1,\ldots, M_{train}$ (M_{train} is the number of train samples), and $l=1,\ldots, d$ is the number of activated features in each iteration.

In the classic k-Nearest neighbor methods, every member among the k nearest neighbors has an equal weight in the vote. As described in Ref. [5], it is nature more weight to those members that are closer to the test sample. A method, Weighted k Nearest Neighbor (wkNN), is proposed [5]. In the wkNN method, the most distant neighbor from the test sample has the smallest weight while the nearest neighbor has the largest weight. The weight of th *i*th neighbor is set:

$$w_i = \frac{i}{\sum_{i=1}^{k} i} \tag{2}$$

3.3 Population

A state vector of all features is denoted as a component or cell. In feature selection problems, we represent the cell by binary bit strings of length N, where N is the total number of attributes. Every bit represents an attribute, the value '1' means the corresponding attribute is selected while '0' not selected. Each vector is an attribute subset.

A number of cells are used. Each cell begins from a random initialization in the feature vector and changes according to a mutation operator.

3.4 Fitness Function

To effectually evaluate the quality of the produced members of the population, the following fitness function is adopted in our method with the classification accuracy of good group and bad group and the number of selected features. Thus, for each cell, the classifiers are called and the produced number of selected features and overall classification accuracy are used to generate the fitness function.

Accuracy of the good group is:

$$AC_1 = \frac{T_1}{T_1 + F_1} \tag{3}$$

Accuracy of the bad group is:

$$AC_2 = \frac{T_2}{T_2 + F_2} \tag{4}$$

where T_1 and F_1 are correct and error classification for good group respectively, and T_2 and F_2 are correct and error classification for bad group respectively.

Hence, the fitness function can be defined as:

$$fitness = \alpha * AC_1 * AC_2 + \beta * (1/(5 + Fs)) \tag{5}$$

where α and β are constant factors used to weight each term in (5), $0 \le \alpha \le 1$ and $\alpha + \beta = 1$. Fs is denoted as the number of the selected features.

4 Experiments

This section describes the experimental setup used to test the performance of the proposed approach. We compare the proposed BCA-based algorithm with the GA-based, PSO-based, ACO-based algorithm in credit scoring by using two public credit card datasets (Australian, and German credit datasets) from the UCI KDD Archive [10].

4.1 Data Sets and Parameter Setting

The data are obtained from UCI Repository of Machine Learning Databases [10] illustrated in Table 2. The Australian credit data consists of 307 instances of creditworthy applicants and 383 instances where credit is not creditworthy. This dataset has a good mixture of attributes: continuous, nominal with small numbers of values, and nominal with larger numbers of values, along with missing attributes in 5% of the samples. To protect the confidentiality of data, the attributes names and values have been changed to meaningless symbolic data. The dataset are preprocessed by fitting suitable values into the missing values for the original data set. The German credit data are more unbalanced, and it consists of 700 instances of creditworthy applicants and 300 instances where credit should not be extended. This data set only consists of numeric attributes. In implementation, the input variables in each dataset were normalized independently in each dimension into the range [0, 1].

Table 2. The data sets from the UCI Repository

No	Names	# Instances	Nominal features	Numeric features	# Class
1	Australian	690	6	9	2
2	German	1000	0	24	2

These data have been classified into two classes: creditworthy, denoted as good group, and not creditworthy, denoted bad group. Hence, the credit scoring task will be to discriminate between these two groups of data. The algorithm was implemented in Matlab 7.10 and Intel at 3.3 GHZ.

The parameters for all algorithms are shown in Table 3.

Table 3. Parameter setting for differen algorithms

Methods	Parameter setting
BCA	Generation: 50, B cells: 5, Clone cells: 5, mutation probability: 0.1
GA	Generation: 100, population size: 50, crossover: 0.9, Mutation: 0.01
PSO	Generation: 50, particles: 25, $w_{\max}$ =0.9, $w_{\min}$ =0.1, c1=2.0, c2=2.0, $V_{\max}$ =6
ACO	*ants*=25, *tmax*=50, *r*=20, r_1=5, *q*=0.5, θ_a =0.25, θ_b =0.95 [5]

In addition, α and β in fitness function are set to 0.8 and 0.2 respectively.

4.2 Evaluation Functions

Seven measures are used to validate the proposed method: number of selected features, the Root Mean Squared Error (RMSE), accuracy of good group, accuracy of bad group, average accuracy, overall accuracy and computational time.

RMSE can be calculated from the formula:

$$RMSE = \sqrt{\sum_{i=1}^{M} | y_i - y_{ei} |^2 / M} \tag{6}$$

where M is the number of samples in the data set, y_{ei} is the classifier model output and y_i is the true class of the test sample i.

Accuracy of good group and bad group defined in (3) and (4) respectively.

The Average Classification Accuracy (ACA) is :

$$ACA = (AC_1 + AC_2)/2 \,. \tag{7}$$

The Overall Classification Accuracy (OCA) is:

$$OCA = \frac{T}{M} 100 \,. \tag{8}$$

In (8), T is the number of the samples classified correctly. In fact, $T=T_1+T_2$.

In the experiments, the algorithms used the datasets with 10-fold cross validation. The mean values of number of selected features, RMSE and classification accuracy of 10-fold cross validation and computational time for three data sets are shown in the following tables Table 4 and 5.

4.3 Experimental Results

We implemented BCA and compared experiments with the other randomized search algorithms GA, PSO and ACO. We used 1NN and wkNN as base classifiers in each of the iterations. The results reported consist of the average number of selected features, RMSE, accuracy of good group, accuracy of bad group, average accuracy, overall accuracy and computational time. The classification results are presented in Table 4 and 5.

From Table 4 and 5, in the two base classifiers 1NN and wkNN, experimental results show that wkNN outperform 1NN except computational time for BCA, GA, PSO and ACO in general. Hence, we focused on the result analysis about wkNN as the base classifier in the following.

Table 4. Results of the proposed algorithm and the compared algorithms on Australian dataset

Methods	Average number of features	RMSE	Accuracy for good group (%)	Accuracy for bad group (%)	Average Accuracy (%)	Overall Accuracy (%)	Computational time(s)
BCA-1NN	6.4	0.2849	92.18	91.64	91.91	91.88	40.4199
BCA-wkNN	5.8	*0.2554*	*92.83*	93.99	*93.41*	*93.48*	186.9204
GA-1NN	7.4	0.2998	90.23	91.64	90.94	91.01	97.7806
GA-wkNN	7.2	0.2719	92.18	92.95	92.57	92.61	289.5219
PSO-1NN	6.9	0.2745	92.18	92.69	92.44	92.46	44.3350
PSO-wkNN	*5.6*	0.2638	92.83	93.21	93.02	93.04	147.0794
ACO-1NN	7.9	0.2973	89.58	92.43	91.00	91.16	*24.4885*
ACO-wkNN	7.4	0.2745	90.23	*94.20*	92.24	92.46	72.6886

It can be observed that RMSE, average accuracy, and overall accuracy of the BCA-wkNN (0.2554, 93.41% and 93.48%) are superior to those of GA-wkNN, PSO-wkNN and ACO-wkNN from Table4. Accuracy of good group of the BCA-wkNN (92.83%) is equivalent to that of PSO -wkNN, but better than GA-wkNN and ACO-wkNN. Accuracy of bad group of the BCA-wkNN is slightly inferior to that of ACO-wkNN, but better than GA-wkNN and PSO-wkNN. Average number of selected features of the proposed BCA-wkNN is slightly larger that of PSO -wkNN, but smaller than that of GA-wkNN and ACO-wkNN. It is also noticed that, for computational time, the proposed BCA -wkNN is superior to GA-wkNN, but slightly inferior to PSO-wkNN and ACO -wkNN.

Table 5. Results of the proposed algorithm and the compared algorithms on German dataset

Methods	Average number of features	RMSE	Accuracy for good group (%)	Accuracy for bad group (%)	Average Accuracy (%)	Overall Accuracy (%)	Computational time(s)
BCA-1NN	11.6	0.4266	85.71	*72.67*	79.19	81.80	*102.7111*
BCA-wkNN	13.1	*0.4123*	87.43	*72.67*	*80.05*	*83.00*	378.3960
GA-1NN	12.6	0.4658	82.43	68.67	75.55	78.30	260.6146
GA-wkNN	11.6	0.4393	86.86	66.33	76.60	80.70	657.7960
PSO-1NN	11.2	0.4722	80.71	70.67	75.69	77.70	139.4050
PSO-wkNN	*10.8*	0.4494	84.14	69.67	76.90	79.80	300.0934
ACO-1NN	12.6	0.4324	88.86	63.67	76.26	81.30	126.8113
ACO-wkNN	11.8	0.4159	*91.14*	63.00	77.07	82.70	334.0931

From Table5, we can observe that RMSE, accuracy of bad group, average accuracy, and overall accuracy of the proposed BCA-wkNN (0.4123, 72.67%, 80.05% and 83%) is superior to those of GA-wkNN, PSO-wkNN and ACO-wkNN. Accuracy of good group of the BCA-wkNN is inferior to that of ACO -wkNN, but better than GA-wkNN and PSO-wkNN. For average number of selected features, the proposed BCA-wkNN performs slightly worse than GA-wkNN, PSO-wkNN and ACO-wkNN. Of course, for computational time, the proposed BCA-wkNN is superior to GA-wkNN, but slightly inferior to PSO-wkNN and ACO-wkNN.

In summary, the experimental results demonstrate that the proposed BCA-based approach is effective and has better classification performance than other optimization techniques such as GA, PSO and ACO for the credit scoring problems.

It is noted that the Australian and German datasets are most widely selected to test the performance of algorithms for financial crises [1]. Hence, in this paper, we also use these datasets to validate the proposed method so that future researchers can make direct and fair comparisons.

5 Conclusions and Discussion

This paper discussed the application of BCA for credit scoring. A novel approach has been proposed to solve the feature subset selection problem using B-Cell algorithm. In the approach, the nearest neighbor classification methods are used in the classification phase. The experimental results show that the proposed BCA-based approach outperforms the compared GA, PSO, and ACO-based approaches in the classification performance for credit scoring problems. In the future work, we shall discuss how to apply the proposed BCA algorithm to other applications such as optimal risk portfolios. In addition, we will use the proposed approach in some specific real-world problems.

References

1. Lin, W.-Y., Hu, Y.-H., Tsai, C.-F.: Machine Learning in Financial Crisis Prediction: A Survey. IEEE Transactions on Systems, Man and Cybernetics-Part C: Application and Reviews (2011), doi:10.1109/TSMCC.2011.2170420
2. Crook, J.N., Edelman, D.B., Thomas, L.C.: Recent developments in consumer credit risk assessment. European Journal of Operational Research 183, 1447–1465 (2007)
3. Vafaie, H., Imam, I.F.: Feature selection methods: genetic algorithms vs. greedy-like search. In: Proceedings of International Conference on Fuzzy and Intelligent Control Systems (1994)
4. Yang, C.-S., Chuang, L.-Y., Ke, C.-H., Yang, C.-H.: Boolean Binary Particle Swarm Optimization for Feature Selection. In: IEEE Congress on Evolutionary Computation, pp. 2093–2098 (2008)
5. Murinakis, Y., Marinaki, M.: Applicaton of Ant Colony Optimization to Credit Risk Assessment. New Mathematics and Natural Computation 4(1), 107–122 (2008)
6. Kelsey, J., Timmis, J.: Immune inspired somatic contiguous hypermutation for function optimisation. In: Cantu-Paz, E., et al. (eds.) GECCO 2003. LNCS, vol. 2724, pp. 207–218. Springer, Heidelberg (2003)
7. Zarges, C.: Theoretical Foundations of Artificial Immune Systems, The dissertation of Technical University Dortmund, Dortmund Germany (2011)
8. Jansen, T., Oliveto, P.S., Zarges, C.: On the Analysis of the Immune-Inspired B-Cell Algorithm for the Vertex Cover Problem. In: Liò, P., Nicosia, G., Stibor, T. (eds.) ICARIS 2011. LNCS, vol. 6825, pp. 117–131. Springer, Heidelberg (2011)
9. Jansen, T., Zarges, C.: Analyzing different variants of immune inspired somatic contiguous hypermutations. Theoretical Computer Science 412(6), 517–533 (2011)
10. Murphy, P.M., Aha, D.W.: UCI Repository of machine learning databases, Department of Information and Computer Science. University of California, Irvine (2001)

Text Categorization Based on Semantic Cluster-Hidden Markov Models

Fang Li and Tao Dong

Intelligence Engineering Lab, Beijing University of Chemical Technology,
Beijing 100029, China
lifang@mail.buct.edu.cn, dtao0424@126.com

Abstract. A new text categorization algorithm based on Hidden Markov Model is proposed. At first, semantic clusters are obtained from training data set. The association between semantic clusters is modeled as Hidden Markov Model. Combining with the forward algorithm, the strategy could realize automatic text categorization. From the simulation, the proposed text categorization algorithm is better in categorization precision. Moreover, it works well independent of the number of considered categories compared to the priori art algorithms.

Keywords: text categorization, Hidden Markov models, Semantic Cluster, text serialization.

1 Introduction

Automatic text categorization is an important research field of Natural Language Processing (NLP). Since the first study of text categorization in the late 1950s, many effective text categorization algorithms have been proposed. Among these algorithms, some are based on Deterministic Language Model by exploring grammar, while others are based on Statistical Language Model. Recently, Statistical Language Model is becoming more and more popular. Vector Space Model (VSM) is a kind of Statistical Language Model. However, there are some disadvantages of VSM. For example, words in VSM are assumed to be independent from each other, so that correlations between words are always ignored.

In the early 1990s, Rabiner proposed a scheme for speech recognition of independent word using HMM[1]. The method builds HMM set to describe various possible recognition results and to evaluate speech signal in different HMM. Recognition is a kind of categorization, which is a reference for us.

This paper presents a text categorization algorithm based on Hidden Markov Models (HMM). Each of these HMMs represents a pre-defined category. We take the same document into each HMM and then assign it to the most likely category. We also propose a scheme that reduces document dimension. In this scheme, all of the words in document are clustered by computing their TF-IDF. The result of cluster is named Semantic Cluster, because the words in the same cluster carry similar semantic information.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 200–207, 2013.

The remainder of the paper is organized as follows. Section 2 discusses the feasibility of Hidden Markov Model in text categorization. In Section 3 we elaborate the strategy to serialize text. In Section 4 we present the whole text categorization algorithm. Section 5 presents the performance evaluation results. Finally, we conclude the paper in Section 6.

2 Hidden Markov Model

A Hidden Markov Model is also a kind of statistical model, which can be described by a quintuple (N, M, A, B, π). In the quintuple, N is the number of hidden states, M is the number of observable states, A is state transition matrix showing the probability of a hidden state given the previous hidden state, and B is the confusion matrix showing the probability of observing a particular observable state given a particular hidden state. π is a vector showing the probability of the model being in a particular hidden state at the very beginning. The following diagram shows a simple Hidden Markov Model.

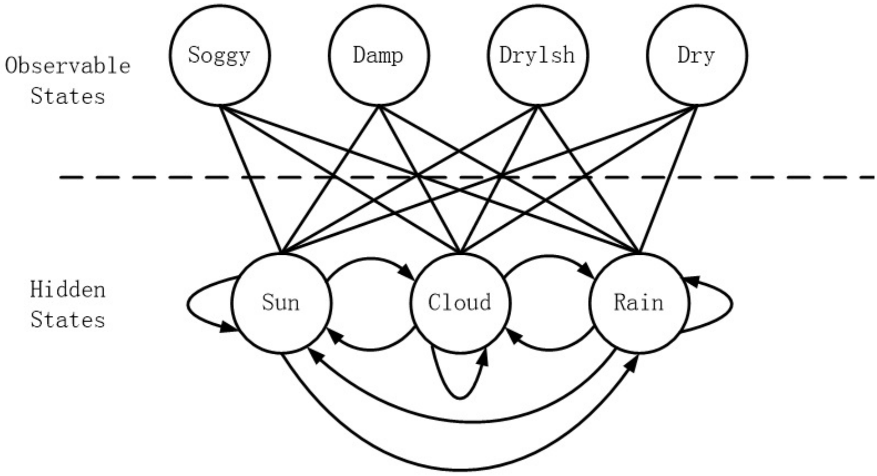


Fig. 1. First-Order Hidden Markov Model

HMM is used to solve three problems. Two of them are related to pattern recognition, which are finding the probability of an observed sequence for a given HMM and finding the sequence of hidden states that most likely generates an observed sequence. Another one is generating a HMM for a given sequence of observations.

3 Semantic Cluster and Text Serialization

HMM is suitable for processing sequence data such as speech signal[1]. However, sequences from document are always so long that the evaluation in HMMs becomes quite difficult[2]. In this part, we use dimension reduction algorithm to process the document[4,5], which can not only reduce the dimension but also keep the categorization information as much as possible.

3.1 Building Semantic Cluster

A semantic cluster is a set of words selected from the document in the training set[6]. A semantic cluster related to N categories is called N-order semantic cluster.

For a cluster with lower order, it can have more categorization information[3]. The following diagram shows the relationship between semantic cluster and training set.

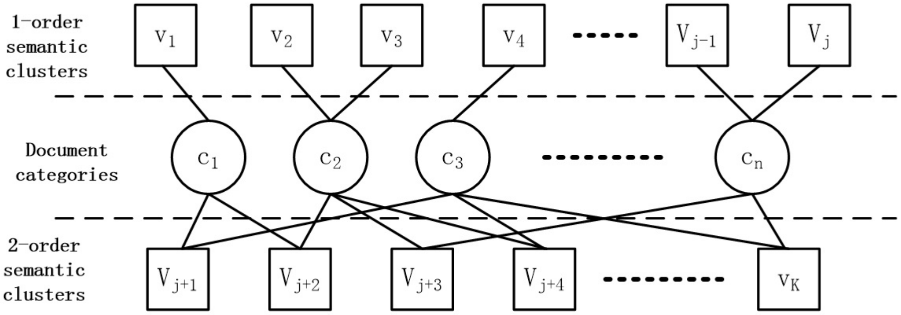


Fig. 2. A 2-order semantic cluster model

In the figure, $C = \{c_1, c_2, \ldots, c_n\}$ is a corpus where c_k is the number k class in training set. $T = \{t_1, t_2, \ldots, t_m\}$ is a word set where $t_i = \{w_{i1}, w_{i2}, \ldots, w_{in}\}$ is an n-dimension vector. We build the semantic cluster as the following steps:

1. Use TF-IDF to compute the weight of t_i in c_k, as below:

$$w_{ik} = tf_{i,c_k} * \log \frac{|D|}{|\{d' \in D \mid t_i \in d' \in c'_k\} + 1|} \tag{1}$$

Where tf_{i,c_k} in the right expression is the number of documents in c_k that contain the term t_i, the other part is the inverse document frequency of t_i.

2. Cluster the word set T. We use hierarchical cluster with cluster iteration threshold K defined as below:

$$K = \sum_{i=1}^{m} C_n^i \quad (2)$$

Where n is the number of categories and m is the biggest order in all the clusters. We assume that each category has an m-order cluster, the best cluster result is less than K, so we can have more cluster iterations at normal situation.

3. Obtain a mapping relationship between word set T and semantic cluster V, as $T \rightarrow V = \{v_1, v_2, \ldots, v_K\}$. This semantic clusters provide the information needed by text categorization.

3.2 Text Serialization

The semantic cluster sequences can be related to Markov Process in the sense that different semantic clusters correspond to different states.

In Table 1, there are 3 different categories including Space, Energy and Electronics, which is from FUDAN corpus[7]. Following the steps in section 3.1, we can build 31 semantic clusters indexed by sequence number. The sequences in category of Space obey some obvious rule which the No.28 cluster has a majority frequency. In the category of Energy, sequences obey some similar rule in Space but No.3 cluster has the majority frequency. Category of Electronics has a different rule which No.2 cluster has the majority. The marked sequences are the documents that usually has common content, the fourth document of Electronics is about electromagnetic waves and space communication, so it has a lot common words with documents from category of Space.

Table 1. Semantic Cluster Frequency Descendent in Different Categories

Category	Semantic Cluster Frequency Descendent Sequence					Exception
Space	28	2	20	21	17	
	28	21	2	25	20	
	28	2	3	17	25	
	28	3	25	2	30	
	28	30	25	4	2	
Energy	3	27	28	2	23	
	3	28	20	23	11	
	28	3	2	10	1	Yes
	25	28	3	1	30	Yes
	3	11	28	26	27	
Electronics	2	3	28	21	22	
	2	3	28	21	27	
	2	3	28	17	24	
	28	21	2	25	3	Yes
	2	3	28	7	17	

4 Text Categorization by HMMs

4.1 Classifier Training

In this part, we usually use Baum-Welch algorithm for training in HMM, as below:

1. Build a mapping table of terms to semantic clusters:

$$T \rightarrow V = \{v_1, v_2, \ldots, v_K\} \tag{3}$$

2. Get the text sequences by using the mapping table:

$$Seqs = \{S_1, \ldots, S_i, \ldots, S_n \mid S_i = \{s_1, \ldots, s_m\}\} \tag{4}$$

3. Use Baum-Welch algorithm for training in HMM, then get the whole classifier.

4.2 Classification Steps

Classification steps are as below:

1. According the initialization vector π, calculate the local probabilities of all states while $t = 1$:

$$x_1(j) = \pi(j) b_{jk_1} \tag{5}$$

2. Then calculate the local probabilities recursively while $t = 2, \ldots, T$:

$$x_{t+1}(j) = \sum_{i=1}^{N} (x_t(i) a_{ij}) b_{jk_t} \tag{6}$$

3. In the end, sum all the local probabilities at $t = T$:

$$\Pr(Y^{(k)}) = \sum_{j=1}^{N} x_T(j) \tag{7}$$

Forward algorithm is used to find the probability of an observed sequence for a given HMM. It takes advantage of recursion in the calculations to avoid exhaustive calculation of all paths in the execution trellis. The whole SC-HMM classifier contains n HMMs (n is the number of corpus categories), we evaluate $Y^{(k)}$ in each HMM and classify a document by the evaluation value:

$$c = \arg\max\{\Pr_i(Y^{(k)}) \mid 1 \le i \le n\} \tag{8}$$

5 Experiments

FUDAN corpus is a widely used data set. Here, we pick 1000 documents from 20 categories uniformly. And, macro Precision, Recall and F1 value are used to evaluate the result. As a comparison, the classic Naïve Bayes classifier and SVM (by LibSVM[8]) classifier is also evaluated.

The macro performances are shown in Table 2. Naïve Bayes classifier incur loss in high-dimensional space than the other 2 classifiers. Macro_F1 value of SC-HMM is a bit higher than SVM.

Table 2. Comparison by Macro Results

Algorithm	Macro Precision%	Macro Recall%	Macro F1%
Naïve Bayes	72.26	69.80	71.35
SVM	83.67	81.48	82.37
SC-HMM	86.42	80.50	83.00

SVM and SC-HMM are evaluated in different categories, as shown in Figure 3. Here, X-axis is the index of the categories. The curve of F1 value difference in Figure 3 shows that SC-HMM classifier has better performance in the categories of big documents, but in the categories of small documents SVM classifier has better performance. So we found SC-HMM classifier has a better information utilization in large documents and the semantic cluster could keep categorization information and remove unrelated redundancy, which also indirectly improve the utilization of information.

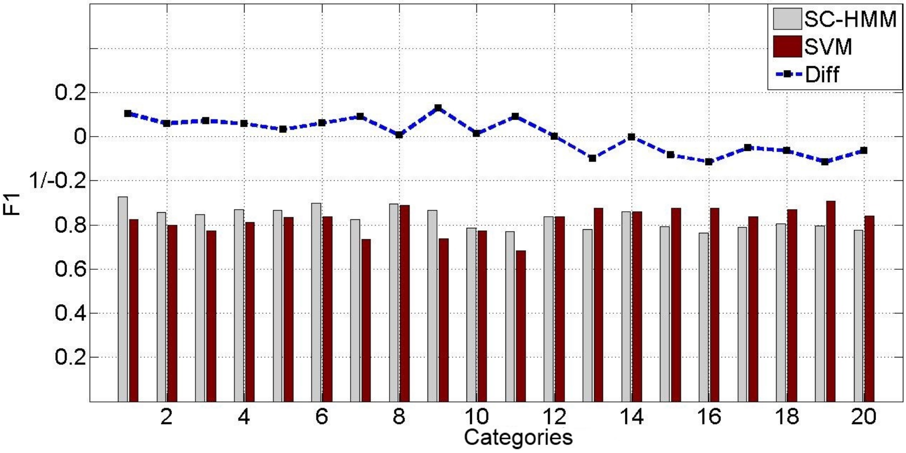


Fig. 3. Comparison of F1 value in different categories

Figure 4 shows the comparison between SC-HMM and HMM in different number of categories. First, it can be seen that with the increase in the number of categories the macro_F1 of SVM has decreased significantly, while the performance of the SC-HMM is more stable. The SC-HMM has a better performance when the number of categories is bigger than 16. When the number of categories is bigger than 2, the SVM classifier is constructed by several sub-classifiers that support only 2 categories. So with the increasing of categories, the number of sub-classifiers increased as superliner growth ($k = C_n^2$), and the complexity of training also increased rapidly. The SC-HMM is constructed by several HMMs as the number of categories, so the complexity grows linearly.

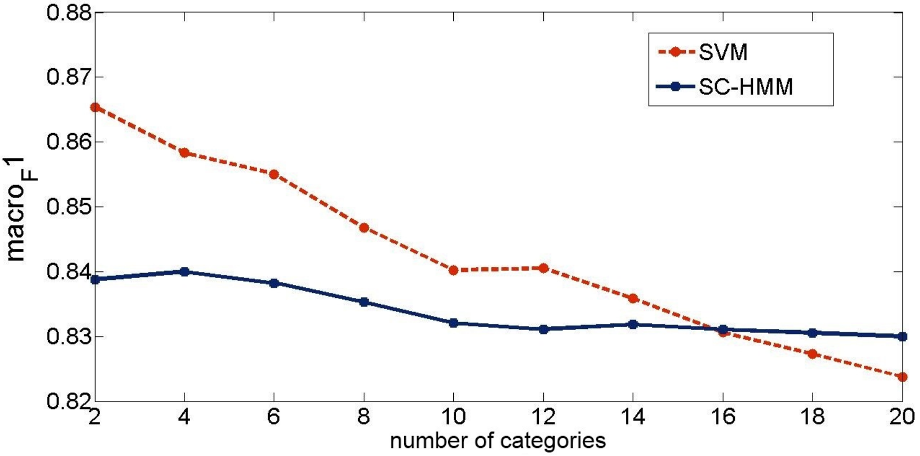


Fig. 4. Comparison of F1 value by increasing categories

As the results shown by experiments, SC-HMM classifier has a better performance than SVM in large vocabulary categories. And as the number of categories becomes larger, SC-HMM has a better performance and stability.

6 Conclusion

In this paper, we propose a new text categorization algorithm based on semantic cluster and Hidden Markov Model. At first we discuss HMM in NLP and its application in text categorization. Next, we propose a semantic-cluster based text serialization method for HMM building. Finally, we propose a new SC-HMM algorithm which overcome some defects of the SVM, which is that the growth of the classification categories number will give a sharp increase to complexity of classification and a decrease to classification accuracy. The simulation shows that SC-HMM is more stable and has better performance in large vocabulary categories.

In the future, we will continue research on: (1) further study of different term weighting schemes and their impact on building semantic clusters; (2) further evaluation of SC-HMM with larger corpus.

References

1. Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE 77(2), 257–286 (1989)
2. Feng, Y., Luo, S.-L., Pan, L.-L., Liu, L.-L., Chen, K.-J.: Method of Text Vector Construction based on Concept Cluster. Journal on Communications 31(8A), 44–47 (2010)
3. Li, W.-B., Sun, L., Zhang, D.-K.: Text Classification Based on Labeled-LDA Model. Chinese Journal of Computers 31(4), 620–627 (2008)
4. Vinay, V., Cox, I.J., Wood, K., Milic-Frayling, N.: A Comparison of Dimensionality Reduction Techniques for Text Retrieval. In: Proceedings of the Fourth International Conference on Machine Learning and Applications, Los Angeles (2005)
5. Uğuz, H.: A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. Knowledge-Based Systems 24(7), 1024–1032 (2011)
6. Luo, Q.-M., Chen, E.-H., Xiong, H.: A semantic term weighting scheme for text categorization. Expert Systems with Applications 38(10), 12708–12716 (2011)
7. Chinese Text Categorization Corpus from Fudan University, `http://www.nlpir.org/download/tc-corpus-answer.rar`
8. LIBSVM: A Library for Support Vector Machines, `http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf`

Reversible Data Embedment for Encrypted Cartoon Images Using Unbalanced Bit Flipping

Wien Hong[1], Tung-Shou Chen[2], Jeanne Chen[2], Yu-Hsin Kao[3],
Han-Yan Wu[1], and Mei-Chen Wu[1]

[1] Department of Information Management, Yu Da University, Taiwan
wienhong@ydu.edu.tw, {hanyan.wu0414,mcwu0715}@gmail.com
[2] Department of Computer Science and Information Engineering,
National Taichung University of Science and Technology, Taiwan
{tschen,Jeanne}@nutc.edu.tw
[3] De Lin Institute Technology Computer Science and Information Engineering Department
hebe@dlit.edu.tw

Abstract. In this paper, we propose a reversible data hiding technique to improve Zhang and Hong et al.'s methods on cartoon images. Zhang and Hong et al. exploit the block complexity for data extraction and image recovery. Their methods are efficient for natural images, however, the results are unsatisfactory when applies on cartoon images consisting of large flat area. By unbalanced flipping the bits of pixel groups, the block complexity before and after flipping can be distinguished and thus the error rate can be further reduced. Experimental results show that the proposed method has lower error rate than those of Zhang and Hong et al.'s methods without degrading the image quality.

Keywords: Reversible data hiding, Image encryption, Cartoon image.

1 Introduction

Data hiding is a method to embed secret data into a digital media for secret communication. Data hiding methods can be categorized into two types, namely reversible and non-reversible. The difference between these two methods is the reversible one has the capability to restore the stego image into its original, undistorted state. All the digital media can be used as carrier for carrying data. Digital image is an often seen carrier because it is widely used over the internet.

The first reversible data hiding method was proposed in Barton's pattern [1] announced in 1997. After that, many reversible data hiding method were proposed and had better performance. In 2003, Tian [2] used two pixels as a group, and embedded a bit into each pixel group by expanding the pixel difference. The maximum payload of their method is 0.5 bpp. Ni et al. [3] in 2006 exploited the image histogram and embedded data by shifting the histogram. In 2009, Tai et al. [4] proposed an efficient reversible data hiding method by shifting the histogram with the assistance of a binary tree. In 2012, Hong [5] used the dual binary tree and error

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 208–214, 2013.

energy control to improve Tai et al.'s method. In Hong's method, the number of shifted pixels is significantly reduced. As a result, the payload is greatly enhanced.

Recently, reversible data hiding method were applied on encrypted images. Since the image owner may not want the data hider to known the informative part of the original content, he or she encrypts the image first, and then the data hider embeds data into the encrypted images. Zhang [6] in 2011 firstly introduced the framework of reversible data hiding in encrypted images. In his method, image owner encrypts the cover image using the exclusive-or (XOR) operation, and then the data hider partitions the encrypted image into blocks of the same size. Each block is separated into two disjoint set. According to the embedded data, the 3 LSBs of one set are flipped. The receiver decrypts the image by using the XOR operation, and use the block complexity to determinate the embedded bits and to recover the original image block. Since the complexity of image block is used to estimate the embedded data, some errors may occur due to the indiscrimination of the block complexity. Hong et al. [7] improved Zhang's method by using the side match technique and use a better metric for measuring the block complexity. As a result, the error rate can be further reduced.

In Zhang and Hong et al.'s method, the smooth images have lower error rates than those of complex ones. However, for an image block with totally flat (pixels having the same gray value) such as cartoon images, their method cannot correctly extract the embedded data because the complexity of image block is indistinguishable. In this paper, we propose an alternative method to overcome this problem so that the method can be applied to the images with large flat area.

2 Related Work

In Hong et al.'s method [7], the cover image I is encrypted by using the XOR operation with the encryption key K_e, and the encrypted image I' is then partitioned into blocks of $n \times n$ pixels. Pixels in each block are randomly separated into two disjoint sets s_1 and s_2. The 3 LSBs of pixels in s_1 or s_2 are flipped according to the to-be-embedded bits. The detailed data embedding procedures are listed in the following sub-sections.

2.1 Data Embedding Procedures

Input: Encrypted image I', secret data S, data hiding key K_h.

Output: Encrypted and embedded image I''.

Step 1. Partition the encrypted image I' into blocks of $n \times n$ pixels.

Step 2. According to the data hiding key K_h, pixels in each block are separated into two disjoint sets s_1 and s_2.

Step 3. Use the raster-scan order to visit each block. Let the visited block be $I'_{c,r}$, where the subscript (c, r) represents the position of each block.

Step 4. Extract a bit $b_{c,r}$ from S. If $b_{c,r}$=0, then flip the 3 LSBs of those pixels in set s_1. If $b_{c,y}$=1, then flip the 3 LSBs of those pixels in set s_2. Let the flipped block be $I''_{c,r}$.

Step 5. Repeats Step 3 - Step 4 until all the secret data are embedded. We denote the encrypted and embedded image as I''.

2.2 Data Extraction and Image Recovery

Input: Encrypted and embedded image I'', encryption key K_e, data hiding key K_h.
Output: Cover image I and secret data S.

Step 1. Use the encryption key K_e to decrypt I''. Let the decrypted image be I^d.

Step 2. Partition the decrypted image I^d into blocks of $n \times n$ pixels.

Step 3. Use the raster-scan order to visit the partitioned blocks. Let the visited block be $I^d_{c,r}$.

Step 4. According to the data hiding key, two sets s_1 and s_2 in $I^d_{c,r}$ are determined. Flip 3 LSBs of pixels in s_1. The flipped result is denoted as $H^1_{c,r}$. Similarly, flip the 3 LSBs of pixels in s_2, and we denoted the result as $H^2_{c,r}$.

Step 5. Use Eq. (1) to calculate the complexity f of $H^1_{c,r}$ and $H^2_{c,r}$, and calculate the difference $D_{c,r}$ between them.

$$f = \sum_{u=1}^{x} \sum_{v=1}^{y-1} \left| p_{u,v} - p_{u,v+1} \right| + \sum_{u=1}^{x-1} \sum_{v=1}^{y} \left| p_{u,v} - p_{u+1,v} \right| \tag{1}$$

In Eq. (1), $p_{u,v}$ represents the pixel value at position (u,v), whereas x and y is the block size.

Step 6. Repeat Step 3 - Step 5 until all $D_{c,r}$ are calculated. Sort $\left|D_{c,r}\right|$ in descending order, and let $\left|D'_{c,r}\right|$ be the sorted result.

Step 7. Scan $\left|D'_{c,r}\right|$ and the corresponding blocks, $0 \le c, r < n$. If none of the upper, lower, left and right blocks of the current visited block are recovered, then use Eq. (2) to extract a bit $b_{c,r}$ and go to Step 10.

$$b_{c,r} = \begin{cases} 0 & \text{if } D_{c,r} > 0, \\ 1 & \text{otherwise.} \end{cases} \tag{2}$$

Step 8. If any of the upper, lower, left and right block of the current block has been recovered, then concatenates the side of the recovered block and $H^1_{c,r}$. Follow the same procedure, we obtain $H^2_{c,r}$.

Step 9. Use Eq. (1) to calculate the complexity of $H'^{1}_{c,r}$ and $H'^{2}_{c,r}$. Let the result be $f'^{1}_{c,r}$ and $f'^{2}_{c,r}$, respectively. Then, use Eq. (3) to extract the embedded bit:

$$b_{c,r} = \begin{cases} 0 & \text{if} f'^{1}_{c,r} > f'^{2}_{c,r}, \\ 1 & \text{otherwise.} \end{cases} \tag{3}$$

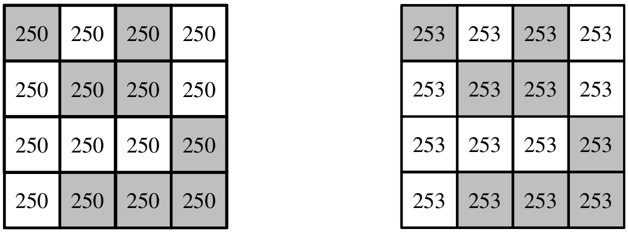
Step 10. If $b_{c,r} = 0$, then $H^{1}_{c,r}$ is the original block. Otherwise, $H^{2}_{c,r}$ is the original block.

Step 11. Repeat Step 7 to Step 10 until all the data bits are extracted and all the blocks are recovered.

3 Proposed Method

In Hong et al.'s method, blocks are separated into two sets. The 3 LSBs of pixels in one set are flipped according to the bit to be embedded. At the extraction stage, the 3 LSBs of pixels in both sets are flipped and use the complexity of flipped block to determine the embedded bit.

However, Hong et al.'s method flips the same number of bits in set s_1 and s_2, which may cause the complexity of the two flipped blocks indistinguishable when all pixels in the original block have the same value. For example, Let all the pixel values in an original block be 250, and the 3LSBs in s_1 are flipped. In the data extraction stage, if the 3 LSBs in s_1 are flipped, we obtain a block with all pixel valued 250 (See Fig. 1(a)). If the 3 LSBs in s_2 are flipped, we obtain a block with all pixels valued 253. Obviously, these two blocks have the same complexity. As a result, Hong et al.'s method cannot tell us which one is the original block.



250	250	250	250
250	250	250	250
250	250	250	250
250	250	250	250

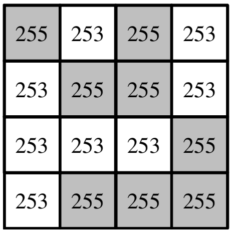
(a) Cover block

253	253	253	253
253	253	253	253
253	253	253	253
253	253	253	253

(b) Flip Block

Fig. 1. Cover block and flip block for Zhang's method

In this section, we propose a new method to avoid the aforementioned ambiguous situation by flipping different bits in different set. The proposed method flips 3 LSBs of pixels in set s_1, and flips only the first and third bits of pixels in set s_2. By doing this, the complexity of the two flipped blocks can be distinguished. For example, Let the pixel values of a 4×4 cover block be 250, and suppose the 3LSBs of s_1 are flipped. In the image recovery stage, if the pixels in s_1 are flipped, the block complexity is 0. However, if the 3 LSBs of s_2 are flipped (See Fig. 2), the block complexity is 28. Obviously, the complexities of these two blocks are distinguishable and thus the embedded data can be correctly extracted. Since the encryption, embedding and extraction procedures are the same as Hong et al.'s method, the detailed procedures can be seen in [7].

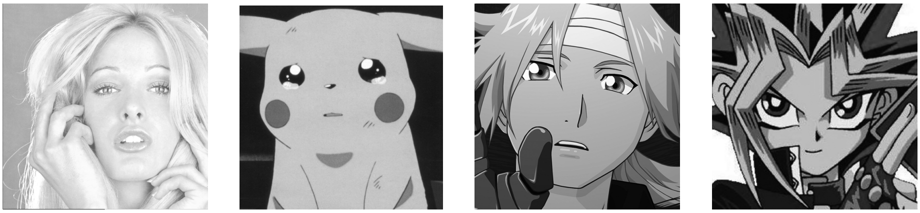


255	253	255	253
253	255	255	253
253	253	253	255
253	255	255	255

Fig. 2. Flip block of proposed method

4 Experimental Results

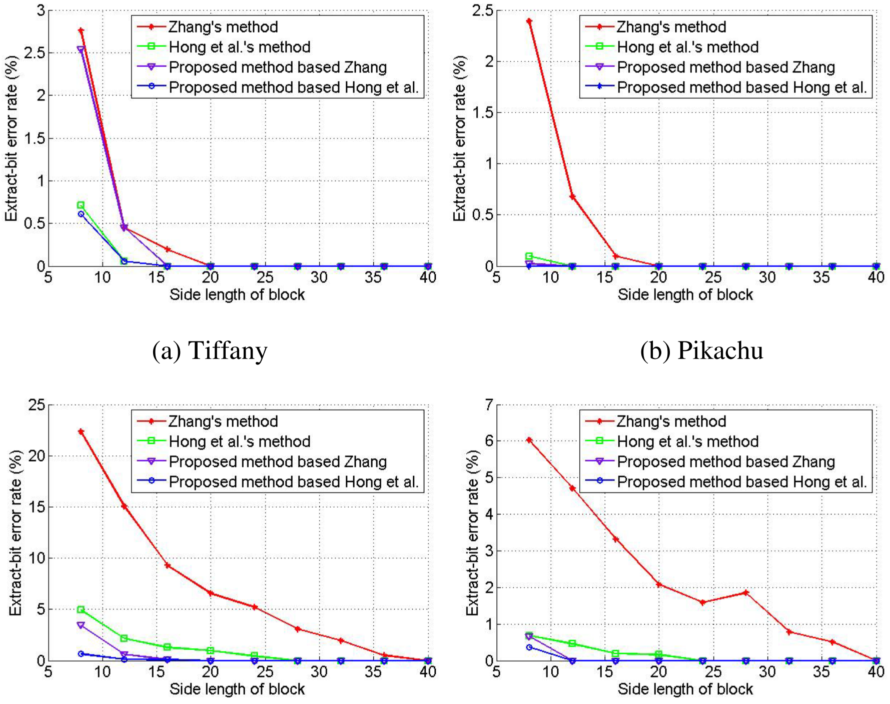
In this section, we perform several experiments to demonstrate the performance of the proposed method and compared the results with those of Zhang and Hong et al.'s method. Four grayscale 8-bit images, including Tiffany, Pikachu, Edward, and YuGi, as shown in Fig. 3, are used as the test images.



(a) Tiffany (b) Pikachu (c) Edward (d) YuGi

Fig. 3. Four testing images

We used the pseudo random number generator (PRNG) to generate the secret data, and use PSNR to measure the image quality. The length of the block was set from 8 to 40. The results were shown in Fig. 4.



(a) Tiffany

(b) Pikachu

(c) Edward

(d) YuGi

Fig. 4. Comparison Results

Fig. 4 shows that the proposed method performs better than those of Zhang and Hong et al.'s method in that the error rate is the lowest. For example, the test image Edward consists of a considerable number of pixels with the same value. As shown in Fig. 4(c), the Zhang's method has the highest error rate. Although Hong et al.'s method improves Zhang's method and has a better performance, the proposed method performs even better than Hong et al.'s method. Note that the proposed method is suitable for images with large flat area. For natural images such as Tiffany shown in Fig. 4(a), the improvement is insignificant.

5 Conclusions

In this paper, we proposed an improved method for reversible data hiding method in encrypted images suitable for images with large flat area. Image of this type often consist of same pixel value in partitioned blocks and thus, the flipping operation provides no clue to distinguish the block complexity. By changing the flipping bits, the block complexity can be easily discriminated and thus the error rate can be further reduced. The experimental results demonstrate that the proposed method effectively reduce the error rate for most of the cartoon images.

References

1. Barton, J.M.: Method and apparatus for embedding authentication information within digital data. U.S. Patent 5 646 997 (July 8, 1997)
2. Tian, J.: Reversible Data Embedding Using a Difference Expansion. IEEE Trans. Circuits Syst. Video Technol. 13, 890–896 (2003)
3. Ni, Z., Shi, Y.Q., Ansari, N., Su, W.: Reversible Data Hiding. IEEE Trans. Circuits Syst. Video Technol. 16, 354–362 (2006)
4. Tai, W.L., Yeh, C.M., Chang, C.C.: Reversible data hiding based on histogram modification of pixel differences. IEEE Trans. Circuits Syst. Video Technol. 19, 906–910 (2009)
5. Hong, W.: Adaptive reversible data hiding method based on error energy control and histogram shifting. Opt. Commun. 285, 101–108 (2012)
6. Zhang, X.: Reversible Data Hiding in Encrypted Image. IEEE Signal Process. Lett. 18, 255–258 (2011)
7. Hong, W., Chen, T.S., Wu, H.Y.: An Improved Reversible Data Hiding in Encrypted Images Using Side Match. IEEE Signal Process. Lett. 19, 199–202 (2012)

A Robust Watermarking Algorithm for 2D CAD Engineering Graphics Based on DCT and Chaos System

Jingwen Wu[1,2], Quan Liu[1,2], Jiang Wang[1,2], and Lu Gao[1,2]

[1] School of Information Engineering, Wuhan University of Technology,
Wuhan, Hubei Province, 430070, China
[2] Key Lab. of Broadband Wireless Networks, Wuhan University of Technology,
Wuhan, Hubei Province, 430070, China

Abstract. As the computer aided design becomes widely used, the copyright violation of two-dimensional CAD engineering graphics urgently needs to be studied. But the current watermarking algorithms have many flaws such as lack of robustness, watermark capacity limitation, and so on. A new algorithm which based on DCT transformation and chaos system is proposed in this paper. The algorithm adopts DCT in watermark preprocessing to reduce the watermark information redundancy, and then classifies the entities of their handles through the chaos system to ensure the security of watermark. The watermark is embedded by modifying entity's line width slightly utilizing HVS characteristics. Experimental results show that the proposed algorithm has good imperceptibility; it's robust against operations such as translation, rotation, scaling, entity addition/deletion and combination of these attacks. The algorithm embeds the meaningful image into 2D CAD engineering graphic; it's useful in practical applications.

Keywords: 2D engineering graphic, DCT, chaos system, copyright protection.

1 Introduction

With the explosive development of the computer technology, digital works are widely used in different fields because it's convenient to preserve, transport and modify. These advantages promote the rapid developments of computer aided design in manufacturing industry, but it brings new problems: copyright violation, illegal theft and malicious modification. Computer aided design in the engineering makes the construction, mechanics, costume design develop rapidly, and for that the two-dimensional CAD engineering graphics become the most popular format among the engineering design field. Digital watermarking technology is an effective method to identify illegal copy, theft and modify. It's important copyright protection method [1]. Currently, Huang and Peng proposed a capacity variable watermarking algorithm for 2D CAD engineering graphics [2]. It is robust against geometric transformation, entity addition and deletion, but it is non-blind. A blind watermarking algorithm for 2D CAD engineering graphics based on entity substitution and the mean value of wavelet coefficients was proposed in [3]. It is tolerant to rotation, translation, scaling

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 215–223, 2013.

and graphic addition, but it enlarges the electronic file after watermark embedded. Peng and Sun proposed several algorithms embedded the watermarking into 2D CAD engineering entity's line, color and other physical characteristics [4,5], but they don't have robustness against entity addition and deletion.

2D CAD engineering graphics belong to vector diagram; it is a kind of object oriented images, which record the entities position accurately through their coordinates [6]. As 2D CAD engineering graphics has small information redundancy, its watermarking capacity is small [7]. It's more difficult for 2D CAD engineering graphic than other digital works to embed watermark.

2 The Proposed Algorithm

2.1 Preliminary Knowledge

2.1.1 Chaos System

Chaos System is a kind of certainty, irregular process in nonlinear system. Logistic mapping:

$$x_{n+1} = \lambda x_n (1 - x_n) \qquad \lambda \in [0,4] \quad x_n \in (0,1) \tag{1}$$

Where λ is a parameter. When 3.569945< $\lambda \leqslant 4$, the sequence generated by the chaos system are non-cycle, convergence, not related. Chaos system is very sensitive to initial value. When λ=4, the mean value of the logistic sequence is 0.5[8].It's very sensitive to the initial value. Different parameters and initial values could produce irrelevant sequences. If attackers don't know the model and parameter, it is impossible to crack.

2.1.2 Discrete Cosine Transformation

Discrete Cosine Transformation (DCT) is one of the most important methods for image processing. It can transform original image from spatial domain to another domain and highlight some characteristics of the image.

Most DCT coefficients of the image are close to zero after the DCT transform. One character of DCT is the greatest energy of the image are concentrate in the upper left corner. The upper left corner reflects the original picture's low frequency data while the lower right corner reflects the high frequency, and most energy of the image always centralized in the low frequency part [9].

2.1.3 Human Visual System

Human Visual System (HVS) masking characteristic mainly displays in three aspects: brightness, frequency rate and color perception [10]. In this paper, we mainly discuss the perception about line width varying. Low and Maxemchuk proposed that if the line width changes no more than 2% (0.176mm), the naked eye cannot feel any differences [11,12]. For example, figure 1 shows three lines with different width, A is 0.15mm, B is 0.20mm, C is 0.22mm, and people cannot feel any differences between them.

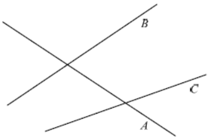


Fig. 1. Three lines of different widths

In Auto CAD2006, line width is ranging from 0.00mm to 2.11mm, there are 24 levels in total. The line width difference between adjacent 2 or 3 levels is negligible; the minimum line width difference is 0.02mm. According to HVS characteristics, if we change line width by just increasing or decreasing one level, it is imperceptible for watermark embedding.

2.2 Generation of Watermark

In the practical application, the watermark should be a meaningful image, not a set of numbers. The 2D CAD engineering graphics have small redundancy to embed watermark, existing algorithms always embed a set of numbers as the watermark. Considering the 2D CAD engineering graphics has small redundancy, watermark preprocessing should be taken to decrease the embedded information. Following measures have been taken to achieve the goal improving the performance of the algorithm: First, DCT is used to reduce the redundancy of the watermark image by transforming the watermark image to watermark sequences. Then, chaos system is used to scramble watermark sequence which improves the security of the algorithm.

Specific proceeds are as follows: Assume the grayscale watermark image size is 64×64, it is divided into small blocks whose size is 8×8, and there are 64 groups in total. Then apply DCT to every small block, and after affined transformation extract ten numbers in the upper left corner to acquire DCT sequences. Because the numbers of DCT sequence are small, the numbers should be expanded to integers in the interval of (0,255) by same multiples for the following steps. Through this transformation, the information of grayscale watermark image decreases greatly. This character is helpful to embed meaningful image into 2D CAD engineering graphics and remedy the restriction of information capacity.

Considering the robustness of the watermarking, chaos system is used to scramble watermark for encrypting. Assume the secret key *K*, it is only known for the owner. In general circumstances, the key is an integer, but the initial value of logistic mapping is a number in the interval of (0, 1), so the linear conversion should be applied to lessen the key to get the initial value X_0. Here the parameter λ equals 4, as 2.1.2, in this situation the mean value of the logistic sequence is 0.5, it is hard for attackers to get the key information. After the logistic mapping, we get the chaos sequence, all the numbers in chaos sequence are in the range of 0 and 1. Then apply range conversion to chaos sequence transforming all the numbers in the interval of (0,255). Finally, apply *xor* operator to the chaos sequence and the DCT sequence to generate the watermark sequence. Through these steps the original watermark image has been encrypted, the security of the algorithm has been promoted.

The whole block diagram of watermark generation is illustrated in Figure 2.

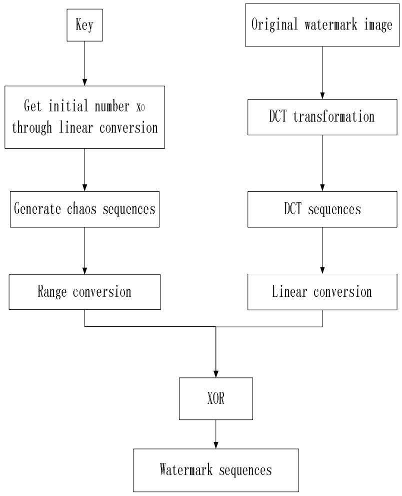


Fig. 2. Block diagram of watermark generation

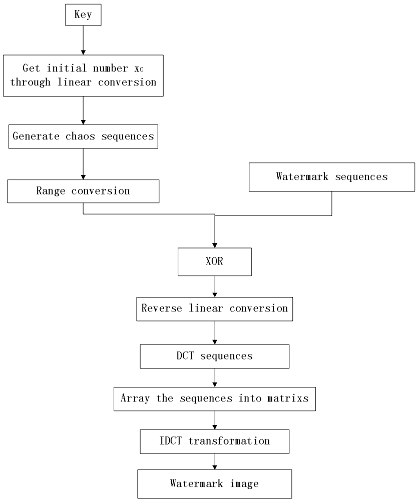


Fig. 3. Block diagram of watermark recovery

The process of restoring the watermark is the inverse procedure of watermark generation. After extracting watermark sequence, use the same key K to get the chaos sequence, then do *xor* with watermark sequence to acquire DCT sequence, apply inverse DCT to it and restore the watermark image. The whole block diagram of watermark image recovery is illustrated in Figure 3.

2.3 Watermark Embedding

The CAD engineering graphics consist of entities and every entity possesses a unique parameter, handle. Handle is the only parameter won't change when inserting pattern interpolation to engineering graphics or doing external reference. Considering the character of handle, taking it as the basis of grouping could guarantee the robustness of the watermark.

The watermark embedding algorithm is presented as follows:

Step 1. Acquire the handle of each entity in the 2D engineering graphics G, assume the number of entity is n, H_i means the handle of the *i-th* entity, where $H_i(1 \le i \le n)$;

Step 2. Transform K to get a number x_i^0 in the range of 0 and 1 as the initial value of the logistic mapping, and a positive integer T_i that it should be more than or equal to 30 as the iteration. Here, the number 30 is to make sure that $x_i^{T_i}$ will iteratively approach to the chaotic area. After the process, a chaos sequence $L_i\{L_1, L_2, ..., L_n\}(1 \le i \le n)$ is generated.

Step 3. Transform the orders of the chaos sequence L_i according to their value, and write down the coordinates q.

Step 4. Rearrange the handles' order as the coordinates q as step 3 to get the scrambling handle sequence $H_i'\{H_1',H_2',...,H_n'\}(1 \le i \le n)$.

Step 5. Divide all the n entities into M small groups according to the scrambling sequence H_i' . The number entity contained in each group should be equal.

Step 6. Embed watermark sequence $C=\{c_1,c_2,...,c_M\}$ into entities which are grouped into M small groups. Embed watermark information $c_i(1 \le i \le M)$ in entities of *i-th* group through modifying the width of entity in this group. Specific steps are: If c_i is 1, the entity's widths of the whole *i-th* group increase one level. While c_i is 0, the entity's widths of the whole *i-th* group keep unchanged.

Step 7. After embedding all the watermark sequence, get the watermarked 2D CAD engineering graphics *G'*.

2.4 Watermark Extraction

Watermark extraction means recover the watermark by extracting it from 2D CAD engineering graphics; it is the inverse process of embedding watermark. The specific procedure is as follows:

Step 1. With the key *K*, we group entities in 2D CAD engineering graphic *G'* following step 1 to step 5. Because of the invariance of entity's handle in engineering graphics, the *M* groups should be the same as foregoing groups of graphic *G*.

Step 2. Extract watermark information embedded into entities of *i-th* group. Specific steps are as follows: Assuming that the *i-th* group is consist of *j* entities, representing as $p_k\left(1 \le k \le j\right)$. Compare the actual width l_{p_k} of p_k with the standard width l_{p_k}' of the layer it belongs to. If $l_{p_k} > l_{p_k}'$, then $c_i\left(k\right)=1$, If $l_{p_k} = l_{p_k}'$, then $c_i\left(k\right)=0$. After this step, we get *j* values. Compare the numbers of 0 and 1 in *j* values of $c_i\left(k\right)$, while 0 is the majority, $c_i=0$; if not, $c_i=1$.

Step 3. The information extracted from all the entities of *M* small groups form watermark sequence $C=\{c_1,c_2,...,c_M\}$.

Step 4. Use the watermark sequences $C=\{c_1,c_2,...,c_M\}$ to restore the watermark image as the block diagram of Fig 3. First get the chaos sequence according the secret key *K*, do xor with the watermark sequence acquire the DCT sequence. Then reconfigure the DCT sequence into matrix through taking out every ten numbers orderly from the DCT sequence, array these ten numbers into the upper left corner of the 8×8 matrix filling the other positions with 0. At last, put all the 8×8 matrixes orderly to get the whole big matrix, do IDCT to restore the watermark image.

3 Experimental Results and Analysis

Experiments have been carried out with Windows XP SP2, MATLAB 2007, AutoCAD2006 Professional, DWG Direct 4.0. Fifty different 2D CAD engineering graphics in DWG format are used in experiment. Fig 4 (a) is an example; it is the original 2D CAD engineering graphic. The corresponding watermarked version generated with the proposed watermark embedding method is shown in Fig 4 (b). Comparing the Fig 4 (a) and Fig 4 (b), the proposed algorithm meets the requirement of the imperceptibility from the human visual system. In the experiment, the parameters are chosen randomly as follows: The key K =8525, and applying linear conversion to the K to get the initial value of the logistic mapping x_0 = 0.52540. Fig 5(a) is the watermark image which size is 64×64. Fig 5(b) is the watermark image restored after DCT transformation, with this step the watermark information data is greatly reduced, but there is little different between it and the original watermark image. Fig 5(c) is the image that is logistic mapping result *xor* with Fig 5(b). Seen from the result, the watermark image had been unreadable after the watermark *xor* with the logistic matrix. It's hard for attacker to recover. Fig 5(d) is the extracted watermark image from the embedded engineering graphic. From the result we know the proposed algorithm is reversible.

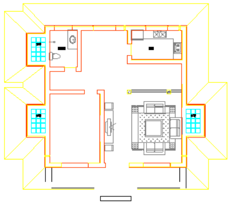


Fig. 4(a). The original 2D CAD graphic

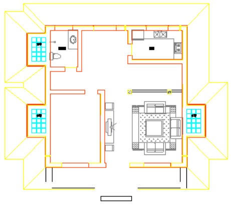


Fig. 4(b). The watermarked 2D CAD graphic

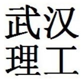


Fig.5(a). Watermark image

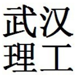


Fig.5(b). Watermark image after DCT

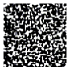


Fig.5(c). The image encrypted after logistic matrix

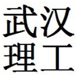


Fig.5(d). Extracted watermark imag

3.1 Correlation Analysis

To accurately calculate the differences among original watermark, watermark after DCT and the extracted watermark, NC (Normalized Correlation)[19] is used to evaluate their similarity. The formula is as follows:

$$NC=\frac{\sum_{i-1}^{N} w(i) * w'(i)}{\sqrt{\sum_{i-1}^{N}\left(w(i)\right)^2} * \sqrt{\sum_{i-1}^{N}\left(w'(i)\right)^2}} \tag{4}$$

Where $w(i)$, $w'(i)$ represents two different watermark that need to be compared. The value is closer to 1, the higher similarity between the two watermarks.

Do experiments with the original watermark, extract 28, 21, 15, 10, 6 numbers of the upper left corner from DCT matrix respectively and calculate the NC between original watermark and watermark after DCT transformation, the computation result shows as Table 1. The result shows extract 10 numbers or more could meet the application expectation, the original watermark and watermark after DCT transformation have high similarity. In following experiments, choose 10 numbers as the extracted data quantity to restore the watermark after DCT transformation.

Table 1. Experimental results of NC

Extracted numbers	28	21	15	10	6
NC	1.000	1.000	0.9986	0.9426	0.8019

3.2 Robustness Test

Experiments were done to analyze the robustness of the proposed algorithm. The experiments are repeated 50 times on the same 2D CAD engineering graphic but with different translation distances, different intensity of attacks.

As seen from the experimental results related tables, the proposed algorithm is robust against translation, rotation, scaling, and the algorithm could extract the watermark correctly. And the algorithm could resist the random entity addition/deletion attack. The performance can meet the actual requirements for slight editing; therefore the proposed algorithm has good actual application value. But it is greatly affected by continues addition/deletion. The robustness of the proposed algorithm greatly affected by entity deletion, we should optimize this respect to improve its robustness.

Table 2. Experimental results of translation

Translation direction	Testing Times	Pass Times	NC
Upward / Downward	50	50	1
Left / Right	50	50	1
Any Direction	50	50	1

Table 3. Experimental results of rotation

Rotation angle(/°)	Testing Times	Pass Times	NC
+30/-30	50	50	1
+45/-45	50	50	1
+90/-90	50	50	1
+120/-120	50	50	1

Table 4. Experimental results of scaling

Scale factor	Testing Times	Pass Times	NC
1/4	50	50	1
1/2	50	50	1
2	50	50	1
4	50	50	1
6	50	50	1

Table 5. Experimental results of entity addition/deletion

Attack method	Testing Times	Pass Times	NC
Add multi-entities randomly	50	50	1
Add multi-entities continuously	50	50	1
Delete single entity randomly	50	24	0.9426
Delete multi entities randomly	50	14	0.8675
Delete single entity continuously	50	23	0.7559
Delete multi-entities continuously	50	10	0.6853

4 Conclusion

In this paper, a robust watermarking algorithm for 2D CAD engineering graphics based on DCT transformation and chaos system is proposed. This algorithm greatly enlarges the watermark capacity, and it meets the requirements of low embedding distortion and high authentication. Experimental results show that the proposed algorithm is imperceptible and reversible, and is tolerant to many incidental operations such as translation, rotation, scaling, entity addition/deletion and combination of these attacks. In our future work, we intend to investigate the feasibility of using other methods, such as self-checking in copyright protection watermarking for 2D CAD engineering graphics in an attempt to further resist continuously attacks and improve the authentication power.

Acknowledgments. This work is supported by National Natural Science Foundation of China under Grant No. 51175389, Key Project of Chinese Ministry of Education under Grant No. 313042, and the Ph.D. Programs Foundation of Ministry of Education of China under Grant No. 20120143110017.

References

1. Yang, Y., Sun, X., Yang, H., Li, C.-T., Xiao, R.: A contrast-sensitive reversible visible image watermarking technique. IEEE. T. Circ. Syst. Vid. 19, 65–67 (2009)
2. Huang, X.F., Peng, F., Deng, T.: A capacity variable watermarking algorithm for 2D engineering graphic based on complex number system. In: Proceedings of International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 339–342. IEEE Press, Harbin (2008)
3. Li, Y., Xu, L.: A blind watermarking of vector graphic images. In: Proceedings of the 5th International Conference on Computational Intelligence and Multimedia Applications, ICCIMA 2003, pp. 424–429. IEEE Computer Society, Washington (2003)
4. Peng, F., Sun, X.M.: Information hiding algorithm for two-dimensional engineering graphics based on characteristics. Comput. Eng. Appl. 43, 54–55 (2007) (in Chinese)
5. Peng, F., Long, M.: Information hiding algorithm for two-dimensional engineering graphics based on chaos system. Acta. Electr. 37, 79–83 (2009) (in Chinese)
6. Cox, I., Miller, M., Bloom, J., Fridrich, J., Kalker, T.: Digital watermarking and steganography. Morgan Kaufmann, San Francisco (2008)
7. Zhou, L., Hu, Y.-J., Zeng, H.-F.: Reversible data hiding algorithm for vector digital maps. Comput. Appl. 29, 990–993 (2009)
8. Huang, R.S.: Chaos and application. Wuhan University Press, Wuhan (2000)
9. Yang, G.B., Du, Q.S.: MATLAB Image and video processing. Publishing house of electronics industry, Beijing (2010)
10. Yang, H.F., Yang, Z.H., Jiang, M.F.: Content Based Image Public Watermarking. In: Proceedings of the Seventh Eurographics Conference on Multimedia, pp. 163–172. Eurographics Association, Switzerland (2004)
11. Brass, J.T., Low, S., Maxemchuk, N.F.: Copyright protection for the electronic distribution of text document. J. P. IEEE 87, 1181–1196 (1999)
12. Low, S., Maxemchuk, N.F.: Performance comparison of two text marking methods. IEEE. J. Sel. Area. Comm. 16, 561–572 (1998)

Detection of Human Abnormal Behavior of the Ship's Security

Fengxu Guan, Xiaolong Liu, and Xiangyu Meng

College of Automation, Harbin Engineering University, Harbin, 150001, China

Abstract. The ship's security depends on the patrol, which is difficult to ensure ship safety in real-time. In order to secure ship more safety, the intelligent video surveillance technology is applied to the ship. Firstly, the background model is established through codebook algorithm, then the movement target of the human are detected accurately. Secondly, the characteristics of the human body are extracted through HU invariant moments. By similarity matching with the abnormal behavior template and the feature of aspect ratio of the human body, the abnormal behavior of the human body is detected. Finally, the experimental results show that this algorithm can be achieved very well. It has obvious advantages on frame difference algorithm and mixed Gaussian algorithm. In the actual environment of anchoring ship, the abnormal behavior of the human body is detected effectively.

Keywords: codebook model, HU invariant moments, behavior detection, abnormal behavior, OpenCV.

1 Introduction

As everybody knows that the berthing ship's security depends on standing guard and patrol on duty, but there are a lot of hidden dangers in usual work. For example, the patrols can not find the frogman sneaking into the ship at any time. Therefore it's very necessary that the intelligent video surveillance technology should be used in the ship's security. Combining the patrol, the intelligent video surveillance technology will bring great benefit, and it can make the patrols' work more simple and effective.

In this paper, we mainly do some research on two points. Firstly, with the Codebook algorithm [1], the background model of the ship's physical environment is created, and then the movement target of the human body is detected accurately. Secondly, extract the characteristics of the human body with HU [2,5] invariant moments, and then match similarity to the template of normal behavior to realize the human abnormal behavior detection.

2 Human Target Detection Based on Codebook Model

2.1 Background Model of Codebook

The YUV color space is adopted as the modeling color space of the codebook. In the YUV color space, the value of each pixel can be represented by using a sequence

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 224–231, 2013.

$X = \{X_1, X_2, \ldots, X_N\}$. A codebook is built for each sequence in codebook model $C = \{c_1, c_2, \ldots, c_L\}$, which contains L code word C_i . The code word is an eight-element group: $c_i = < \breve{Y}_i, \hat{Y}_i, \bar{U}_i, \bar{V}_i, f_i, \lambda_i, p_i, q_i >$, Among: $\_\breve{Y}_i, \hat{Y}_i$ represent the maximum and minimum Y component of the code word; $\bar{U}_i, \bar{V}_i$ represent the average value of the codeword corresponding pixel point U and V; f_i represents the number of pixels which are included in this code word; λ_i represents the maximum time interval that the pixel is not be matched; p_i and q_i respectively represent the first and the last successful match time.

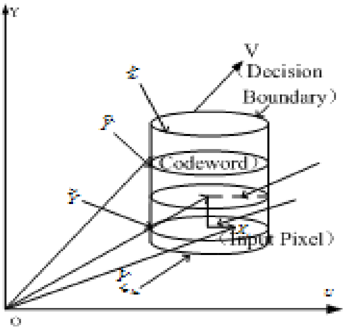


Fig. 1. Codebook matching model under the YUV color space

In Fig 1, codebook model under the YUV space is a cylinder which is perpendicular to the Y-axis, it makes background segmentation simply. For the input pixel point X_i:

$$colordist = (U_t - \bar{U}_t)^2 + (V_t - \bar{V}_t)^2 \leq \varepsilon \qquad brightness = Y_t \in \_Y_{low}, Y_{hi}^{-}$$

$$\text{Among: } Y_{low} = \alpha \hat{Y}, Y_{hi} = \min\{\beta \breve{Y}_i, \hat{Y}_i / \alpha\} \tag{1}$$

To complete the modeling of the first n-frames video, some works are need. Firstly, initialize the codebook model, and then convert the pixel value of every frame t from RGB space to YUV space; finally, match with the code established by the formula (1). If the match is successful, update the codebook; if unsuccessful, a new codeword must be rebuilt.

The trained codebook not only contains the background pixels, but also contains some changed foreground pixels and noise point, which causes the codebook redundant. Maximum time λ is used to remove redundant codeword, and a threshold value T is set. When λ is less than this threshold value T, codeword is retained, and otherwise is removed. M is the final codebook, such as formula (2):

$$M = (c_m \mid c_m \in C, \lambda < T) \tag{2}$$

2.2 Foreground Detection Based on Codebook Model

According to the process of codebook background modeling above, a codebook model can be established under the Y, U, and V coordinate component by setting the background model frame number N and the threshold value T, which contains main background information. Foreground and background can be segmented for the subsequent video frames, and the foreground object is extracted effectively.

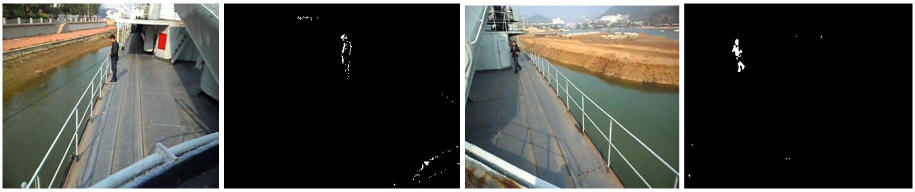
Each read frame of the video image is divided to extract the background, based on the establishment of good codes present model, and the division method is in accordance with equation (1). By changing the color matching parameters ε appropriately, it can be found whether background model matches the codeword. Discriminating manner: if code word matches the model, as foreground; not matches model, as background. In order to accelerate the segmentation speed, end the frame ahead when a codeword is matched.

In order to adapt the real-time changes in the background, it is necessary to set a time T' to update codebook model, which is used to update matched frequently code according to formula (1).In addition, it is needed to delete the code word which has not been visited exceed T_{late} time according to the formula (2).

There are videos which are screened about the crew walking on the ships normally, falling into the water accidentally, and the enemy attacking aboard. According to the code of the detection algorithm, the videos are handled by setting the background modeling frames N as 50, threshold T as 25, time parameters T as n-N/5 (n is the current number of frames, N is model training frame).

In fig 2a), when the ships anchor at the pier, the ships crew that stays a short time around the railing is misjudged as the background by the detection algorithm of the adaptive codebook. The last detection result is a fast disappearance of the contour, as it is shown in fig 2b).

In fig 2(c) and 2(d), when the target works into the shadows, since codebook algorithm sensitivity to light, the target integrity is affected, which results in fracture.



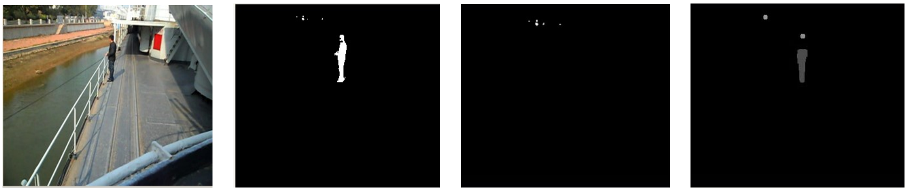
a) Original image b) Target as background c) Complete target d) Fracture target

Fig. 2. Detected results of codebook algorithm

The test results of the video above will cause misjudgment of later human behavior.

2.3 Setting Background Updating Parameter and Optimizing Results

Based on the background model M which has been established, the moving target detection test in the last section, codebook model can be updated by setting update time, removal noise and front sights which are added in modeling. The result is updating the background at the beginning of the video surveillance more frequently. After regular time monitor, update frequency becomes very slow. Typically a codebook modeling process requires only 50 frame video to include better background information. The time required to update the background process is less than the time of the modeling of the codebook. The normal video frame rate is 25-30fps, which shows that the update rate is probably less than 2s.After adding the running time of the program, the time of updating a codebook model does not exceed 2s.In this way, we can consider extending the update time T. The number of frames is set to 10 minutes, which means T=15000. Couple a restriction with codebook model update, which is when there is not a target, false detection ratio of foreground and background is set as FB, FB=area (foreground)/area (background).Take FB threshold value as 0.1. When FB<0.1, according to updated T, update the background mode. When FB>0.1,and updating background model, the updating condition takes precedence over the update time of T, so that you can get needs of human movement target detection to study in the actual environment in this article. As shown below, the time of the target stopped is 5 seconds; the effect of detecting will not change. As shown in Figure 3, the target stops 5 seconds in the video, effect of detection of using codebook algorithm don't change, it has obvious advantages on frame difference algorithm and mixed Gaussian algorithm.



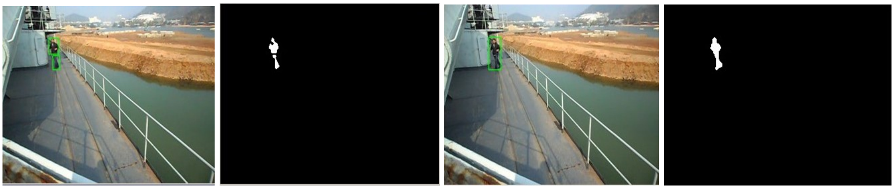
a) Original image b) Codebook model c) Frame difference d) Mixed Gaussian

Fig. 3. Test results after the optimize of parameters

By setting a reasonable parameters T of the background updating, we can guarantee that the target behavior in the video scene is less subject to the influence of background updating.

For generating the breakage of target walking into the shadow, the breaking point can be connected with a morphological closing operation. Using the function cvFindcounters of OpenCV[3], all contours of regions including human target can be found in the binary image. Then calculate all contours' area according to the function cvContourArea. The region of human target can be filled through the function cvDrawCounters and a threshold value A, and then the human target can be extracted.

$$\text{As: Target}=\begin{cases}\text{Set background } 0 & \text{Contour area} < A \\ \text{Set foreground } 255 & \text{Contour area} > A\end{cases}.$$



a) Fracture target b) Repaired target

Fig. 4. Subsequent processing of the test results

In Fig 4, we can draw the conclusion that when target works into the shadows, since the luminance component Y of YUV produces larger changes, the phenomenon of target fracture is occurred in the shadows interchange. Breakage can be repaired well by the closing operation of image, and the small area of interference can be eliminated by the object extraction approach that is mentioned above.

3 Abnormal Human Behavior Detection

3.1 Characterization of Target Behavior

The invariant moments means that image characteristics tend to have the invariant attributes of the translation, rotation and proportion. Describing the characteristics of things or images is the invariant moments' main purpose. Humans using eyes to identify the characteristics of the image is often manifested in the form of "summation", so the invariant moments are integral operation of image elements. After a series of algebraic transformation, MKHU [4], firstly proposed the invariant moments of continuous invariant functions in 1962, which had been used for shape recognition and deducing seven invariant moments [4] of translation, rotation and scale invariance of second-order and third-order, and then it was applied to the automatic characteristic recognition system [5].

Seven HU moments normalized second-order and third-order center moments are as follows:

$$\begin{aligned}
&M_1 = \eta_{20} + \eta_{02} \\
&M_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
&M_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
&M_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
&M_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \\
&M_6 = (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{03} + \eta_{21}) \\
&M_7 = (3\eta_{12} - \eta_{03})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})\left[3(\eta_{30} - \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]
\end{aligned} \tag{3}$$

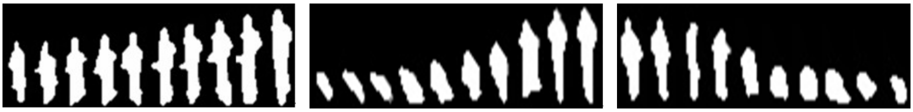
Seen from the expressions: when $x < \overline{x}$ or $y < \overline{y}$, negative numbers maybe appear in this expression. And $M_1 < M_7$ also has a great range of numerical changes. To this end, the absolute value of the data can be used, and data can be adjusted in the logarithmic form. Actually invariant moments [6] are used as follows:

$$\phi_i = \left|\lg\left|M_i\right|\right|, i = 1,2,3,4,5,6,7 \tag{4}$$

It can be proved that seven invariant moments in the case of continuous conditions have the characteristics of translation, rotation and scale invariance.

3.2 Feature Extraction of Human Movement

This paper studies the abnormal behavior about someone climbing the ship or falling out off the ship. Under normal circumstances, the process of the two abnormal behaviors continues 4 to 5 seconds. We take the average amount of each separated by 10 target feature to study. In Figure 5, an abnormal behavior of the process can be described in 10 feature amount of the target physical changes.



a) Normal walking b) Abnormal behavior

Fig. 5. Several characteristics of human behavior

Each characteristic can be obtained by calculating HU moments of the above ,and three groups target characteristics sequence characteristics can be described as $M_i (i = 1,2,3,4,5,6,7)$, then $\phi_i = \{m_1, m_2, m_3, m_4, m_5, m_6, m_7\}$ can be obtained by logarithmic $M_i, \phi_i = \left|\lg\left|M_i\right|\right|, i = 1,2,3,4,5,6,7$. Assume taking the average of 100 frame target characteristics interval of10, then a 10×7 matrix of description of behavior

$$A = (\phi_1, \phi_2, \phi_3, \phi_4, \phi_5, \phi_6, \phi_7, \phi_8, \phi_9, \phi_{10})^T \tag{5}$$

3.3 Detecting Abnormal Behavior of Human Body on Ships

In order to realize the abnormal behavior identification, we must firstly determine a standard template, which is, considering the normal behavior of person on the ships as the standard. Take several groups videos of people walking properly on the ships, the body contour can be obtained by detecting from them. HU moments of 100 frames can be calculated by the body contour characteristics; a 10x7 matrix can be obtained by calculating the average of each separated by 10 frames, and then the average of several groups of normal can be taken as the template.

Calculate hausdorff distance [7] D(A,B) between the groups of normal and abnormal behavior with template, and use the method of average filtering [8].That is, considering the average of each separated by 10 frame as a characteristic, then the similarity metrics of distance is measured by matching characteristic with the template.

Table 1. Comparison of similarity metrics

Normal	1	2	3	4	5	**Abnormal**	1	2	3	4	5
D	3.44	4.25	2.29	1.91	0.67	**D**	8.09	6.47	15.4	8.59	4.38

Through the comparison of the two sets of data, it may also be not enough to distinguish the two acts well. For example, normal 3 and abnormal 5 Max Distance is very close, so it is very easy to make a mistake. The technology which gets rid of the extreme value is used before that every 10 characteristics mean is taken. The method of average filtering which can remove extreme value is used to get rid of the influence of the interference point. Similarity measure [9] is recalculated as follows:

Table 2. Comparison of similarity metrics by new method

Normal	1	2	3	4	5	**Abnormal**	1	2	3	4	5
D	3.31	2.78	2.04	1.82	0.76	**D**	8.84	7.31	16.15	10.48	4.71

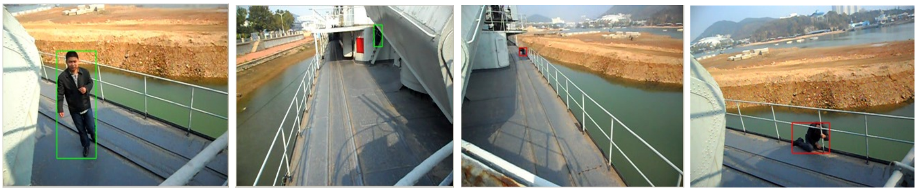
From the comparison to the normal 2 of chart 1, it can be seen that the extremum and average filtering method can deal with larger interference well. Behavioral characteristics of two different human can be distinguished more clearly, but still there is normal behavior1 and abnormal behavior 5 have similar distance value. This combines with the aspect ratio to distinguish the abnormal behavior. For the abnormalities in 5 above, the aspect ratio of the rate of change of the body contour during human behavior can be obtained by calculating body contour high aspect ratio of the rate of change in every 100 video sequence frames.

The change rates V=Height/Width of aspect ratio of human body are shown in Table 3.

Table 3. Comparison of aspect ratio of the rate of change

Normal	1	2	3	4	5	**Abnormal**	1	2	3	4	5
V	1.26	1.12	1.58	1.25	1.56	**V**	2.88	3.12	2.78	3.25	3.86

According to the data in the previous table, similarity measure of the distance threshold value is set to 4 and the high aspect ratio of the rate of change threshold value is set to 2. The above 10 kind's cases can be completely separated, and the detection of human abnormal behavior can be detected. In Figure 6, once criterion condition is met, the abnormal behavior can be detected.



a) Detection of normal behavior b) Detection of abnormal behavior

Fig. 6. Experimental results

4 Conclusions

A program has been designed to detect abnormal behavior in this paper. First, in the case of high accuracy and high efficiency, the algorithm of codebook model has been used to detect the moving target of human body quickly and accurately in the environment of ship mooring at the pier. It has obvious advantages on frame difference algorithm and mixed Gaussian algorithm. Second, HU invariant moments has been used to describe the characteristics of human body, and the normal behavior template library has been established. Third, the Hausdorff distance was combined with the change rate of aspect ratio, using this method, human abnormal behavior in the safety and security of the ship can be detected quickly and accurately.

Experimental results show that the method presented achieves the better recognition effect of the abnormal behavior in the relatively static background. But the problem of poor adaptability has been presented in a dynamic background, and it will to be addressed in future studies.

Acknowledgment. This work was supported by the Fundamental Research Funds for the Central Universities (HEUCF041314).

References

1. Kim, K.: Background modeling and subtraction by codebook construction. In: 2004 International Conference on ICIP 2004, pp. 24–27 (2004)
2. Xu, G.-Y., Cao, Y.-Y.: Action Recognition and Activity Understanding: A Review. Journal of Image and Graphics 14(2), 189–195 (2009)
3. Zhen, L., Qi, Y.: OpenCV Tutorials. 4, 204–285 (2009)
4. Aggarwal, J.K., Cai, Q.: Human motion analysis: A review. Computer Vision and Image Understanding 73(3), 428–440 (1999)
5. Hu, M.K.: Visual pattern recognition by moment invariants. IREF Transactions on Information Theory, 179–187 (1962)
6. Hu, M.K.: Visual Pattern recognition by moment invariant. IRE. Trans on lnf. Theory, 179–187 (1986)
7. Wang, L., Hu, W.-M., Tan, T.-N.: Gait-Based Human Identification. Chinese Journal of Computers 26(03), 353–360 (2003)
8. Chen, C.C.: Improved moment invariants for shape discrimination. Pattern Recognition 26(5), 683–686 (1993)
9. Hao, C., Gang, L.: Analysis and Research of Digital Image Invariance Moment Invariants in Case of Discrete. Chinese Journal of Naval Engineering University 15(4), 89–92 (2003)

HYBit: A Hybrid Taint Analyzing Framework for Binary Programs

Erzhou Zhu[1], Haibing Guan[2], Alei Liang[2], Rongbin Xu[1], Xuejian Li[1], and Feng Liu[1,*]

[1] Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education
& School of Computer Science and Technology, Anhui University, Hefei, China, 230601
{ezzhu,xurb_910,lxj,fengliu}@ahu.edu.cn
[2] Shanghai Key Laboratory of Scalable Computing and Systems,
Shanghai Jiao Tong University, Shanghai, China, 200240
{hbguan,aleiliang}@sjtu.edu.cn

Abstract. For the purpose of discovering security flaws in software, many dynamic and static taint analyzing techniques have been proposed. The dynamic techniques can precisely find the security flaws of the software; but it suffers from substantial runtime overhead. On the other hand, the static techniques require no runtime overhead; but it is often not accurate enough. In this paper, we propose HYBit, a novel hybrid framework which integrates dynamic and static taint analysis to diagnose the security flaws for binary programs. In the framework, the source binary is first analyzed by the dynamic taint analyzer; then, with the runtime information provided by its dynamic counterpart, the static taint analyzer can process the unexecuted part of the target program easily. Furthermore, a taint behavior filtration mechanism is proposed to optimize the performance of the framework. We evaluate our framework from three perspectives: efficiency, coverage, and effectiveness, and the results are encouraging.

Keywords: Binary Taint Analysis, Dynamic Analysis, Static Analysis, Software Flaw/Vulnerability, Security.

1 Introduction

Taint analysis is a form of information-flow analysis which establishes whether values from unauthenticated methods and parameters may flow into security-sensitive operations [1]. As a technique for detecting vulnerabilities in applications, taint analysis can be divided into two categories: dynamic analysis and static analysis [2]. Static analysis is the process of analyzing a program’s code without actually executing it. The technique has low overhead with respect to the utilization of system resources. However, it has the limitation of imprecision when it handles the dynamic structures (pointers, aliases and conditional statements) of the target program [3].

Dynamic analysis analyzes the program at runtime; it is more precise than its static counterpart. Much attention has been drawn to suspicious data tracking with dynamic taint analysis. Taintcheck [4] describes a dynamic-taint based approach to prevent overwrite attacks. Dytan [5] is a generic dynamic taint analysis framework, and can

[*] Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 232–239, 2013.

handle the data flow and control flow in its taint analysis. LIFT [6] provides a way to facilitate monitoring the tainted data when the software is executed. However, the dynamic analyzing techniques suffer from large runtime overhead, and can only detect software vulnerabilities when the attacks have been launched.

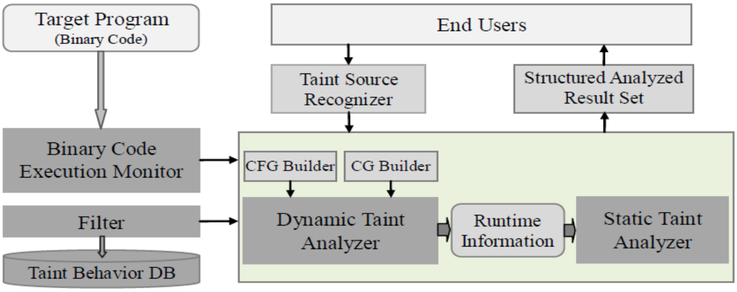


Fig. 1. The Overall Architecture of the HYBit

In order to take advantage of the merits of both dynamic and static analysis, we propose HYBit, a hybrid framework which integrates dynamic and static taint analysis to discover software flaws or vulnerabilities. Since the source code of most software is hard to acquire, our framework is designed to handle the binary code. In this paper, we make the following contributions: 1) propose HYBit, a novel hybrid framework which integrates the dynamic and static taint analysis to track the flaws or vulnerabilities for binary programs; 2) Design a dynamic technique to track the tainted data and their propagation behaviors; 3) Present a static analysis technique to complete the target program by adding the unexecuted part of the target program; 4) Design a novel taint behavior filtration mechanism to further optimize our framework.

2 The Implementation of HYBit

2.1 Framework Overview

As Fig.1 shows, the proposed hybrid framework, HYBit, contains the following main components: the Binary Code Execution Monitor, used to monitor the target program during its execution; the Taint Source Recognizer, used to define suspicious taint sources; the Dynamic Taint Analyzer, used to analyze the target program dynamically and to deliver the required information to the Static Taint Analyzer; the Static Taint Analyzer, used to analyze the unexecuted part of the target program; and the Filter, which is a component to optimize our framework. Our framework is based on CrossBit, which is designed and implemented as a dynamic binary translator, which aims at quickly migrating existing executable code from one platform to another at low cost. We can refer to [7, 8] for more information about CrossBit.

2.2 Dynamic Taint Analysis

During the execution of the target program, the Dynamic Taint Analyzer traces the taint propagation by marking every memory byte or register which was influenced by

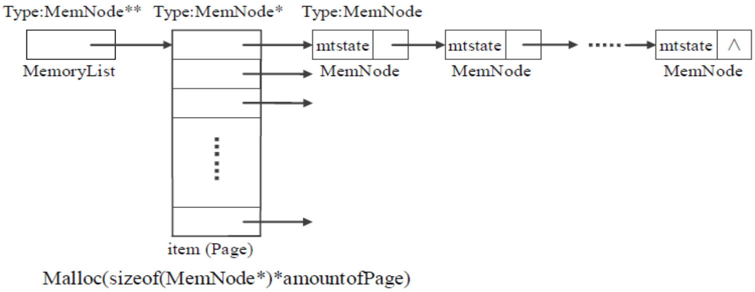


Fig. 2. The Data Structure of Memory Model

the tainted sources. It closely monitors all the marked data, record their dangerous behaviors and finally, report them as potential vulnerabilities.

Taint Source Locating. At the beginning of the dynamic taint analysis, it should locate the taint source in the memory space and registers, which are most likely contaminated by the input data from unsafe channels. In the framework, there are two means to identify taint sources, user-defined or system default. In the user-defined model, end users can specify some execution points (memory area or register) they may be interested in as taint sources. In the system default model, the framework will automatically identify the input from the outside, and take it as suspicious.

Taint Propagation Tracking. The Dynamic Taint Analyzer monitors each instruction which refers to the tainted data to track the taint propagation. It works with the underlying Binary Code Execution Monitor. Given taint sources, dynamic analyzing roles, and the instrumented target program, the Dynamic Taint Analyzer performs taint analysis during the run, and, marks any data that is derived from tainted data as tainted. Actually, this type of information flow tracking is implemented on the intermediate instruction level, and we just need to take care of tainted or untainted states of operands instead of their concrete values. During the process of dynamic taint analysis, the system has to build memory and register models to record the taint states of each memory byte and register.

Memory Model. Since not all the memory locations are tainted during the analysis process, we only need to record the memory addresses that are affected by the tainted sources. In the framework, a chaining hash table (as Fig.2 shows) is used to record the tainted memory bytes.

In the figure, an item corresponds to one page of the target program's virtual space, and the number of memory pages determines the number of items. Each page (the size of a page is multiple of 32 bytes) that corresponds to an item is split into several 32-byte sub-pages, and a sub-page is accommodated by a MemNode. Consequently, a page is saved by several MemNodes that belong to a single chain. There are two fields in the MemNode data structure, a pointer field (next) that identifies the next item, and a data field (unsigned int mtstate) that is used to record the taint states of 32 memory bytes. In the mtstate data field, one bit corresponds to a memory byte. The value of this bit is 1 if and only if the corresponding memory unit is tainted. Since the mtstate is a 32-bit variable, a MemNode can record the taint states of 32 memory bytes. In the model, a MemNode is added to the hash table if and only if at least one of the 32 memory bytes is tainted.

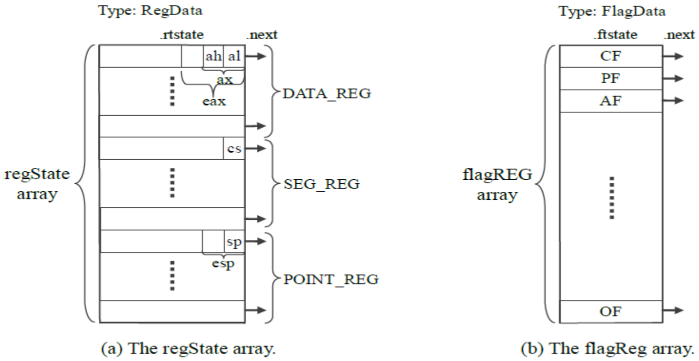


Fig. 3. The Data Structure of Register Model

Register Model. In this model, we use one bit represent the state of a register. There is a special relationship among registers: some registers are part of others. That means, when the state of a register is changed, the state of other registers may be affected. This relationship must be taken into consideration. As shows in Fig.3, we use two arrays to record the taint states of registers, the regState and the flagState.

In regState, Fig.3(a), each element (RegData) incorporates two fields, a pointer field (next) that is used to track the changes of the registers except EFLAGS, and a data field (unsigned int rtstate) that is used to record the taint states of the corresponding register. For the data registers (DATA_REG), the last three bits of rtstate are used to record their taint states. Take the eax register for example, the last and penultimate bits denote the states of al and ah respectively, the last two bits record the states of ax, and the whole three bits specify the state of eax.

In flagState, as shows in Fig.3(b), we use the ftstate (Boolean variable) data field to record the taint states of the flag bits (1 denotes the register is tainted and 0 means untainted). Meanwhile, a pointer field (next) is used to track the changes of each flag.

2.3 Static Taint Analysis

During the dynamic analysis, the CG (Call Graph) of the target program has been built and for each function in the CG, the CFG (Control Flow Graph) of the executed code has also been constructed. Meanwhile, branch points like conditional jumps will be added to the branch list of the function for use in static completion analysis.

For each function call of the target program, the static analyzing module uses the following steps to traverse the branch point list (a kind of runtime information, which initially stores the branch points of the executed part of the function), and meanwhile supplement the unexecuted part of the function: 1) Collects the branch points of the executed part of the function, and stores them in a branch point list P; 2) For each branch point p in P, utilizes b=GetNextBB(p) to get the next unexecuted basic blocks; 3) If the newly derived basic block b is not in B (the basic block set used to accommodate the unexecuted basic blocks), turns to 4), else turns to 2); 4) If b's last instruction is a conditional branch p', adds p' into P, else turns to 5); 5) Adds b into B; 6) Sets a new edge e, which starts from p and ends with b, and stores it into E (the edge set used to accommodate the unexecuted edges); 7) Turns to 2).

After executing the above steps, information (branch points, edges, and basic blocks) about the structure of the unexecuted code is available, and subsequently, the visualized CFG can be generated for the user for manually analysis. By integrating the information about individual functions, we can get the whole structure of the target program. Given such information, the static analyzer is able to apply taint analysis on the unexecuted part of the target program which cannot be reached by the dynamic analyzer.

2.4 Framework Optimization

In this part two approaches, instruction-level filtering and function-level filtering, are employed to optimize the performance of the proposed framework.

Instruction-Level Filtering. In HYBit, we divided the instructions into three categories: 1) instructions which can propagate tainted data, such as LOAD, MOV. When dealing with this type of instructions, we mark any data that is derived from tainted data as tainted. 2) Instructions which cannot propagate tainted data, such as NOP, JMP. This type of instruction does not affect taint propagation, and we can skip them to reduce the work load of the analysis. 3) Special instructions, whose results do not depend on the input, such as the XOR instruction (xor eax, eax). In this case, eax is always set clean (untainted) after the execution of the instruction.

Function-Level Filtering. Actually, a large part of the binary code in software is loaded from system libraries such as Kernel32.dll, MSVCRT.dll and USER32.dll. Since the behaviors of these modules are predictable, the filtration mechanism summarizes the taint effects of these API functions and skips them. In the first place, it has to examine the source code and documents of these APIs, gather their taint propagation information and store all these kinds of information into the Taint Behavior DB.

After the taint effects of API functions are collected, some principles are designed to provide the decision of the code filter for skipping the irrelevant ones: 1) If a function does not do anything in the taint propagation, the taint status of the program doesn't change; 2) If a function can propagate the tainted data, the taint status of the program will remove the source parameter from the set of tainted data; 3) If a function can propagate the tainted data, the destination of the tainting should be marked as tainted and brought into the monitoring of the system.

3 Experimental Evaluation

Fig.4 assesses the efficiency of HYBit by comparing the performance with the native platform. This experiment is based on the SPEC CINT2006 benchmarks on Windows. We normalized execution time (the ratio of our time to native execution time, the native value is always set to 1) of HYBit. The results of the NATIVE, HYBit, and HYBit_NO refer to the execution time of the binary code running on the underlying platform CrossBit directly, running with analysis by HYBit, and with analysis but without the taint behavior filtration optimization.

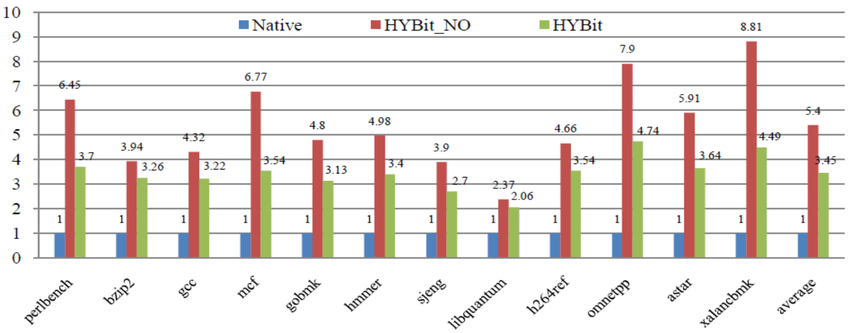


Fig. 4. The Performance Evaluation of the HYBit

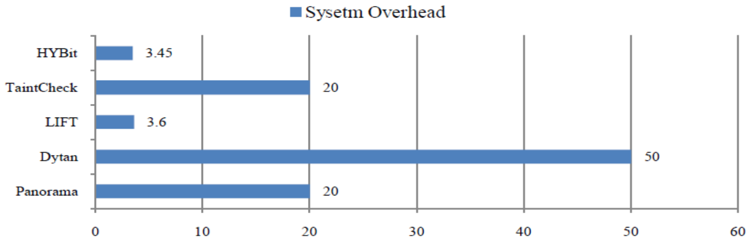


Fig. 5. Overhead Comparison between HYBit and Other Attack Diagnosis Tools

As Fig.4 shows, we achieve the overhead of 5.4 times on average to the native platform without the optimization of taint behavior filtration. However, when the optimization is introduced to the system, HYBit achieves 3.45 times to the native. It can be observed from the figure that the performance of HYBit differs among the target programs. One reason is the structural distinction among these programs. Little time is spent on analyzing library functions that can be summarized and filtered by the optimization component of the framework, so the structure of the .exe module of the target program is an important factor in the performance of our system.

Fig.5 shows the average overhead comparisons between HYBit and other popularly used binary-level attack diagnosing tools: TaintCheck [4], Dytan [5], LIFT [6], and Panorama [9]. Compared to tools such as Panorama (slowed down the target programs by 20x on average), Dytan (50x), LIFT (3.6x) and TaintCheck (20x), HYBit inflicts much lower runtime overhead (3.45x).

Fig.6 evaluates the coverage of HYBit by counting how much code of the target program can be covered by the framework. This experiment introduces the SPEC CINT2006 benchmarks to test four kinds of results: Total BB (total number of basic blocks in the target program), Executed BB (number of basic blocks which are analyzed by the dynamic component), Complemented BB (number of basic blocks analyzed by the static component), Missed BB (number of basic blocks which are not recognized by HYBit). The Coverage Rate can be calculated as Equation 1. The value of Total BB is the sum of executed BB, complemented BB and missed BB.

$$\text{Coverage Rate} = \frac{\text{Executed BB} + \text{Complement ed BB}}{\text{Total BB}} \tag{1}$$

From Fig.6 and Equation 1, we can derive that the Coverage Rate of HYBit reaches 90% on average. There are two reasons why these indices are not 100%: 1) HYBit cannot handle an indirect control transfer which depends on the context; 2) the information that is used for the algorithm of static completion is acquired from the dynamic analyzer, however, this kind of information only holds for a specific instance.

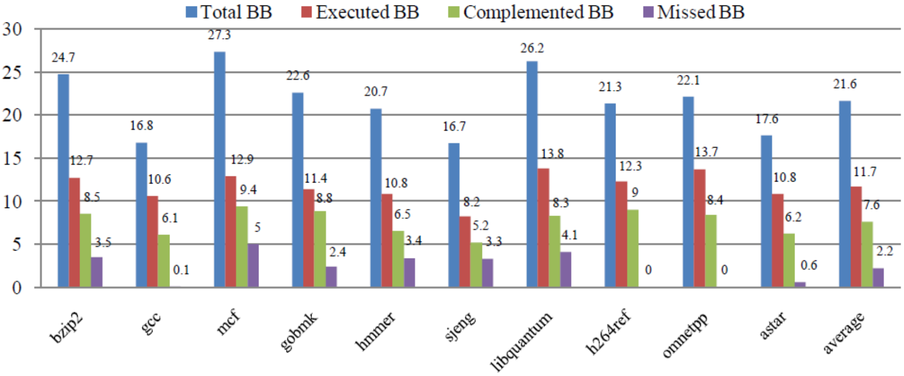


Fig. 6. Coverage Evaluation of HYBit

Table 1. Effectiveness Verification on General Software

Target Program	Attacks	Attacks Detected	Normal Test	Latent Bugs Detected	Source
BufferAttacker	11	11	41	41	Example Program
TxtEdit	13	13	20	11	Example Program
IrfanView 4.25	31	31	12	6	CVE-2010-1509
Foxit Reader 3.0 build 1120	22	22	23	19	CVE-2009-0836(0837)

Table 1 evaluates the effectiveness of HYBit by applying it to software vulnerability detection. This table provides the results of running IrfanView, Foxit Reader, BufferAttacker and TxtEdit on HYBit. These tested applications all popular used in computers. In the table, the “Attacks” and “Attack Detected” columns show the number of test cases which cause buffer overflow and whether they are detected. With HYBit, all of these malicious behaviors are discovered. The “Normal test” and “Latent Bugs Detected” show the number of normal test cases which do not bring out buffer overflow and if the weak spots will still be found. From the table, we can see that HYBit can discover the possible weak points and the latent software vulnerabilities in all parts of the target programs.

4 Conclusions and Future Work

This paper proposes HYBit, a novel hybrid framework which integrates dynamic and static taint analysis to track the flaws or vulnerabilities for binary programs. Dynamic analysis and static analysis techniques offer two complementary approaches for checking vulnerabilities. HYBit exploits the strengths of the two, and eliminates the drawbacks. In order to further improve the performance, a taint behavior filtration optimization mechanism is proposed. The results of the experiments on benchmarks show that the system is efficient and effective and covers most part of the target program. In the future, program slicing will be applied to optimize the framework. Symbolic execution may also be implemented to extend the utility of the framework.

Acknowledgement. This work was supported by the academic and technical leader recruiting Foundation of Anhui University, the National Natural Science Foundation of China (Grant No. 61003131), the National High Technology Research and Development Program(863Program) of China (Grant No. 2012AA010905), the National Basic Research Program (973 Program) of China (Grant No. 2012CB723401).

References

1. Tripp, O., Pistoia, M., et al.: TAJ: Effective Taint Analysis of Web Applications. In: Proceedings of the 2009 ACM SIGPLAN Conference on Programming Language Design and Implementation, pp. 87–97. ACM Press, New York (2009)
2. Csallner, C., Smaragdakis, Y., Xie, T.: Dsd-crasher: A hybrid analysis tool for bug finding. ACM Transactions on Software Engineering and Methodology 17(2), 1–37 (2008)
3. Zuliani, P., Platzer, A., Clarke, E.M.: Bayesian statistical model checking with application to simulink/stateflow verification. In: Proceedings of the 13th ACM International Conference on Hybrid Systems: Computation and Control, pp. 243–252. ACM Press, New York (2010)
4. Newsome, J., Song, D.: Dynamic taint analysis for automatic detection, analysis, and signature generation of exploits on commodity software. In: Proceedings of 2005 Network and Distributed System Security Symposium. Internet Society, Virginia (2005)
5. Clause, J., Li, W., Orso, A.: Dytan: a generic dynamic taint analysis framework. In: Proceedings of the 2007 International Symposium on Software Testing and Analysis, pp. 196–206. ACM Press, New York (2007)
6. Qin, F., Wang, C., Li, Z., et al.: Lift: A lowoverhead practical information flow tracking system for detecting security attacks. In: Proceedings of the 39th Annual IEEE/ACM International Symposium on Microarchitecture, pp. 135–148. IEEE Computer Society, Washington (2006)
7. Yang, Y., Guan, H., Zhu, E., et al.: CrossBit: A Multi-Sources and Multi-Targets DBT. In: The First International Conference on Cloud Computing, GRIDs, and Virtualization, pp. 41–47. IARIA (2010)
8. Guan, H., Zhu, E., Wang, H., et al.: SINOF: A dynamic-static combined framework for dynamic binary translation. Journal of Systems Architecture 58(8), 305–317 (2012)
9. Yin, H., Song, D., Egele, M., et al.: Panorama: capturing system-wide information flow for malware detection and analysis. In: Proceedings of the 14th ACM Conference on Computer and Communications Security, pp. 116–127. ACM Press, New York (2007)

The Application of the Pattern Recognition Algorithms in Security Assessment of Structural Health Monitoring for Bridges

Yilin Guo[*]

Research Institute of Highway, Bridge Technology Centre, Beijing, China

Abstract. Each year bridge collapse causes huge loss in China. The damage identification of bridges is a difficult problem. The Pattern recognition is an important method in security assessment of structural health monitoring. Taking a railway bridge as an example, the paper introduces the application of the pattern recognition algorithms in damage identification. It is concluded that preparation work involved infinite element analysis, feature extract, and sample training is important to improve the identification effect for the pattern recognition.

Keywords: pattern recognition, damage identification, structural health monitoring.

1 Introduction

The bridge damage has become a worldwide problem. Each year bridge collapse causes huge loss in China. So it is necessary to carry out the structural health monitoring for important and dangerous bridges. The process of implementing a damage detection and characterization strategy for engineering structures is referred to as Structural Health Monitoring (SHM). Here damage is defined as changes to the material or geometric properties of a structural system, including changes to the boundary conditions and system connectivity, which adversely affect the system's performance. The SHM process involves the observation of a system over time using periodically sampled dynamic response measurements from an array of sensors, the extraction of damage-sensitive features from these measurements, and the statistical analysis of these features to determine the current state of system health. For long term SHM, the output of this process is periodically updated information regarding the ability of the structure to perform its intended function in light of the inevitable aging and degradation resulting from operational environments. After extreme events, such as earthquakes or blast loading, SHM is used for rapid condition screening and aims to provide, in near real time, reliable information regarding the integrity of the structure [1].

Pattern recognition is an important method in security assessment of structural health monitoring for bridge. It is emerged in 1920s and has been developed to

[*] Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 240–247, 2013.

systematic disciplines. Pattern recognition algorithms generally aim to provide a reasonable answer for all possible inputs and to perform "most likely" matching of the inputs, taking into account their statistical variation. For bridge damage identification, if we classify the structure's response according to the damage feature, it is possible to use the pattern recognition to classify the response for an existing bridge. So pattern recognition algorithms have become an important method in damage identification of bridges.

2 Pattern Recognition in Damage Identification

The SHM problem can be addressed in the context of a statistical pattern recognition paradigm [2]. This paradigm can be broken down into four parts: (1) Operational Evaluation, (2) Data Acquisition and Cleansing, (3) Feature Extraction and Data Compression, and (4) Statistical Model Development for Feature Discrimination. When one attempts to apply this paradigm to data from real world structures, it quickly becomes apparent that the ability to cleanse, compress, normalize and fuse data to account for operational and environmental variability is a key implementation issue when addressing Parts 2-4 of this paradigm. These processes can be implemented through hardware or software and, in general, some combination of these two approaches will be used.

2.1 Operational Evaluation

Operational evaluation begins to set the limitations on what will be monitored and how the monitoring will be accomplished. This evaluation starts to tailor the damage identification process to features that are unique to the system being monitored and tries to take advantage of unique features of the damage that is to be detected.

2.2 Data Acquisition, Normalization and Cleansing

The data acquisition portion of the SHM process involves selecting the excitation methods, the sensor types, number and locations, and the data acquisition/storage/transmittal hardware. Again, this process will be application specific. Economic considerations will play a major role in making these decisions. The intervals at which data should be collected is another consideration that must be addressed.

Because data can be measured under varying conditions, the ability to normalize the data becomes very important to the damage identification process. As it applies to SHM, data normalization is the process of separating changes in sensor reading caused by damage from those caused by varying operational and environmental conditions. One of the most common procedures is to normalize the measured responses by the measured inputs. When environmental or operational variability is an issue, the need can arise to normalize the data in some temporal fashion to facilitate the comparison of data measured at similar times of an environmental or operational cycle. Sources of variability in the data acquisition process and with the system being monitored need to be identified and minimized to the extent possible. In general, not all sources of variability can be eliminated. Therefore, it is necessary to make the appropriate measurements such that these sources can be statistically quantified. Variability can arise from changing environmental and test conditions, changes in the data reduction process, and unit-to-unit inconsistencies.

Data cleansing is the process of selectively choosing data to pass on to or reject from the feature selection process. The data cleansing process is usually based on knowledge gained by individuals directly involved with the data acquisition. As an example, an inspection of the test setup may reveal that a sensor was loosely mounted and, hence, based on the judgment of the individuals performing the measurement, this set of data or the data from that particular sensor may be selectively deleted from the feature selection process. Signal processing techniques such as filtering and re-sampling can also be thought of as data cleansing procedures.

Finally, the data acquisition, normalization, and cleansing portion of SHM process should not be static. Insight gained from the feature selection process and the statistical model development process will provide information regarding changes that can improve the data acquisition process.

2.3 Feature Extraction and Data Compression

The area of the SHM process that receives the most attention in the technical literature is the identification of data features that allows one to distinguish between the undamaged and damaged structure. Inherent in this feature selection process is the condensation of the data. The best features for damage identification are application specific.

One of the most common feature extraction methods is based on correlating measured system response quantities, such a vibration amplitude or frequency, with the first-hand observations of the degrading system. Another method of developing features for damage identification is to apply engineered flaws, similar to ones expected in actual operating conditions, to systems and develop an initial understanding of the parameters that are sensitive to the expected damage. The flawed system can also be used to validate that the diagnostic measurements are sensitive enough to distinguish between features identified from the undamaged and damaged system. The use of analytical tools such as experimentally-validated finite element models can be a great asset in this process. In many cases the analytical tools are used to perform numerical experiments where the flaws are introduced through computer simulation. Damage accumulation testing, during which significant structural components of the system under study are degraded by subjecting them to realistic loading conditions, can also be used to identify appropriate features. This process may involve induced-damage testing, fatigue testing, corrosion growth, or temperature cycling to accumulate certain types of damage in an accelerated fashion. Insight into the appropriate features can be gained from several types of analytical and experimental studies as described above and is usually the result of information obtained from some combination of these studies.

The operational implementation and diagnostic measurement technologies needed to perform SHM produce more data than traditional uses of structural dynamics information. A condensation of the data is advantageous and necessary when comparisons of many feature sets obtained over the lifetime of the structure are envisioned. Also, because data will be acquired from a structure over an extended period of time and in an operational environment, robust data reduction techniques must be developed to retain feature sensitivity to the structural changes of interest in the presence of environmental and operational variability. To further aid in the extraction and recording of quality data needed to perform SHM, the statistical significance of the features should be characterized and used in the condensation process.

2.4 Statistical Model Development

The portion of the SHM process that has received the least attention in the technical literature is the development of statistical models for discrimination between features from the undamaged and damaged structures [3]. Statistical model development is concerned with the implementation of the algorithms that operate on the extracted features to quantify the damage state of the structure. The algorithms used in statistical model development usually fall into three categories [4]. When data are available from both the undamaged and damaged structure, the statistical pattern recognition algorithms fall into the general classification referred to as supervised learning. Group classification and regression analysis are categories of supervised learning algorithms. Unsupervised learning refers to algorithms that are applied to data not containing examples from the damaged structure. Outlier or novelty detection is the primary class of algorithms applied in unsupervised learning applications. All of the algorithms analyze statistical distributions of the measured or derived features to enhance the damage identification process [5].

3 Application in Railroad Bridges

Because the damage identification is closely related to the external load which is not able to measure exactly, we should analyze the outer load and then select the proper method when we make use of pattern recognition to identify the bridge damage [6]. The support vector machine is the pattern classifier. When it is used to identify the railway bridge damage, the action of the train load is analyzed first then we confirm the proper index and set up the sample library. At last we select the proper support vector classifier to train the sample.

3.1 Research Ideas

Because the live load takes a large portion, the dynamic response is obviously. According to operation of railroad bridges, the travel time of train is divided into some domains. The damage state of every component is invariable in a short period, and we can analyze the important component's damage according to the time sequence of each domain. If we define the structure as the intact condition or damage condition, the variable of intact condition is positive and the variable of damage condition is negative. Because of the comparability of structural damage feature, the bridge is divided into several sections. According to the arrival time of the train, we analyze the damage by sections. If one section's damage is detected, the early warning is started and the analysis is stop. The process is shown in Fig.1. In the light of the flow, the pattern recognition can be divided into three steps:

(1) Set up the damage index;
(2) Form the sample library;
(3) Select the categorizer.

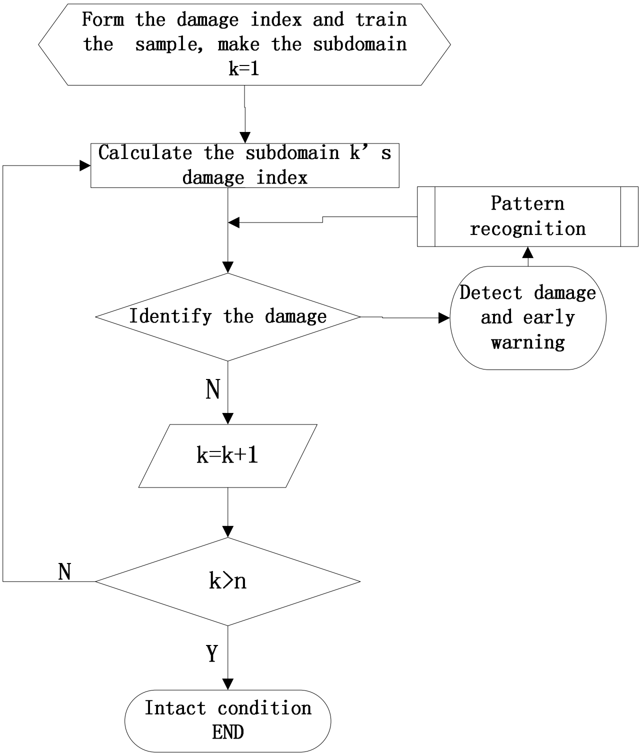

Form the damage index and train the sample, make the subdomain k=1
Calculate the subdomain k' s damage index
Pattern recognition
Identify the damage
Detect damage and early warning
N
k=k+1
N
k>n
Y
Intact condition END

Fig. 1. The process of the damage identification

3.2 The Example Analysis

Taking the continuous beam bridges as an example, we study the damage identification when the train travels. The span arrangement is 100m+192m+100m. The section area is 38m^2. The moment inertia is 502m^4. The modulus of elasticity is 34.5GPa. The density is 2500kg/m^3. The bridge is divided into 32 elements and laid 13 acceleration sensors whose sampling frequency is 90Hz. It is shown in Figure2. The length of the train is 400m and the travel velocity is 400m/h. The travel time on the bridge is 14.256s. So take this period's data as the sample.

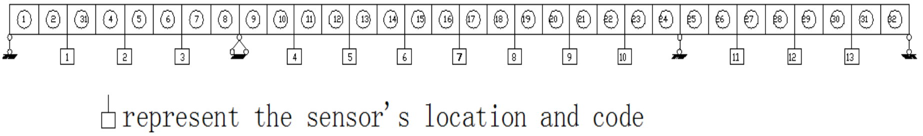


Fig. 2. The bridge's layout of element partition and sensor arrangement

Analyze the sensitivity of structural damage and confirm the damage location first. Utilizing the finite element analysis, the span centre and abutment are the easy location to damage. We can assume that the damage is invariable for a period. Travel time is divided into some domains. Select two domains for damage identification. Assuming that the time when the head of the train come into the bridge is 0s, the two domains are 0~1.8s and 2.367~5.256s.

In the domain of 0~1.8s, the train travels in the first span. Analyzed the damage in the first span, the sample library is divided into two kinds: the one is positive because the index is in intact conditions; the other is negative because the index is in damage conditions. There are 1800 samples in positive kind and 3600 samples in negative kind. In negative samples, there are three kinds of damage conditions. The 5% damages bring in the 3rd~4th elements are 1200. It is same to the 5th~6th elements and 7th~8th elements.

In the 2.367~5.256s, the train travels in the second span. Analyzed the damage in the second span, the sample library is also divided into two kinds: the one is positive because the index is in intact conditions; the other is negative because the index is in damage conditions. There are 1800 samples in positive kind and 7200 samples in negative kind. In negative samples, there are six kinds of damage conditions. The 5% damages bring in the 9th~10th elements are 1200. It is same to the 13th~14th elements, 15th~16th elements, 17th~18th elements, 19th~20th elements, and 23th~24th elements.

The support vector machine by the method of least squares is the pattern classifier, and kernel function is:

$$K(x',x) = \sum^{\mathbf{n}}_{\mathbf{1,j}} \mathbf{=18(x',x)} \tag{1}$$

n—sampling points

For the first domain, the sampling points are 150. For the second domain, they are 200. The sample is trained by the classifier and then identified the damage. Taking advantage of the finite element analysis, we set up 9 kinds of damage pattern in Table1.

Table 1. Damage pattern by the finite element analysis

The second domain	1	2	3	4	5	6	7	8	9
Damage Location	Intact	5, 6	5, 6	4	4	15, 16	15, 16	23	23
Damage Degree	Intact	6%	4%	8%	6%	6%	3%	8%	4%

The fourth, fifth, and sixth elements belong to the first domain whose identified code is 2, 3, 4, and 5 in Table1. In the same way, the second domain's identified code is 6, 7, 8, and 9 in Table1. For the same structure, the damage index is different from variable load. To check up the identification level for classifier, we select 6000 indexes to test for each structure. The identification result is shown in Table 2 and Table 3.

Table 2. The identification result of the first domain

The first domain	
Identified Code	Identified precision
1	97.7%
2	100%
3	87%
4	100%
5	74.7%

Table 3. The identification result of the second domain

The second domain	
Identified Code	Identified precision
1	95.3%
6	100%
7	43.2%
8	99.4%
9	65.4%

As shown in table, code 1, 2, and 4 are identified well for the first domain. Code 1, 6, and 8 are identified well for the second domain. The reason is that the collection data is similar to the sample library. Code 3, 5, 7, and 8 are identified badly. The reason is the collection data is between the intact and damage sample, so it is difficult to classify correctly.

4 Conclusion

By means of the example for pattern recognition, we can draw the flow conclusions in damage identification for bridges:

(1) Before the damage identification, the pattern recognition needs to do some preparation work involved structural infinite element analysis, feature extraction and sample training. It is the preparation work that improves the identified precision and reduces the identified difficulty.

(2) Before the pattern recognition, it is necessary to calculate the damage index by infinite element method and form the sample library. So the infinite element analysis and patter recognition are two independent processes.

(3) Because the pattern recognition processing is complicated, the effect of the identification is determined by calculation scale, sample selection and support vector machine classifier.

References

1. Teng, L., Zeng, C.: Application of Mode Recognition Technique in the Evaluation of Bridge State and Safety Inspection. China Railway Science 4, 47–51 (2005)
2. Deshan, S., Chunyu, F., Qiao, L.: Statistical Pattern Recognition of Structural Damage Detection of Railway Bridges. Bridge Constraction 1, 18–22 (2011)
3. Zhou, B., Yang, X., Wang, L.: An Identification Method for Bridge Based on Pattern Recognition. Computer & Digital Engineering 4, 173–176 (2010)
4. Xiang, Y., Zhou, C., Li, Y., Li, C.: Health monitoring and altering system on-line of bridge structures: Signal analysis and extraction method for damage identification. Journal of Transport Science and Engineering 2, 33–39 (2009)
5. Fu, Y., Zhu, F., Zan, X.: Missing Data Imputation in Bridge Health Monitoring System Based on the Support Vector Machine. Chinese Journal of Sensors and Actuators 12, 1706–1710 (2012)
6. Zhang, Q.: Damage Feature Extraction and Novelty Detection for Bridge Health Monitoring. Journal of Tongji University 3, 258–262 (2003)

Experimentation of Data Mining Technique for System's Security: A Comparative Study

Ahmed Chaouki Lokbani[1], Ahmed Lehireche[2], and Reda Mohamed Hamou[1]

[1] Dr Moulay Tahar University of SAÏDA, Computer Science Department, Algeria
[2] Djillali Liabbes University of Sidi Bel Abbes, Science Department, Algeria
{Ahchlokbani,hamoureda}@yahoo.fr, elhir@univ_sba.dz

Abstract. Given the increasing number of users of computer systems and networks, it is difficult to know the profile of the latter and therefore the intrusion has become a highly prized of community of network security. In this paper to address the issues mentioned above, we used the data mining techniques namely association rules, decision trees and Bayesian networks. The results obtained on the KDD'99 benchmark has been validated by several evaluation measures, and are promising and provide access to other techniques and hybridization to improve the security and confidentiality in the field.

Keywords: Association rules, Intrusion Detection, Data Mining, Knowledge Discovery, KDD 99.

1 Introduction and Problematic

Currently computers and the Internet in particular play an increasingly important role in our society. Networks and computer systems have become today an indispensable tool for the proper functioning and development of most companies. Thus, computer systems and networks are deployed in various fields such as banking, medicine or the military. The increasing interconnection of these various systems and networks has made them accessible to a diverse population of users that continues to increase. Since these networks have emerged as potential targets of attacks, the security has become an essential problem and an unavoidable issue.

Many mechanisms have been developed to ensure the security of computer systems and particularly to prevent intrusions, unfortunately, these mechanisms have limitations. In fact, computer systems have vulnerabilities that allow attackers to bypass prevention mechanisms. For this, a second line of defense is necessary: intrusion detection. For each system, a security policy must be defined to guarantee the security properties that must be made by the latter. This policy is expressed by rules, setting three distinct objectives:

The confidentiality-that is to say, the non-occurrence of unauthorized disclosure of information;

- Integrity that is to say, the non-occurrence of improper alterations of information;
- The availability is to say being ready to use.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 248–257, 2013.

In this study, we define intrusion, a violation of one of these three objectives. Several approaches have been developed to ensure that the security policy defined for a computer system is respected. Artificial intelligence has seen many methods such as data mining and its various techniques that will be used for intrusion detection.

In this article we are experiencing some data mining techniques namely association rules, Bayesian networks and decision trees in the area of intrusion detection. The intrusion detection system designed and will be tested on a benchmark called KDD'99 which is a structured database that will be detailed later.

2 State of the Art

The concept of intrusion detection system was introduced in 1980 by James Anderson. But the subject has not been very successful. It was not until the publication of an intrusion detection model by Denning in 1987 to mark, really the start of the field.

The research in the field is then developed; the number of prototypes has increased enormously. A lot of money has been invested in this type of research in order to increase the safety of its machines. Intrusion detection has become an industry mature and proven technology: almost all the simple problems have been solved, and no major progress has been made in this area in recent years, software vendor's focus more improves the existing detection techniques.

Some tracks remain relatively unexplored:

- Mechanisms to respond to attacks,
- The architecture for intrusion detection systems distributed
- Standards for interoperability between different systems, intrusion detection,
- The search for new paradigms to perform intrusion detection.

One approach to computer security is to create a completely secure system is prevention. But it is rarely possible to make a completely watertight for several reasons.

- Most computer systems have security flaws that make them vulnerable to intrusions. Find and repair all is not possible for technical and economic reasons. (Balasubramaniyan, 1998)
- Existing systems with known vulnerabilities are not easily replaced by safer systems, mainly because they have interesting features that do not have the systems safer, or because they cannot be replaced for economic reasons.
- Deploy systems without faults is very hard or impossible because of the faults are unknown or unavoidable.
- Even the most secure systems are vulnerable to abuse by legitimate users who take advantage of their privileges, or suffer from neglect of safety rules.

In response to these difficulties to develop secure systems, a new model of security management systems has emerged (Figure 1).

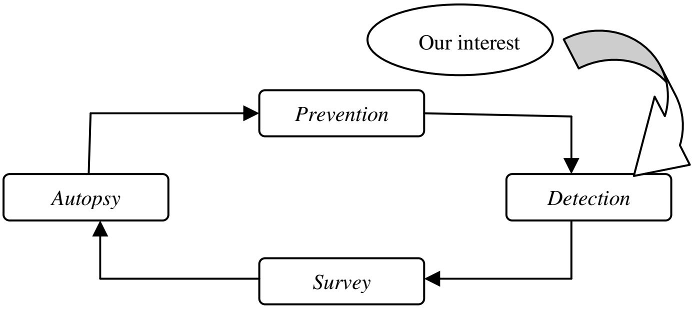


Fig. 1. A model of security management an IT system

In this more realistic approach, prevention is one of the four parts of the security management. The detection part is looking for the exploitation of new loopholes. Part investigation tries to determine what happened, based on information provided by the detection part. Part autopsy is to look how to prevent similar intrusions in the future.

In the past almost all the attention of researchers has focused on the prevention part. Detection is now much taken into account, but the other two parties have not yet received the attention they deserve.

As previous work related to the field of intrusion detection and operational safety we find in the literature two major approaches to intrusion detection: the approach by signature and behavioral approach.

In this paper we are particularly interested in the detection part we try to solve by data mining techniques already revealed.

3 Proposed Approach

3.1 Networks Bayesian

Bayesian Networks are graphical models that represent the relationships probabilized between a set of variables they have huge advantages over other techniques. These networks can intuitively represent an area of knowledge; many experiments show that it is often easier to formalize knowledge as a causal graph that in the form a system based on rules. In addition, they can handle all the data incomplete and can learn the causal relationship can help us make decisions.

A Bayesian network can be formally defined by:

- A directed acyclic graph G, G = G (V, E) where V is the set of nodes of G, and E the set of edges of G.
- A finite probability space (Ω, p).
- A set of random variables associated with the nodes of the graph defined on [Ω, p] such t $p(v_1, v_é, \ldots, v_n) = \prod_{i=1..n} p(v_i\,|C(v_i)$ with C (Vi) is the set of parents of Vi in the graph.

A Bayesian network is therefore a causal graph, which was associated with a probabilistic representation underlying. This representation allows rendering quantitative reasoning about causality that can be done inside the graph.

The use of Bayesian networks is essential to compute the conditional probabilities of events connected to each other by relations of cause and effect. This use is called inference. If the graph is composed of n nodes, denoted X={X1, X2,..., Xn}. The general problem of inference is to compute $p\ (X \mid Y)$, where $Y \subset X$ and $Xi \notin Y$.

3.2 Decision Trees

Decision trees (Quinlan, 1986) (Quinlan, 1993) represent one of the most widely known and used in classification. Their success is partly due to their ability to deal with complex problems of classification. In fact, they offer an easily understood and interpreted, and an ability to produce logical rules of classification.

A decision tree is composed of:

Decision nodes each containing a test on an attribute.

Branches generally corresponding to one of the possible values of the selected attribute; Sheets comprising the objects that belong to the same class.

The use of decision trees in classification problems is done in two steps:

The construction of a decision tree from a training database.

Classification or inference of classifying a new instance from the decision tree built in the previous step.

In our study, the attribute is the number 37 (A37) KDD'99's benchmark that will be detailed later.

3.3 Rules of Association

To formalize the principles of association rules, we assume that the data D (a set of objects or transactions) to explore are binary, ie each transaction that can be described by a finite set of attributes I = {i1, . . . , im}, also called items. Each transaction t will be a subset of I. In addition, it assigns to each transaction identifier (TID for "Transaction IDentifier"): D = {t1, . . . , tn}. Table 1 below illustrates this type of data, for a total of 10 transactions {{t1,... ,t10}described by seven items {i1, . . . , i7}. We will use throughout this section illustration of the example described by Table 1 to understand the formalism of association rules.

For a set X of items (called itemsets typically) of I, we say that a transaction T contains X if and only if $X \subseteq t$. For example, the itemset {i3, i4} is contained in the transaction t1.

Table 1. Sample illustration

t1	i3 i4 i5
t2	i1 i3 i5 i6
t3	i1 i3 i4 i7
t4	i2 i4 i6
t5	i1 i2 i4 i5 i7
t6	i1 i2 i3 i4 i5 i7
t7	i2 i4 i5 i6
t8	i1 i4 i6
t9	i3 i4 i6
t10	i2 i3 i5 i6.

We call, support of an itemset X is the ratio between the number of transactions containing X and the total number of transactions in D. This proportion is denoted Sup.

$$\mathrm{sup}(X)= \frac{|\{t\in D:X\sqsubseteq t\}|}{|D|} \tag{1}$$

In other words, the support of an itemset corresponds to the frequency of appearance thereof in the data. In the data listed above, we have for example:

$$\mathrm{sup}(i_1)= \frac{|\{t2,t3,t5,t6,t8\}|}{|10|}=\frac{5}{10}=50\%.$$

$$\mathrm{sup}(i_3,i_4)= \frac{|\{t2,t3,t6,t9\}|}{|10|}=\frac{54}{10}=40\%.$$

By the definition of an itemset, we have the following property:

Property 1. (anti monotonicity) Given two itemsets X and Y, we have:

$$X\subseteq Y \Rightarrow \mathrm{sup}\ (x) \geq \mathrm{sup}\ (y) \tag{2}$$

Property 2. Less intuitively, also note that:

$$\mathrm{sup}\ (x \cup y) \leq \min\ (\mathrm{sup}(x)\,,\mathrm{sup}\ (y)) \leq \max\ (\mathrm{sup}(x)\,,\mathrm{sup}\ (y)) \leq \mathrm{sup}\ (x \sqcap y) \tag{3}$$

4 Experimentation and Results

4.1 Corpus Used

Data base KDD'99' s are oriented intrusion detection, they represent lines of TCP / IP dump where each line is a connection characterized by 41 attributes (detailed in Table 2), as the duration of the connection, the protocol type, etc.. Taking into account the values of its attributes, each connection in KDD'99' s is considered a normal connection or an attack.

Table 2. List of attributes based connections KDD'99' s

Basics Attributes
A1 connection time (number of seconds) A2 protocol type, eg. tcp, udp, etc.. A3 Network Service (destination) ex. http, telnet A4 connection status (normal or error) A5 nb data (in bytes) from the source to the destination A6 nb data (in bytes) from the destination to the source A7 1 if connection is from / to the same host / port 0 otherwise A8 nb fraguements "erroneous" A9 # of urgent packets …

In our study we chose a method based on the selection of attributes that can select a subset of relevant attributes from the original set of attributes according to a performance criterion. These methods allowing characterizing data more quickly and therefore cost in computation time will be minimized. The selection of attributes does not change the original data representation: selected attributes keep their semantic departure and can then be more easily interpreted by the user. In our study, the choice fell on the attribute A37.

According Dash [1997], a procedure for selection of attributes is generally composed of four steps.

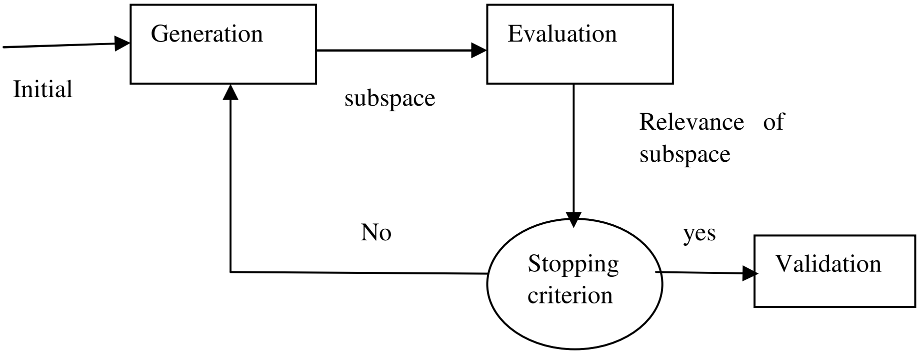


Fig. 2. The different stages of a selection procedure of attributes

4.2 Contingency Table and Evaluation Measure

Contingency table (or table of co-occurrence) is a tool often used when it is desired to study the relationship between two variables that take discrete values (or categories). In our case, the variables are in the columns, actual (also known as "Gold Standard") and in the lines, the result of the filter. The sum of each column gives the actual number of elements in each class and of each line gives the number of elements seen by the classifier in each class. The following table shows the shape of the contingency table.

Table 3. Form of the contingency table

	True « Abnormal »	True « Normal »
Classement « Abnormal »	VP	FP
Classement « Normal »	FN	VN

Where

VN: True negatives: Normal seen by the filter as Normal or Normal correctly classified.

FN: False negative: Abnormal seen as Normal or Abnormal not correctly classified.

VP: True positives: Abnormal seen as Abnormal or Abnormal correctly classified.

FP: False positive: Normal seen as Abnormal or Normal not correctly classified.

Error Rate by Class

(False positives and false negatives): it is the fraction of the number of objects in a category erroneously classified in another class.

$$FPR = \frac{FP}{VN+FP} \qquad FNR = \frac{FN}{VP+FN} \tag{4}$$

Rate of Good Ranking

(True positives and true negatives or sensitivity and specificity)

$$VPR = \frac{VP}{VP+FN} = 1 - FNR \qquad VNR = \frac{VN}{VN+FP} = 1 - FPR \tag{5}$$

Precision and Recall

The precision indicates the proportion of spam messages among detected as spam, while the recall is the ratio between the number of detected spam rightly and the total number of spams.

$$Precision = \frac{VP}{VP+FP} \qquad Recall = \frac{VP}{VP+FN} = VPR \tag{6}$$

Accuracy

This is the total error rate, the two classes combined.

$$Accuracy = \frac{(VP+VN)}{(VP+FN+VN+FP)} \tag{7}$$

4.3 Results

Table 4 shows the results of different classifiers used in our study. The results are excellent and give an ascendancy over decision trees. Bayesian networks are powerful tools for reasoning and decision under uncertainty as decision trees are most commonly used in classification problems.

In terms of time complexity, Bayesian networks have polynomial complexity and are less expensive than the decision trees that have a non-polynomial complexity.

Association rules are easy to interpret and use. The model based on association rules is simple unequivocal because the calculations are elementary (it only calculates the frequency of occurrence) but the method is expensive in computation time. Clustering and minimum support method can reduce the calculation time but can inadvertently remove important rules. Regarding the quality of the rules, the method can produce rules trivial or useless. Trivial rules are rules obvious, therefore, do not provide information. Unnecessary rules are rules difficult to interpret.

Figures 4 and 5 give an overview on the ancestry of decision trees relative to Bayesian networks and association rules.

Table 4. Results of learning by naïve bayes with data cleaning

Approach	# Con.	Precision	recall	F-measure	Accuracy	Kappa statistic	Confusion matrix	
							Anormal	Normal
Bayesian networks	25192	89.60 %	89.60 %	89.60 %	89.59 %	79.06 %	12272	1177
							1445	10298
Decision Trees	25192	99.60 %	99.60 %	99.60 %	99.55 %	99.11 %	13389	60
							51	11692
Association rules	25192	96.30 %	96.20 %	96.20 %	96.20 %	99.11 %	12652	797
							159	11584

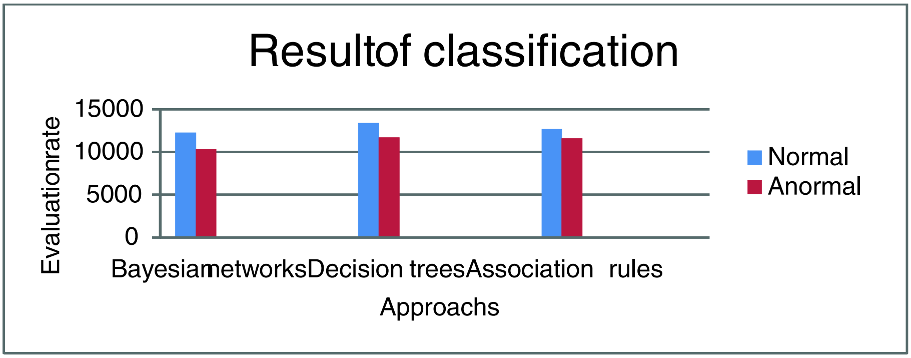


Fig. 3. Synthesis of the results of classification for the different approaches

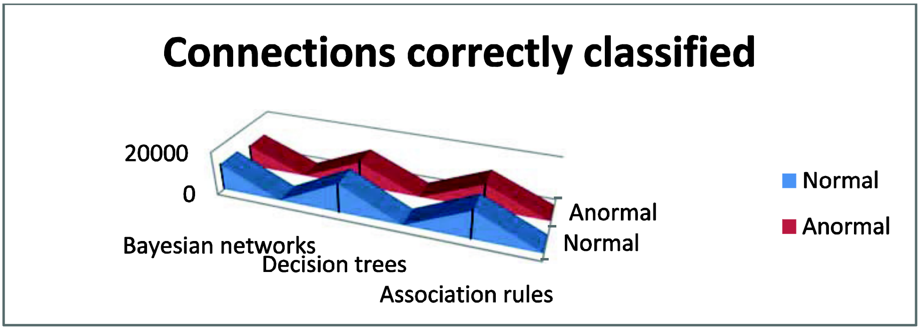
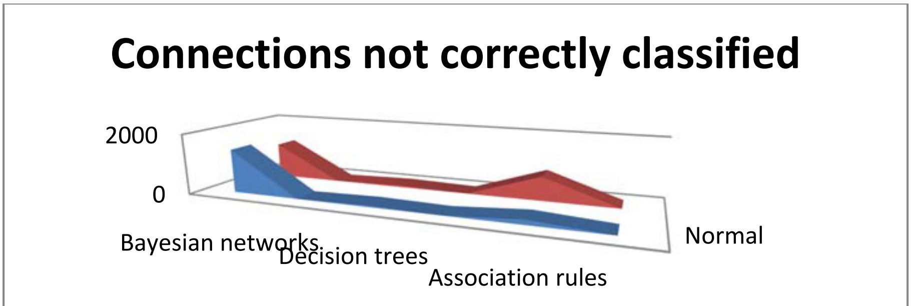


Fig. 4. Connections properly classified for the different approaches



	Bayesian networks		Decision trees		Association rules	
🟦 Normal	1445		51		159	
🟥 Anormal	1177		60		797	

Fig. 5. Connections not properly classified for the different approaches

5 Conclusion and Perspectives

We experimented three approaches supervised for intrusion detection for known benchmark namely KDD'99. Approaches tested gave good results for each with advantages and disadvantages. The tests were carried out on 10% of existing connections of KDD'99. 10% of the test database thus found has been a learning base. Attribute selection was made on the attribute A37 of KDD'99.

As future work, we plan to develop a meta-classifier based on supervised approaches give good results to improve the intrusion detection system, exploiting the complementarily of supervised namely decision trees, networks Bayesian and association rules and explore other data mining techniques.

References

1. Denning, D.E., Neumann, P.G.: Requirements and model for IDES-A real-time intrusion detection system. Comput. Sci. Lab, SRI International, Menlo Park, CA, Tech. Rep. (1985)
2. Balasubramaniyan, J.S., Garcia-Fernandez, J.O., Isacoff, D., Spafford, E., Zamboni, D., Zamboni, D.: An architecture for intrusion detection using autonomous agents. In: 14th Annual Computer Security Applications Conference (ACSAC 1998), December 7-11. IEEE Computer Society, Scottsdale (1998) ISBN 0-8186-8789-4
3. Debar, H., Dacier, M., Wespi, A.: Towards a taxonomy of intrusiondetection systems. Computer Networks 31(8), 805–822 (1999)
4. Anderson, D., Lunt, T.F., Javitz, H., Tamaru, A., Valdes, A.: Detecting unusual program behavior using the statistical component of the nextgeneration intrusion detection system (NIDES). Technical Report SRI-CSL-95-06, Computer Science Laboratory, SRI International, Menlo Park, CA, USA (May 1995)
5. Axelsson, S.: Intrusion detection systems: a survey and taxonomy, rapport de recherche (2000)
6. Cooper, G.F.: Computational complexity of probabilistic inference using bayesian belief networks. Artificial Intelligence 42, 393–405 (1990)
7. Dougherty, J., Kohavi, R., Sahami, M.: Forrests of fuzzy decision trees. In: Proceedings of ICML 1995: Supervised and Unsupervised Discretization of Continuous Features, pp. 194–202 (1995)
8. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. Hardcover (2000)
9. Esposito, F., Malerba, D., Semeraro, G.: A comparative analysis of methods for pruning decision trees. IEEE Pattern Analysis and Machine Intelligence 19, 476–491 (1997)
10. Fayyad, U.M., Irani, K.B.: On the handling of continuous-valued attributes in decision tree generation. Machine Learning 8, 87–102 (1992)
11. Hamou, R.M., Lehireche, A., Lokbani, A.C., Rahmani, M.: Representation of textual documents by the approach wordnet and n-grams for the unsupervised classifcation (clustering) with 2D cellular automata:a comparative study. Journal of Computer and Information Science 3(3), 240–255 (2010) ISSN 1913-8989 E-ISSN 1913-8997
12. Hamou, R.M., Abdelmalek, A., Mohamed, R.: Visualization and clustering by 3D cellular automata: Application to unstructured data. International Journal Of Data Mining And Emerging Technologies 2(1) (2012) Print ISSN : 2249-3212. Online ISSN : 2249-3220

13. Hamou, R.M., Amine, A., Rahmani, M.: A new biomimetic approach based on social spiders for clustering of text. In: Lee, R. (ed.) Software Engineering Research, Management and Appl. 2012. SCI, vol. 430, pp. 17–30. Springer, Heidelberg (2012)
14. Friedman, N., Goldszmidt, M.: Building classifiers using bayesian networks. In: Proceedings of the American Association for Artificial Intelligence Conference, AAAI 1996 (1996)
15. Furnkranz, J.: Pruning algorithms for rule learning. Machine Learning 27, 139–171 (1997)
16. Hamou, R.M., Lehireche, A., Lokbani, A.C., Rahmani, M.: Text Clustering by 2D Cellular Automata Based on the N-Grams. In: 1st International Symposiums on Cryptography, Network Security, Data Mining and Knowledge Discovery, E-Commerce and its Applications, October 22-24. IEEE, Qinhuangdao (2010)
17. Ilgun, K., Kemmerer, R.A., Porras, P.A.: Probability propagation. IEEE Transactions on Software Engineering 21(3), 181–199 (1995)

Brownian Snake Measure-Valued Markov Decision Process

Zhenzhen Wang[1] and Hancheng Xing[2]

[1] School of Information Technology,
Jinling Institute of Technology, Nanjing, 211169, China
[2] School of Computer Science & Engineering,
Southeast University, Nanjing 210096, China
{wangzhenzhen,xhc}@seu.edu.cn

Abstract. This paper presents a model called Brownian snake measure-valued Markov decision process (BSMMDP) that can simulate an important characteristic of human thought, that is, when people think problems, sometimes they can suddenly connect events that are remote in space-time so as to solve problems. We also discuss how to find an (approximate) optimal policy within this framework. If Artificial Intelligence can simulate human thought, then maybe it is beneficial for its progress. BSMMDP is just following this idea, and trying to describe the talent of human mind.

Keywords: Brownian snake measure-valued Markov decision process, Markov decision process, human mind, Brownian snake.

1 Introduction

Markov decision process (MDP) model, that integrates Markov stochastic process theory with reinforcement learning method, has been paid more and more attention in the theoretic and applied areas of Artificial Intelligence, and with one understanding MDPs more deeply one wonders to use it to simulate the human mind, so that it's possible for AI to be really a kind of "human-like" intelligence.

Because the environment is generally unobservable or partially observable, the Markov decision process carried out in the totally observable environment is naturally replaced by partially observed Markov decision process (POMDP). Let us carefully investigate the exact solutions of POMDP (the first exact solution was proposed by E.J. Sondik in 1971.): its essence is to substitute belief state space for true state space, and substitute the transition probability over the belief state space for the transition probability over the true state space, and so under invariable action space, if the reward function (for instance) on the belief state space is defined as the expectation of true states, then solving a POMDP on a physical state space can be reduced to solving a MDP on the corresponding belief state space. It indeed is a good thing because an agent does not know which real state he is in and all he knows is just a belief state (See [1], pp.481-483).

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 258–266, 2013.

The agent's belief state in POMDP model is a probability distribution that reflects a whole grasp for the agent of the real space. The agent just depends on this grasp to help itself to think problem. Moreover it relies on prior knowledge (such as transition probability and reward function in previous MDP) and partial observation to transform its belief state, and thereby finds an optimal policy to problem. In this way it will become a real "intelligent person". Concerning the human thought, under fully (or partially) unknown circumstance people think problems and find the way out relying on a vague grasp of problem. Accompanied by deeply understanding problem, this grasp, more exactly speaking, --a "measure" to different parts of the problem will become more and more clear, and finally a complete solution to the problem will be found. Now the belief state of agent in POMDPs just simulates such phenomenon of human thought. So POMDP model gives us an important illumination that simulating human thought not only should be an important task in AI, but also usually be a key point for acquiring success in AI.

Another important phenomenon of human thought is that often people are able to suddenly link things that are remote in space-time, that is, the associative function plays an important role in human mind. People usually think problem in a logical manner (that is due to the long-term learning and experiences of human beings), but when having no idea about problem, the activities of human thought always move randomly, i.e., the thought "particle" moves as something as Brownian movement and shows quanta's characteristic, i.e., its locations and its proceeding directions, which are hardly controlled by rationality, go out of their way to seek unexpected way out in mind space: and that is the phenomenon we have described above: things distant from another in time or space will be linked together in order to search the route for solving problems. This paper presents a model, called Brownian snake measure-valued Markov decision process (BSMMDP) that tries to simulate this phenomenon, and a (approximate) meta-algorithm for finding optimal policy within this framework is also provided.

2 Brownian Snake Measure-Valued Markov Decision Processes

2.1 Reflective Brownian Movement on Semi-straight Line

Suppose $\overline{w}$ is a trajectory of reflective Brownian movement on the semi-straight line, and, $\overline{w}(0)=0$. We write $n(d\overline{w})$ as the Itô measure of the Brownian movement on straight line, and it is standardized as $n(\sup\limits_{s\geq 0}\overline{w}(s)>\varepsilon)=\dfrac{1}{2\varepsilon}$, $\varepsilon>0$ [2]. Next we introduce the conception of local time. A process $\left\{L_t^a(\overline{w}), t\geq 0, a\geq 0\right\}$ is called local time process of a reflective Brownian movement $\overline{w}(t)$ if it satisfies the following qualities: (i) $\forall a\geq 0, t\rightarrow L_t^a(\overline{w})$ is non-descending, and it has an increment only at $\overline{w}(t)=a$; (ii) for an arbitrary measurable function $\Psi: R_+\rightarrow R_+$, $s\geq 0$: $\int_0^s \Psi(\overline{w}(u))du=\int_0^\infty \Psi(a)\bullet L_s^a(\overline{w})da$. If we do not seek technical details (though this is required by mathematics for strictness), then intuitively, $L_t^a(\overline{w})$ could be considered as time cumulants of the trajectory $\overline{w}(\bullet)$ taking value a from time 0 to time t; that is, as if the Brownian movement trajectory $\overline{w}(\bullet)$ puts a "clock" at point a on the axis of ordinates; this clock would "walk" the moment the trajectory takes the value a, and stop when it takes other values. $L_t^a(\overline{w})$ is just the total time cumulants of the clock of the Brownian movement taking value a in the time interval [0, t]. We write $L_\infty^a(\overline{w})$ to denote the total time cumulants of Brownian movement taking value a in the whole trajectory.

$L_\infty^a(\bullet)$ is a random variable because $L_\infty^a(\overline{w})$ depends on $\overline{w}$.

The following figure 1 shows trajectory $\overline{w}(\bullet)$ of certain Brownian movement and several corresponding conceptions.

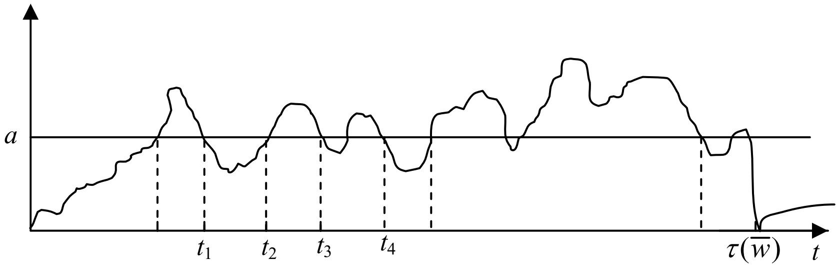


Fig. 1. A trajectory of some Brownian movement

At the points $t_1,t_2,t_3,\cdots$, $\overline{w}(\bullet)$ takes value a, and then the clock $L_\bullet^a(\overline{w})$ that is put by the Brownian movement at value a walks at the time $t_1,t_2,\cdots$. $L_\infty^a(\overline{w})$ is the total cumulants of these walkings. If $\tau(\overline{w})$ is used to denote the time when the trajectory $\overline{w}$ first takes value 0 after leaving from time 0, then there is the relation ($n(d\overline{w})-a.s$): $L_\infty^a(\overline{w})=L_{\tau(\overline{w})}^a(\overline{w})$. (See [2])

2.2 Markov Decision Processes

Given a Markov decision process M = (S, A, T, R), here S is state space and A is action space. They generally are finite spaces, that is S = (s^1, s^2, • • •, s^n), in which s^i is a state, A = (a^1, a^2, • • •,a^m), in which a^j is an action. This paper considers only the case in which S and A are finite, though in theory they may also be infinite (countable or even uncountable). T is a transition probability, for example, T(s, a, s') denotes the probability that action a, when executed in state s, transfers the system to state s'. It is convenient to suppose that for $\forall s \in S$ and $\forall a \in A$, $a \in A(s)$, i.e., a belongs to the set $A(s)$ of actions applicable in state s. It is easy for this supposition to hold only if when $a \notin A(s)$, let $T(s, a, s) = 1$. For the transition probability, we require: $\forall s \in S$,

$\forall a \in A$, $\sum_{s' \in S} T(s,a,s') = 1$. R is a reward function, and $R(s)$ denotes the reward attained by the agent in state s; also suppose for $\forall s, 0 \le R(s) \le \text{some constant}$.

We will use the terms MDP and bottom process interchangeable. Both of them refer to the Markov decision process M = (S, A, T, R). Moreover, we write $\xi_a(t),\ t = 0,1,2,\cdots$ as a random variable that obeys the transition probability $T_a(s,s') \equiv T(s,a,s')$, $\forall s, s' \in S$, and use $\xi(t)$ to represent an arbitrary random variable that obeys some transition probability. In other words, the agent chooses an action similar to Brownian movement at one step, and then varies according to ξ_a of this action (such as action a).

2.3 Brownian Snake

Suppose $\bar{w}$ is an arbitrary trajectory of Brownian motion. Fix an initial action a and initial state z. For $\forall u \in [0, \tau(\bar{w})]$, we define a Markov decision process Γ_u, which satisfies the following qualities:

1^0. Except that the first step takes the action a in the initial state z, Γ_u's state space, action space, transition probability and reward function are all the same as bottom process. Exactly speaking, the probability rule of Γ_u is the same as $\xi(t)$ ($t = 0,1,2,\cdots$) of bottom process, except $\xi(0) = z$ at time 0, that is, $\xi(t)$ first chooses action a in state z, i.e., $\xi(1) = \xi_a(1)$, next the same as the general random variable $\xi(t)$ $t = 2,3,\cdots$.

2^0. Γ_u terminates at time $[\bar{w}(u)]$, that is, its probability rule is the same as such $\xi(t)$ ($t = 2,\cdots,[\bar{w}(u)]$) whose first action is a when starting from state z, where $[x]$ denotes the greatest integer that is smaller than x.

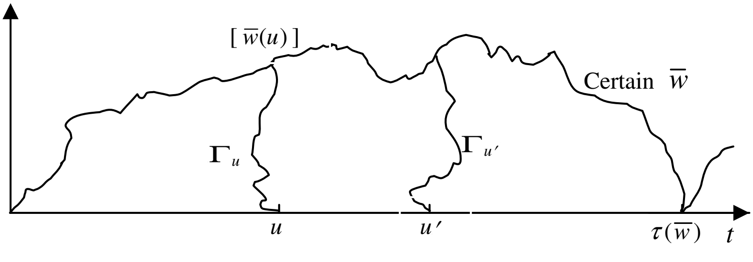


Fig. 2. A picture of $(\Gamma_u, u \in [0, \tau(\bar{w})])$

We call $(\Gamma_u, u \in [0, \tau(\bar{w})])$ Brownian snake.

Intuitively, if each motion trajectory Γ_u is imagined as thinking activities of the agent starting from state z, then because the domain of the definition of $\bar{w}$ is treated as observation time, and at every time u, the decision maker's thinking process Γ_u, as a whole, obeys (relating to some action a) the probability rule of the bottom process $\xi(t)$ starting from the state z and terminates at time $[\overline{w}(u)]$, the living time $\overline{w}(u)$ of $\{\Gamma_u,\ u \geq 0\}$ varies according to the observation time u, and thereby Γ_u shows its trajectory in a twisting way, intuitively just like a crawling snake. Thus it is called Brownian snake considering $\overline{w}$ is the trajectory of reflective Brownian movement. When $\overline{w}(u)$ is not integer, this indicates the minder stops thinking at time $[\overline{w}(u)]$ without having to get any thinking results, and turns to the next $\Gamma_{u'}$. Yet at this time, $\xi(\overline{w}(u))$ (which has no definition) is replaced by $\xi([\overline{w}(u)])$.

Now if we stipulate that if $u > \tau$, then let $\Gamma_u(t) = z$, $t = 0,1,2,\cdots$, then the process $(\Gamma_u, u \geq 0)$ is defined on the whole axis.

Let $Q_z^{\overline{w}}$ be the probability rule of $(\Gamma_u(t), u \geq 0, \Gamma_u(0) = z, t = 0,1,2,\cdots,[\overline{w}(u)])$ decided by $\overline{w}$. Relating with the Itô measure on Brownian trajectories, we can define measure M_z on all the Brownian snakes $(\Gamma_u(t), u \geq 0, \Gamma_u(0) = z, t = 0,1,2,\cdots,[\overline{w}(u)])$ of all the Brownian trajectories starting from z as follows:

$$M_z(d\overline{w}, dw) = n(d\overline{w})Q_z^{\overline{w}}(dw) \tag{1}$$

where w is some "trajectory" (realization) of $(\Gamma_u(t), u \geq 0, \Gamma_u(0) = z, t = 0,1,2,\cdots,[\overline{w}(u)])$.

Let Θ be a "two-dimensional" space,

$$\Theta = \left\{ (\overline{w}, w) \left| \begin{array}{l} \overline{w} \text{ is one trajectory of the Brownian movement. } w \text{ is one trajectory of} \\ \text{the Brownian snake } (\Gamma_u(t), u \geq 0, \Gamma_u(0) = z, t = 0,1,2,\cdots,[\overline{w}(u)]). \end{array} \right. \right\}$$

Then M_z is just the σ-finite measure on Θ, that is for every "sample" $(\overline{w}, w)$ in Θ, its value is $M_z(d\overline{w}, dw) = n(d\overline{w})Q_z^{\overline{w}}(dw)$.

2.4 Brownian Snake Measure-Valued Process

On space Θ, we define a random variable Y_t ($t > 0$) which takes measure value on state space S, that is the value of Y_t at the sample $(\overline{w}, w)$ is a measure on S. If ϕ is an arbitrary function on S, then

$$\int_S \phi(s) Y_t(\overline{w}, w)(ds) = \sum_{i=1}^{n} \phi(s^i) Y_t(\overline{w}, w)(s^i) \triangleq \langle Y_t(\overline{w}, w), \phi \rangle \tag{2}$$

is a random variable which takes real number on Θ, and the expectation of this random variable $\langle Y_t(\overline{w}, w), \phi \rangle$ (the notation of this expectation is still written as M_z) is (t is a positive integer):

$$
\begin{aligned}
M_z\langle Y_t(\overline{w},w),\phi\rangle &= \int n(d\overline{w})\int_0^\infty d_sL_s^t(\overline{w})Q_z^{\overline{w}}(\phi(\Gamma_s(t)))\\
&= \int n(d\overline{w})\int_0^\infty d_sL_s^t(\overline{w})E_z\phi(\xi(\overline{w}(s)))\\
&= E_z\phi(\xi(t))\int n(d\overline{w})\int_0^\infty d_sL_s^t(\overline{w}) \qquad (3)\\
&\qquad (\text{ Because } d_sL_s^t(f) \text{ has mass only at} f(s)=t\,)\\
&= E_z\phi(\xi(t))\int n(d\overline{w})\cdot L_\infty^t(\overline{w}) = S_t\phi(z)\\
&\qquad (\text{Because } n(L_\infty^a)=1)
\end{aligned}
$$

where S_t denotes the semigroup of ξ, i.e., $S_t\phi(z)=\sum_{i=1}^{n}\phi(s^i)T^t(z,a,s^i)$, where $T^t(z,a,s^i)$ denotes such a probability that the bottom process ξ first takes action a in state z, and transfers to state s^i after t steps. If t is an arbitrary positive real number, then in above formulas t related to bottom process is replaced by positive integer $[t]$, and the approximate value of $M_z\langle Y_t(\overline{w},w),\phi\rangle$ can be attained. The above Y_t corresponds to initial state z, that is, the initial measure is the unit measure δ_z on state space. As for a general initial measure on state space S, how to define a measure-valued process? We can first define a Poisson measure $\mathcal{N}$ on $S\times\Theta$ whose intensity is $\mu(dz)M_z(d\overline{w},dw)$. Then let

$$
\langle X_t,\phi\rangle=\begin{cases}\langle\mu,\phi\rangle & t=0\\ \int\mathcal{N}(ds,d\overline{w},dw)\langle Y_t(\overline{w},w),\phi\rangle & t>0\end{cases} \qquad (4)
$$

where $\phi\in pb\mathcal{B}(S)$, i.e., ϕ is a non-negative bounded $\mathcal{B}(S)$ measurable function on S. So $X=(X_t,t\geq 0)$ is a measure-valued process whose bottom process is ξ and initial measure is μ . (See [2] P.125)

So X_t is called Brownian snake measure-valued process. It is easy only to think that the time is discrete in practice, i.e., X_t , $t=0,1,2,\cdots$. It is abbreviated as BSMP.

Intuitively, X_t is such a measure on state space that is attained by integrating the measure-valued random variable $Y_t(\overline{w},w)$ with Poisson measure relating with initial measure on $S\times\Theta$ through traversing all Brownian motion trajectories and all trajectories of Brownian snake decided by one of the Brownian motion trajectories. Because initial measure disperses in state space everywhere, and the value of Y_t depends on every Brownian motion trajectory and every Brownian snake on it, it is true that X_t , as a measure, connects events that are far away in time or space. So this measure can be used to measure state space, that is, it makes an estimation to every state. The above statement is about one certain action a. If there is m actions, then we have m BSMPs. We use $X(a)$ to denote the BSMP corresponding action a, i.e.,

$(X_t,\ t=0,1,2,\cdots)$ corresponding action *a*. Let $\mathfrak{X}=\{X(a),\ a=a^1,a^2,\cdots,a^m\}$, and $\{\mathfrak{X},A,T,R\}$ is called Brownian snake measure-valued Markov decision process, abbreviated as BSMMDP. Here *A*, *T*, *R* are state space, transition probability and reward function on the bottom process respectively.

2.5 Searching for Optimal Policy of BSMMDPs

If $\phi(s)$ in (4) is replaced by the reward function $R(s)$, then (4) is changed to:

$$\langle X_t,R\rangle=\begin{cases}\langle\mu,R\rangle & t=0\\ \int\mathcal{N}(ds,d\overline{w},dw)\langle Y_t(\overline{w},w),R\rangle & t=1,2,\cdots\end{cases}\tag{5}$$

Intuitively, (5) means the average of all the rewards about Poisson measure when every Brownian snake, starting from every state *s* (its initial measure is $\mu(s)$) at time *t* and obeying $\xi(\bullet)$ of bottom process (the first action is *a*), terminates at *t*. Notice that the above average relies on initial measure μ and action *a*, and so the exact notation should be $\langle X_t(a,\mu),R\rangle$.

Similar to traditional *Q*-value function, we define measure-action value function as:

$$Q(\mu,a)=\sum_{t\in T}\gamma^t\langle X_t(\mu,a),R\rangle\tag{6}$$

where $T=\{0,1,2,\ \cdot\cdot\cdot\}$, γ is a discount factor, $0<\gamma<1$. So for the initial measure μ, an optimal action can be chosen:

$$\pi^*(\mu)=\arg\max_a Q(\mu,a)\tag{7}$$

If we treat the measure as the belief state of agent for μ can be arbitrary measure, then the policy found above is just the optimal policy corresponding to every belief state in which the agent will choose optimal action.

It can be seen from the above statement that we also give a methodology of finding optimal policy the moment we complete constructing the model. So BSMMDP is a construction-oriented algorithm, and unlike POMDP (partially observable MDP), it has not an obvious belief state space, nor does it talk about the transition probability of belief state. When the agent thinks he is in a certain belief state—that is he takes into an overall consideration of the state space, and he will construct the measure $X_t(a)$, $a\in A$, $t\in T$ when he uses measure μ to denote this consideration, then he will compute $Q(\mu, a)$. Moreover, he will compute the optimal action that should be taken at the belief state μ.

3 Approximate Solution to the Optimal Policy of BSMMDP

It has been said that BSMMDP is construction-oriented, and so according to its constructive "semantics" it is easy to get an approximate method of the optimal policy.

(1) Suppose the agent's belief state is μ, which is a measure on state space. Suppose $\mu(s^i) = c_i$, $i = 1, 2, \ldots, n$. According to (3) and (4), we can get (in approximate sense): corresponding some action a,

$$\langle X_t, R\rangle = \begin{cases} \langle \mu, R\rangle = \sum_{i=1}^{n} R(s^i)\mu(s^i) = \sum c_i R(s^i) & t = 0 \\ \sum_{i=1}^{n} \mu(s^i) S_{a,t} R(s^i) = \sum_{i=1}^{n} c_i S_{a,t} R(s^i) & t = 1, 2, \cdots \end{cases} \tag{8}$$

where $S_{a,t}R(s^i) = \sum R(s')T^t(s^i, a, s')$, and $T^t(s^i, a, s')$ denotes such probability that $\xi(\bullet)$ first takes action a in s^i and transfers to s' after t steps. Having $<X_t, R>$, it is not hard to compute:

$$Q(\mu, a) = \sum_{t=0}^{\infty} \gamma^t \langle X_t, R\rangle \tag{9}$$

(Note that Q-function can be attained by traditional Q-learning method.) Finally we get the approximate optimal policy:

$$\pi^*(\mu) = \arg\max_a Q(\mu, a) \tag{10}$$

If the agent's belief state has changed, then the above process is repeated on the new belief state and then an optimal action will be found on the new belief state. How attaining belief state yet depends on specific problem.

(2) If we consider BSMMDP $\{X_t\}$, $t \in T$ as a series of belief states attained by the agent with time under the initial belief μ, and the whole grasp of the agent to the state space varies with time, then we can approximately compute the $<X_t, R>$ of every action a according to (8), and write it as $<X_t, R>_a$. Then let

$$\pi^*(X_t) = \arg\max_a \langle X_t, R\rangle_a \quad t = 0, 1, 2, \cdots \tag{11}$$

which is the optimal policy the agent should take at every step.

This methodology of optimal policy reflects general habit of human beings who always decide their actions on the basis of their own experiences. For example, $\sum R(s')T^t(z, a, s')$ reflects the history experience that the agent acts t times after choosing action a in state s.

4 Conclusion

The main objective of this paper is to present a BSMMDP model that is able to simulate an important phenomenon of human thought. People are able to combine different events that are far away in time and space to think problems. This characteristic, for regular decision problems, corresponds that people solve problems by using accumulate experience and knowledge, while for non-regular (i.e., very difficult or new) decision problems, corresponds that people creatively solve problems relying on inspiration inspired by tangling various things in time and space. Our model embodies the two sides. As a subobjective, the paper provides an approximate meta-algorithm for finding an optimal policy within this model. This algorithm is promising because the exact solution of optimal policy of BSMMDP is generally infeasible in practice.

The paper provides a new understanding of artificial intelligence. In order to create high intelligent "machines", artificial intelligence should simulate some behaviors of human thought. If it could do this, a great advance to "intelligent human" is made. Following this idea, we simulate some pattern of human thought--the associative ability. When encountering problem, one always first wants to use his experiences and knowledge to think. Although the thinking process is primarily in a logic pattern, it essentially is random in subconsciousness. The so-called logic pattern means the inference of thinking is not casual but continuous. Such continuity usually takes causality as its fundamental. As to randomicity, it is not definitely "moving" but "vibrating" which is controlled by feeling, wish or fantasticism in subconsciousness. When, especially, a problem cannot be solved for a long time, one must turn to creative ability, not only depending on experience and knowledge. The measure-valued process $\{X_t\}_{t \geq 0}$ described above is an important factor in drawing creative ability of human being. It can link various episodes that are remote in space-time together, and all of these episodes put together will touch mind, draw inspiration, and in the end, solve the problem. We believe that such "human-like" structure will help one to more deeply understand and develop artificial intelligence.

Acknowledgments. This work is supported by the Research Fund of Jinling Institute of Technology (Grant No. jit-b-201207).

References

1. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn., pp. 626–627. Pearson Education Asia Limited, Tsinghua University Press (2006)
2. Zhao, X.L.: An Introduction of Measure-Valued Branching Processes, vol. 70, pp. 43–58. Science Press, Beijing (2000) (in Chinese)
3. Lovejoy, S.: A Survey of Algorithmic Methods for Partially Observed Markov Decision Processes. Annals of Operations Research 28, 47–66 (1991)
4. Sondik, J.: The Optimal Control of Partially Observable Markov Processes. PhD Thesis, Stanford University (1971)
5. Roy, N., Gordon, G.: Finding Approximate POMDP Solutions Through Belief Compression. Journal of Artificial Intelligence Research 23, 1–40 (2005)
6. Chatterjee, K., Doyen, L., Henzinger, T.A.: Qualitative Analysis of Partially-Observable Markov Decision Processes. In: Hliněný, P., Kučera, A. (eds.) MFCS 2010. LNCS, vol. 6281, pp. 258–269. Springer, Heidelberg (2010)

A Strategy to Regulate WSN Nodes' Energy Consumption Based on Emission Rate

Bo Song, Yan Wang, and Hailong Zhang

Shenyang Normal University College of Software, Liaoning Shenyang 110034, China
songbo63@yahoo.com.cn, wyan428@hotmail.com,
zhanghailong88@163.com

Abstract. For the necessary of low-power data transmission in wireless sensor networks (WSN), this work is main for the data characteristics' analysis of WSN and so we have create a data model based on note load method. And finally we get the energy consumption of the sensor network model and its data transmission delay model. We have report the network lifetime maximization solution algorithm on the premise of ensure the application delay requirements. The theoretical analysis and simulation results shown that the method can effectively extend the network lifetime.

Keywords: wireless sensor networks, energy consumption shared, data transmission, network lifetime.

1 Introduction

Because of its powerful abilities of data acquisition, signal processing and wireless communications, wireless sensor networks (WSNs) have been widely used in environment monitoring, military reconnaissance, etc. In WSN, nodes in wireless sensor networks rely on battery power, yet the node energy is limited, and thus minimizing the WSN energy consumption becomes a very important factor in the design of routing. WSNs generally use node hop-by-hop transmission mode for data transmission, and a forward node which itself may also play the role of collecting information. So the forward node which is the more closed to the sink node bears the more data volume, and its energy consumption brings the more burden. This causes some nodes under the circumstance which the overall network energy is also very abundant premature lose theirs effectiveness due to theirs own energy depletion, and then lead to death of the entire transmission paths, i.e. the maximum energy node determines the lifetime of the network path [1][2]. So, an effective method to prolong the network lifetime is sharing the energy consumption of nodes to reduce the maximum energy consumption. How to effectively reduce the maximum energy consumption of nodes, some effective algorithms and satisfactory test results have been given [3], but many of these algorithms require the large analysis and complex calculation to ensure its accuracy, and these will aggravate the energy burden of nodes. In this paper, according to the characteristics of WSN, the main parameters

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 267–274, 2013.

number of hops will be used to construct the data model of nodes bearing and the energy consumption model of nodes, and to give the algorithm which to determine the sending rate of nodes based on hop count, and to validate the model and algorithm by the simulation experiments which the parameters of environment is certain.

2 The Theories

To research on WSN node energy consumption, first we need to build wireless communication model of energy consumption and node data send delay model. Wireless communication energy consumption is divided into the transmission energy and receive energy, their energy consumption models are showed by Equation (1) and (2) [4].

$$E_{\text{memt}} = s \cdot E_{\text{elec}} + s \cdot \varepsilon_{\text{elec}} \cdot d^{\alpha} \tag{1}$$

$$E_{\text{memr}} = s \cdot E_{\text{elec}} \tag{2}$$

In the equations, E_{elec} is sending and receiving circuit energy consumption, only associates with the device; d is node signal transmission distance; s is the carrying amount of data; α is the distance parameter, associates with the communication distance.

The sending delay model of node data is showed as Equation (3) and (4) [5].

$$\tau_{\text{memt}} = \frac{s}{b \cdot \Re} \tag{3}$$

$$\Gamma = \sum \tau_{\text{memt}} = \sum_{i=1}^{n} \frac{s_i}{b_i \cdot \Re} \le \Gamma_{\max} \tag{4}$$

In the model, n is hop count of the path; b is the sending rate, and it is generally that wireless sensor nodes are equipped with multiple transmit rate; $\Re$ is the channel code rate, depending on the transmission protocol what is used, this paper chooses IEEE 802.15.4 protocol.

If we want to control the change of energy consumption by regulation the sending rate b, the energy consumption model of node need to be transformed as following [5]:

$$E_{\text{memt}} =| C \cdot | 2^{\frac{1}{\tau \cdot \Re}} - 1 | + F | \cdot \tau \cdot \Re = [C \cdot (2^{b} - 1) + F] \cdot \frac{s}{b} \tag{5}$$

$$E_{\text{memr}} = F \cdot \frac{s}{b} \tag{6}$$

Thus the total energy consumption of nodes is:

$$E_{\text{mem}} = E_{\text{memt}} + E_{\text{memr}} = [C \cdot (2^b - 1) + 2F] \cdot \frac{s}{b} \tag{7}$$

So, for a sensor network which its topology has been indentified, the energy consumption of nodes E_{mem} relates to node transmission data quantity s and transmission rate b. If s is considered as a parameters after the network path has been determined, and the system parameters C and F are close constant values [6]， then, change rule of E_{mem} can be obtained only by examining the property of $y = \frac{2^b}{b}$, which is showed as figure 1(a); As well as to examine the rule of τ_{memt} only need to the property of $y = \frac{1}{b}$, which is showed as Fig. 1(b).

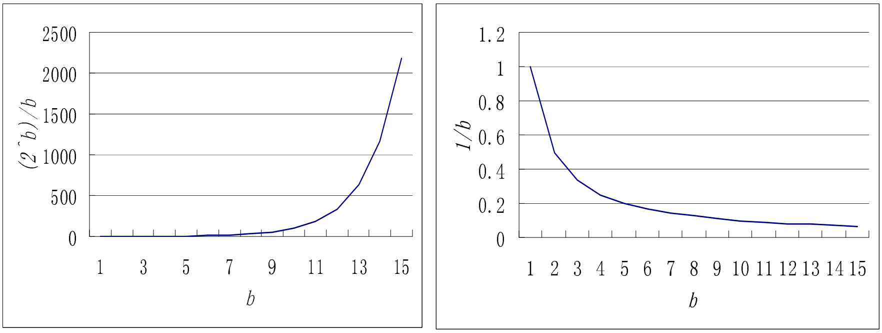


Fig. 1. (a) $y = \frac{2^b}{b}$; (b) $y = \frac{1}{b}$

We can see from figure 1 that data delay reduces with the increasing of b, but the energy consumption of nodes increases. And the trend of growth is faster than the decreasing trend of delay. In order to make the energy consumption shared among nodes to prolong the network lifetime, and to ensure the data delay is not more than the system requirements, the node which carrying a large quantity of data will reduce the sending rate to reduce the maximum energy consumption. At the same time, the sending node which has the small amount of data will increase the sending rate to ensure the total delay system which is not more than the system requirements limit.

3 Establishment of Data Models about the Hop Count

According to the characteristic of WSN, the network node is approximately well-distributed, and the structure of network topology based the number of hops can be considered as a concentric circle distribution which the sink node is the centre of the circles, as shown in Fig.2 (a). The forward nodes which have the same hops from the sink node are in the same concentric circle, and the concentric circle's radius is the hop count of node. The number of nodes in each concentric circle has something with jump number. If each node has only 1 units of information, the nodes sending data model can be created.

Set the number of nodes for each jump is:

$$X_i = 2\pi \cdot i \cdot x \qquad i \in \{1, \ldots, \ n\} \tag{8}$$

Where, n is hop-count, x is the parameter about the relationship of node-number in each hop and hop-counts of nodes. Since each node only has 1 units of information, the data volume which is generated by each node is X_i. Therefore, the amount of data model in a node bearing for is:

$$s_i = \frac{X_{i+1}+\ldots+2\pi\cdot n\cdot x}{X_i}+1 = 2\pi\cdot x\cdot\frac{(n+1+i)\cdot(n-i)}{2\cdot 2\pi\cdot i\cdot x}+1 = \frac{(n+1+i)\cdot(n-i)}{2\cdot i}+1 \tag{9}$$

Based on the equation (9), Fig. 2(b) gives the changes of nodes bearing data with the number of hops in different hop n networks. From the figure we can see that with the increase of n, the increase trend which nodes send data volume is enhancement. Fig.2 shows that node bearing data model (Equation (9)) basically reflects the objective condition of sending data quantity by WSN node.

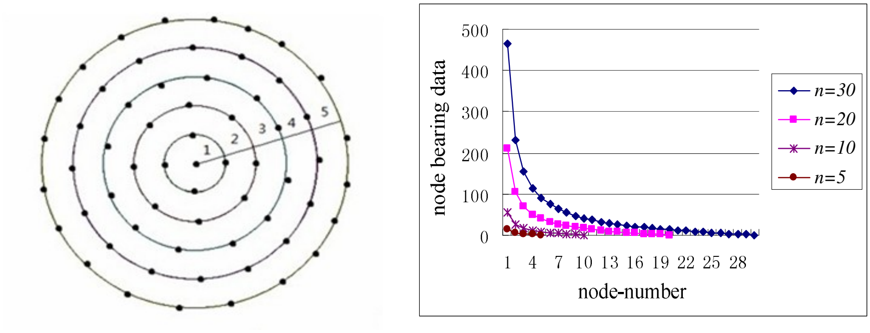


Fig. 2. (a) Network structure which $n = 5, x = \frac{4}{2\pi}$; (b) The condition of the network bearing data with different hop nodes

4 Algorithm

To address the problem which energy consumption of the nodes which has more data bearing is relatively larger, and seriously affect the network lifetime, based on the nodes bearing data model (Equation 9), nodes sending data delay model (Equation 3 and Equation 4) and wireless communication node energy consumption model which has been established in the previous section, this section gives a energy consumption shared algorithm which using different node sending rate.

The main idea of the algorithm is to reduce the energy consumption by reducing the data transmission rate of higher energy consumption of nodes, at the same time through increased the data transmission rate of lower energy consumption node, to ensure that the whole system data delay is not more than the system requirements, to complete the network nodes energy consumption shared to prolong the network lifetime. Fig. 3 is the flow chart of algorithm.

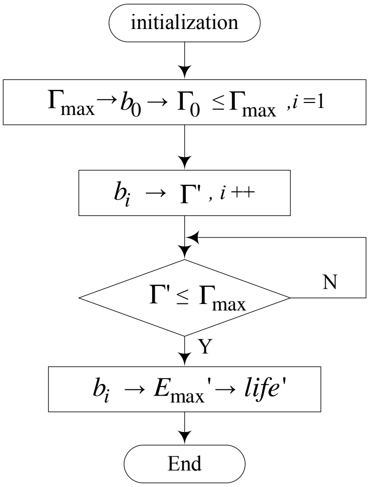

initialization
$\Gamma_{\max} \to b_0 \to \Gamma_0 \le \Gamma_{\max}$,i =1
$b_i \to \Gamma'$, i++
$\Gamma' \le \Gamma_{\max}$
N
Y
$b_i \to E_{\max}' \to life'$
End

Fig. 3. Flow chart of algorithm

Step 1: Initialization:

Input: the channel code rate $\Re$, the maximum delay which system allowable $\Gamma_{\max}$, the initial energy of the nodes E_0 , and node bearing data sending data, data sending delay and energy consumption model.

Output: the number of hops which execute larger transmission rate and the corresponding network lifetime.

Step 2: All nodes of *b* are taken as the same maximum value b_0 which can meet $\Gamma_0 \le \Gamma_{\max}$ (to determine the initial value of *b*).

Step 3: All the nodes of *b* are taken as b' which is smaller value than b_0 , to get the value Γ_0 (by reducing the transmission rate of b to reduce power consumption of node which bearing a large amount of data).

Step 4: Compare Γ_0 and $\Gamma_{\max}$, there will be Γ_0 is larger (by increasing the nodes' sending rate which has less data to reduce network delay).

```
i=1
$b_i$ is the value $b''$ which is larger than $b_0$
Then get the value $\Gamma'$
i++
```

Step 5: Compare Γ' and $\Gamma_{\max}$ (through the cycle the sending rate of fewer data nodes is increased successively, at the expense of its energy to reduce transmission delay, until it meets the system requirements).

If $\Gamma' > \Gamma_{\max}$, go to step 4

If $\Gamma' \le \Gamma_{\max}$, go to step 6

Step 6: i which meet $\Gamma' \le \Gamma_{\max}$ is the number of implement b'', network lifetime $life' = E_0 / E_{\max}'$.

Step 7: End.

5 Simulation and Performance Analysis

In the argument in table 1 as the simulation environment, the results of Fig. 4-5 and table 2 can be gotten by the algorithm which be given in section 4.

Table 1. System initialization data

b	x	n	b_0	b'	b''	C(J·bit·m$^{\alpha-1}$)	F(J·bit^{-1})	$\Re$(bit/s)	$\Gamma_{\max}$ (ms)	E_0 (J)
2,4,6,8	$\frac{4}{2\pi}$	30	4	2	6,8	7×10^{-9}	1×10^{-8}	20×10^{3}[7]	25,30,35	0.5

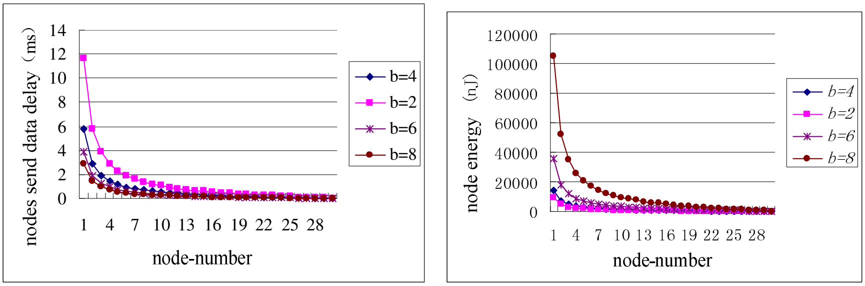


Fig. 4. (a) Nodes send data delay; (b) energy conservation of nodes

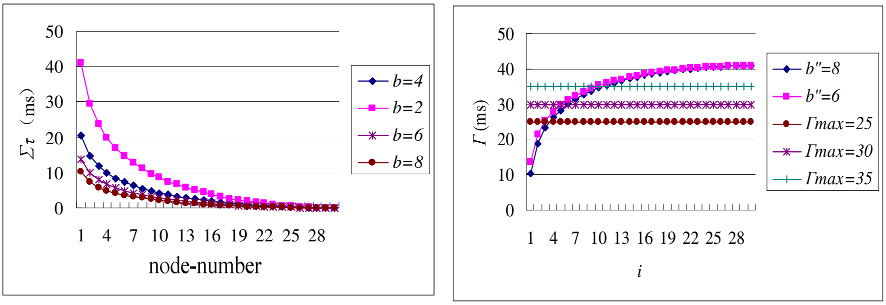


Fig. 5. (a) The total amount of delay of each node; (b) Γ with the change of i

Table 2. Comparison of algorithms

$E_{\max}$ (nJ)	*life*	Comparative index		*i*	$E_{\max}$' (nJ)	*life'*	Performance improvement (%)
14531.25	34408	$\Gamma_{\max}$ =25ms	b ''=6	2	17825.33333	28049	-22.67
			b ''=8	3	34746.25	14390	-139.11
		$\Gamma_{\max}$ =30ms	b ''=6	5	6991.833333	71512	51.88
			b ''=8	5	20531.875	24352	-41.29
		$\Gamma_{\max}$ =35ms	b ''=6	9	3662.388889	136522	74.80
			b ''=8	10	9476.25	52763	34.79

For comparison of test results, a newly proposed node energy consumption shared strategy was compared with the algorithm which this paper proposed [8]. The main idea of the newly algorithm is that a part of nodes which far from the sink is responsible for forwarding data not by forwarding nodes which are close to sink, but directly sent to the sink, in order to reduce the energy consumption of the forwarding nodes which their distance to sink is closer. Thus, the change ratio of the data forwarding volume is important in this algorithm. According to the calculation, when the ratio ϑ is 30%, it can effectively reduce data capacity of the forwarding node which the energy consumption is largest in network and energy consumption.

Through the analysis of simulation results, and the comparison with other energy optimization scheme, the following conclusions for the energy consumption optimization strategies that this paper proposed can be drawn:

- Nodes bearing data model and the node energy consumption model can objectively reflect the real situation of WSN nodes, and meet the system requirements;
- The algorithm proposed in this paper shows that in most conditions, specifically the difference of $\Gamma_{\max}$ and Γ_0 is not small, this algorithm can more effectively achieve network lifetime. In order to extend the network lifetime by using this algorithm in more cases, combination with the other mature node energy consumption optimization algorithms can be considered;
- Although the larger signal transmission rate b can reduce node data transmission delay effectively, it will also cause the relatively larger growth trend of energy consumption. So the value b'' should be chosen relatively close to b_0 which is mentioned in this algorithm; similarly, in the choice of b' , it should not excessive pursuit of reducing energy consumption, but select it close to the value of b_0 .

6 Conclusion

In order to improve the situation that in WSN some nodes run out of energy quickly caused premature death of entire network, this paper constructs a node data model based on node hop and a node energy consumption model according to network topology characteristic. And on the basis of the two models, proposed the simple optimization algorithm which using different data transmission rate reached the purpose of the node energy consumption of network nodes share. The simulation results has confirmed that the node model and optimization algorithm can improved to the condition of the network in a certain extent, but in the process of simulation network environment is an ideal environment that to assume that the network topology of uniform distribution and no conflict communication channel and no disturbance. Therefore, in the next study it should increase to study the change of the environment and make a more comprehensive consideration for these changes.

References

1. Song, C., Liu, M., Gong, H.-G., et al.: ACO-based algorithm for solving energy hole problems in Wireless Sensor Networks. Journal of Software 20(10), 2729–2743 (2009)
2. Lian, J., Naik, K., Agnew, G.: Data capacity improvement of wireless sensor networks using on-uniform sensor distribution. International Journal of Distributed Sensor Networks 2(2), 121–145 (2006)
3. Yu, R., Sun, Z., Zhou, H.-J., et al.: QoS and energy aware routing algorithm for wireless sensor networks. Journal of Tsinghua University (Science and Technology) 47(10), 1634–1637 (2007)
4. Chen, G., Li, C.F., Ye, M., Wu, J.: An unequal cluster-based routing strategy in wireless sensor networks. Wireless Networks (JS) 15(2), 193–207 (2009)
5. Yang, Y., Krishnamachari, B.: Energy-latency tradeoffs for data gathering in wireless sensor networks. In: Prasanna, V.K. (ed.) Proceedings of the INFOCOM 2004, vol. 1, pp. 7–11 (2004)
6. Raghunathan, V., Schurgers, C., Park, S., Srivastava, M.B.: Energy aware wireless microsensor networks. IEEE Signal Processing Magazine 19(2), 40–50 (2002)
7. IEEE 802.15.4 2003: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)
8. Zeng, Z.-W., Chen, Z.-G., Liu, A.-F.: Energy-Hole Avoidance for WSN Based on Adjust Transmission Power. Chinese Journal of Computers 33(1), 12–22 (2010)

Contact Network Model with Covert Infection

Xiaomei Yang[1,2], Jianchao Zeng[1], and Jiye Liang[2]

[1] Complex System and Computer Intelligence Laboratory,
Taiyuan University of Science& Technology, Taiyuan 030024, China
[2] School of Management, Shanxi University, Taiyuan 030006, China

Abstract. Through analyzing the type of infected population, a contact model with covert infection is introduced. The stability of this model in small world network is studied according to the property of contact network. To the difference of immune pattern, the effectiveness and feasibility of the proposed model is validated.

Keywords: small world network, contact model, covert infection.

1 Introduction

Recently, kinds of epidemic diseases had a great impact on human's normal life. In order to predict the trends of these diseases correctly and make an effective control method, it is necessary to analysize the spread of infectious diseases mechanisms. By far, scholars have proposed some epidemiological models, such as SI, SIS and SIR model[1]. Besides, other models are designed for further study in the transmission rule of the infectious diseases. For example, Li M Y investigated the role of incubation in disease transmission and introduced a SEIR model for the transmission of an infections disease that spreads in a population through direct contact of the hosts[2]. To model the SARS epidemic, Li B built a SEIuIiR model with self-cure [3]. With increasing maturity of complex network, imitating people's contact relations, many disease transmission network model were introduced [4,5]. Considering a few of diseases are spread via the proximity contact between persons, the research on people's contact behaviors showed small-world characteristics [6]. In epidemiology, the infections are classified into overt infections and covert infections [7]. However, the infections were only assumed to overt in current research. Thus, through analyzing people's proximity contact behaviors, a contact model with covert infection in small world network is proposed by virtue of complex social theory.

2 SIAR Model

Considering the infections are divided into overt infections and covert infections, a new epidemic transmission model is proposed which is named as SIAR model. In this model, the popution is divided into five types including susceptible, infectioins,

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 275–283, 2013.

isolated and recoverd. We assume S, I, A, R, N as the number of susceptible, infectioins, isolated, recoverd and the total population size. The susceptible population is infected at a constant rate α. Among the infections, the overt infections are isolated at a constant rate β_1 and the covert infections are vaccinated at a constant rate β_2. The isolated population is cured at a constant rate μ by the correct treatment. The vaccine wears off at a constant rate η. Then, the disease transmission route is showed as Fig.1.

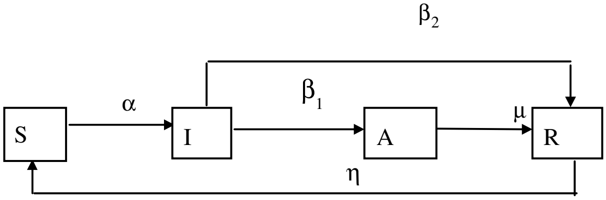


Fig. 1. The transfer diagram for SIAR model

From Fig.1, the SIAR model can be found as

$$\begin{cases} \dfrac{dS}{dt} = -\alpha SI + \eta R \\ \dfrac{dI}{dt} = \alpha SI - \beta_1 I - \beta_2 I \\ \dfrac{dA}{dt} = \beta_1 I - \mu A \\ \dfrac{dR}{dt} = \mu A + \beta_2 I - \eta R \\ S + A + I + R = N \\ S, I, A, R \geq 0 \end{cases} \tag{1}$$

3 The SIAR Model in Complex Network

The tradition epidemic model only used the differential equations to explain the disease transmission mechanics. By means of complex network theory, face-to-face proximity contact behavior mostly shows small-world characteristic in the present research [8]. Therefore, the transmission principle of SIAR model is proposed in the small-world network. Let $s(t), i(t), a(t), r(t)$ be the fractions of susceptible, infectioins, isolated, recoverd at time t. Using $s(t) = \frac{S}{N}$, $i(t) = \frac{I}{N}$, $a(t) = \frac{A}{N}$, $r(t) = \frac{R}{N}$, the SIAR model in small-world network is as follows.

$$\begin{cases} \dfrac{ds(t)}{dt} = -\alpha\langle k\rangle s(t)i(t) + \eta r(t) \\ \dfrac{di(t)}{dt} = \alpha\langle k\rangle s(t)i(t) - \beta_1 i(t) - \beta_2 i(t) \\ \dfrac{da(t)}{dt} = \beta_1 i(t) - \mu a(t) \\ \dfrac{dr(t)}{dt} = \mu a(t) + \beta_2 i(t) - \eta r(t) \end{cases} \tag{2}$$

For $s(t) + a(t) + i(t) + r(t) = 1$, the equation (2) is simplified as

$$\begin{cases} \dfrac{ds(t)}{dt} = -\alpha\langle k\rangle s(t)i(t) + \eta - \eta i(t) - \eta s(t) - \eta a(t) \\ \dfrac{di(t)}{dt} = \alpha\langle k\rangle s(t)i(t) - \beta_1 i(t) - \beta_2 i(t) \\ \dfrac{da(t)}{dt} = \beta_1 i(t) - \mu a(t) \end{cases} \tag{3}$$

Let $\dfrac{ds(t)}{dt} = 0, \dfrac{di(t)}{dt} = 0, \dfrac{da(t)}{dt} = 0$, we can obtaion two equilibrium solutions as

$$(s,i,a) = (1,0,0) \text{ and } (s,i,a) = \left(\frac{\beta_1+\beta_2}{\alpha\langle k\rangle}, \frac{\eta(1-\frac{\beta_1+\beta_2}{\alpha\langle k\rangle})}{\beta_1+\beta_2+\eta+\frac{\eta\beta_1}{\mu}}, \frac{\eta\beta_1(1-\frac{\beta_1+\beta_2}{\alpha\langle k\rangle})}{\mu(\beta_1+\beta_2+\eta+\frac{\eta\beta_1}{\mu})}\right).$$

Assume that

$$\begin{cases} u(s,i,a) = -\alpha\langle k\rangle s(t)i(t) + \eta - \eta i(t) - \eta s(t) - \eta a(t) \\ v(s,i,a) = \alpha\langle k\rangle s(t)i(t) - \beta_1 i(t) - \beta_2 i(t) \\ w(s,i,a) = \beta_1 i(t) - \mu a(t) \end{cases} \tag{4}$$

The Jacobin matrix is derived as

$$J[u(s,i,a), v(s,i,a), w(s,i,a)] = \begin{vmatrix} -\alpha\langle k\rangle i(t) - \eta & \alpha\langle k\rangle i(t) & 0 \\ -\alpha\langle k\rangle s(t) - \eta & \alpha\langle k\rangle s(t) - \beta_1 - \beta_2 & \beta_1 \\ -\eta & 0 & -\mu \end{vmatrix} \tag{5}$$

Substituting the first solution $(s,i,a) = (1,0,0)$ into (5) J_1 is given that

$$J_1 = \begin{vmatrix} -\eta & 0 & 0 \\ -\alpha\langle k\rangle - \eta & \alpha\langle k\rangle - \beta_1 - \beta_2 & \beta_1 \\ -\eta & 0 & -\mu \end{vmatrix} \tag{6}$$

The characteristic equation can be found that

$$\lambda^3+[\mu+\eta-(\alpha\langle k\rangle-\beta_1-\beta_2)]\lambda^2+[\mu\eta-(\mu+\eta)(\alpha\langle k\rangle-\beta_1-\beta_2)]\lambda\ -\mu\eta(\alpha\langle k\rangle-\beta_1-\beta_2) \tag{7}$$

Let $a_3=1,\ a_2=\mu+\eta-(\alpha\langle k\rangle-\beta_1-\beta_2)$ and

$$a_1=\mu\eta-(\mu+\eta)(\alpha\langle k\rangle-\beta_1-\beta_2),\ a_0=-\mu\eta(\alpha\langle k\rangle-\beta_1-\beta_2).$$

From Eqs.(7), if $\alpha\langle k\rangle-\beta_1-\beta_2<0$, then $a_i>0(i=0,1,2,3)$. We can obtain that

$$\begin{aligned}\Delta_1&=a_2>0,\\ \Delta_2&=a_1a_2-a_0a_3\\ &=\mu\eta\cdot(\mu+\eta)-(\mu+\eta)^2(\alpha\langle k\rangle-\beta_1-\beta_2)+(\mu+\eta)\cdot(\alpha\langle k\rangle-\beta_1-\beta_2)^2>0\end{aligned} \tag{8}$$

According to the stability decision principle of ordinary differential equation, the solution $(s,i,a)=(1,0,0)$ is a stability solution.

Assume that the basic reproduction number $R_0=\dfrac{\beta_1+\beta_2}{\alpha\langle k\rangle}$. So if $R_0>1$, then $\alpha\langle k\rangle-(\beta_1+\beta_2)<0$ and the first solution $(s,i,a)=(1,0,0)$ can be stability gradually.

Substituting the second solution into (5), J_2 is given that

$$J_2=\begin{vmatrix}-\alpha\langle k\rangle\dfrac{\eta(1-\dfrac{\beta_1+\beta_2}{\alpha\langle k\rangle})}{\beta_1+\beta_2+\eta+\dfrac{\eta\beta_1}{\mu}}-\eta & \alpha\langle k\rangle\dfrac{\eta(1-\dfrac{\beta_1+\beta_2}{\alpha\langle k\rangle})}{\beta_1+\beta_2+\eta+\dfrac{\eta\beta_1}{\mu}} & 0\\ -(\beta_1+\beta_2+\eta) & 0 & \beta_1\\ -\eta & 0 & -\mu\end{vmatrix} \tag{9}$$

The characteristic equation can be found that

$$\begin{aligned}&\lambda^3+[\mu+\eta-\frac{\eta(\alpha\langle k\rangle-\beta_1-\beta_2)}{\beta_1+\beta_2+\eta+\dfrac{\eta\beta_1}{\mu}}]\lambda^2+[\mu\eta+\frac{\mu\eta(\alpha\langle k\rangle-\beta_1-\beta_2)}{\beta_1+\beta_2+\eta+\dfrac{\eta\beta_1}{\mu}}\\ &+(\beta_1+\beta_2+\eta)\cdot\frac{\mu(\alpha\langle k\rangle-\beta_1-\beta_2)}{\beta_1+\beta_2+\eta+\dfrac{\eta\beta_1}{\mu}}]\lambda+\eta\beta_1\cdot\frac{\eta(\alpha\langle k\rangle-\beta_1-\beta_2)}{\beta_1+\beta_2+\eta+\dfrac{\eta\beta_1}{\mu}}\\ &+\mu\cdot(\beta_1+\beta_2+\eta)\cdot\frac{\alpha\langle k\rangle-\beta_1-\beta_2}{\beta_1+\beta_2+\eta+\dfrac{\eta\beta_1}{\mu}}\end{aligned} \tag{10}$$

Similarly, when $\alpha\langle k\rangle-\beta_1-\beta_2>0$, the second solution $(s,i,a)=(\frac{\beta_1+\beta_2}{\alpha\langle k\rangle},\frac{\eta(1-\frac{\beta_1+\beta_2}{\alpha\langle k\rangle})}{\beta_1+\beta_2+\eta+\frac{\eta\beta_1}{\mu}},\frac{\eta\beta_1(1-\frac{\beta_1+\beta_2}{\alpha\langle k\rangle})}{\mu(\beta_1+\beta_2+\eta+\frac{\eta\beta_1}{\mu})})$ is a stability solution.

From the above derivation, the following theorems are given.

Theorem 1. For the system (3), the disease-free equilibrium is globally asymptotically stable if $R_0>1$.

Theorem 2. For the system (3), the endemic equilibrium is locally stable if $R_0<1$.

The basic reproduction number $R_0=\frac{\beta_1+\beta_2}{\alpha\langle k\rangle}$ represents the average number of infected contacts generated by infected persons. It indicates the epidemic threshold λ_c in the small world network, then $\lambda_c=\frac{\beta_1+\beta_2}{\alpha\langle k\rangle}$.

4 Simulation Result and Analysis

1. The validation of theorem 1 and theorem 2

Let the small world network scale $n=10000$, $m=3$, $p=0.5$, the parameter $\alpha=0.1$, $\beta_1=0.7$, $\beta_2=0.3$, $\mu=0.9$, $\eta=0.1$, the trends of $S(t), I(t), A(t), R(t)$ is as Fig.2 in ten network realities.

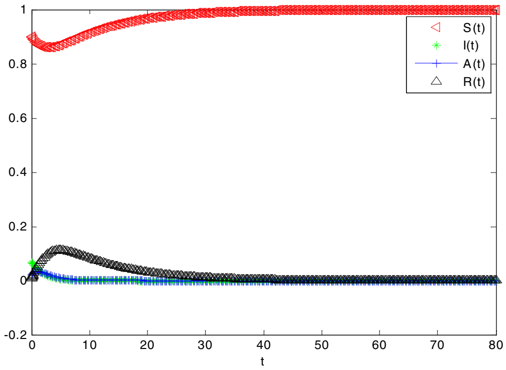


Fig. 2. $S(t), I(t), A(t), R(t)$ of SIAR model in small world network ($\alpha=0.1$)

In Fig.2, $R_0 > 1$. From this figure, it is known that the number of every kind of individuals becomes stable solution $(s,i,a,r) = (1,0,0,0)$ gradually along with the increase of evolution generation. It validates the conclusion of theorem 1.

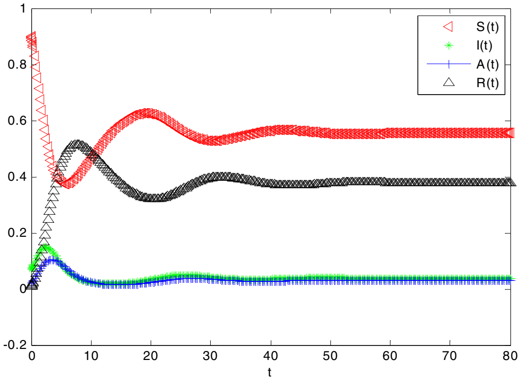


Fig. 3. $S(t), I(t), A(t), R(t)$ of SIAR model in small world network ($\alpha = 0.3$)

Furthermore, assume that $\alpha = 0.3$ and the other parameters are fixed, $S(t), I(t), A(t), R(t)$ is as Fig.3 in ten network realities.

In Fig.3, $R_0 < 1$. From Fig.3, it is known that the number of every kind of individuals becomes stable solution $(s,i,a) = (0.5552, 0.0377, 0.0293)$. The theoretical solution is $(s,i,a) = (0.5556, 0.0377, 0.0293)$ according to theorem 2. So the simulation solution is well consistent with the theoretical prediction of SIAR model.

2. The influence of covert infection rate with transmission trend

(1) In the limited immune pattern

For SIAR model, it means that the immune time has specific time limit when $0 < \eta < 1$. The recovered could transfer into susceptible at a constant rate η after the immune time passed its validity period. To different covert infection rate β_2, assume $0 < \eta = 0.1 < 1$, $\beta_1 = 0.6$ and the other parameters are fixed, the trends of $I(t)$ and $R(t)$ is as Fig.4.

From Fig.4, we know that the more β_2 is, the more $I(t)$ and $R(t)$ are.

According to theorem 2, the steady-state infection density is

$$i(\infty) = \frac{\eta(1 - \dfrac{\beta_1 + \beta_2}{\alpha\langle k\rangle})}{\beta_1 + \beta_2 + \eta + \dfrac{\eta\beta_1}{\mu}} \quad \text{when } R_0 < 1.$$

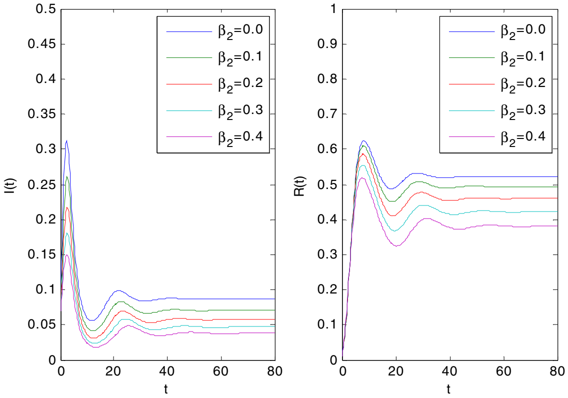


Fig. 4. $I(t), R(t)$ to the different covert infection rate（$\eta = 0.1$）

Let the small world network scale $n = 10000$, $m = 3$, $p = 0.5$, the parameter $\alpha = 0.3$, $\beta_1 = 0.6$, $\mu = 0.9$, $\eta = 0.1$, R_0 and $i(\infty)$ is as Fig.5 to the different covert infection rate. From Fig.5, it is found that there is a positive correlation between β_2 and R_0. However, β_2 and $i(\infty)$ has negative linear relationship.

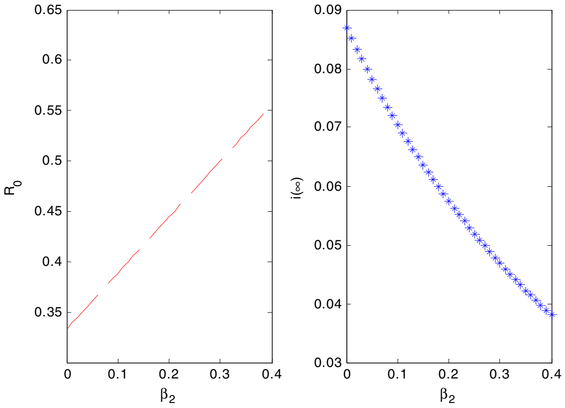


Fig. 5. R_0 and $i(\infty)$ to the different covert infection rate

(2) In the unlimited immune pattern

For SIAR model, the immune time hasn't time limit when $\eta = 0$. The trends of susceptible, infectioins, isolated and recoverd are described as Fig.6 to different covert infection rate. Let $\eta = 0$, the trend of $I(t)$ and $R(t)$ to the different covert infection rate is as Fig.7.

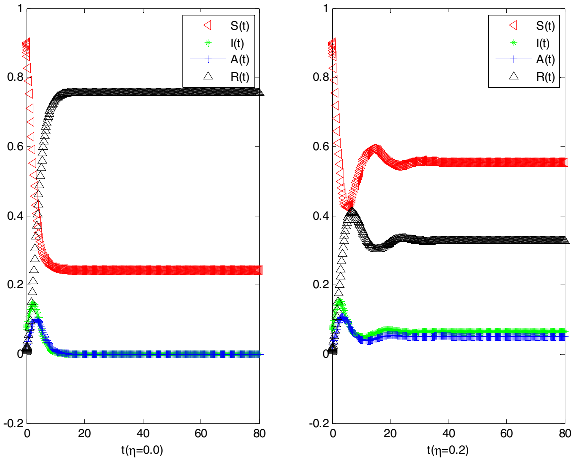


Fig. 6. $S(t), I(t), A(t), R(t)$ of SIAR model in small world network ($\eta = 0.0, 0.2$)

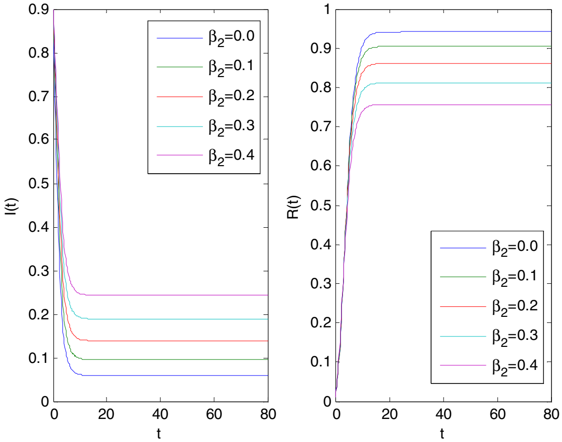


Fig. 7. $I(t), R(t)$ to the different covert infection rate ($\eta = 0$)

In Fig.6, the difference of two subplots is the number of susceptible and recoverd. From Fig.7, it is noted that the more covert infection rate is, the decrease infection individuals are. Nevertheless, the recovered individuals show a contrary tendency.

5 Conclusion

Through analyzing the type of infections, the SIAR model of a contact model with covert infection is introduced. According to the proximity contact between humans and the spread of the disease characteristics, the stability of this model in small world network is studied in this paper which combines with the small world network structure and the specific immune pattern. Then, the effectiveness and feasibility of the proposed model is validated and the covert infection rate's affection to the epidemic spread trend is explored. It is helpful to mine the epidemic mechanics.

Acknowledgments. Project supported by the doctoral fund of Taiyuan University of Science& Technology (Grant No. 20102028).

References

1. Luo, Y.Q., Gao, S.J., Yan, S.X.: Pulse vaccination strategy in an epidemic model with two susceptible subclasses and time delay. Applied mathematics 2, 57–63 (2011)
2. Li, M.Y., Graef, J.R., Wang, L.C., et al.: Global dynamics of a SEIR model with varying total population size. Math. Biosci. 160(2), 191–213 (1999)
3. Li, B., Xu, H.X., Guo, J.J.: Modeling the SARS epidemic considering self-cure. Journal of Engineering Mathematics 20(7), 20–28 (2003)
4. Xia, C.Y., Wang, L., Sun, S.W., et al.: An SIR model with infection delay and propation vector in complex networks 3(69), 927–934 (2012)
5. Bao, Z.J., Jiang, Q.Y., Yan, W.J., et al.: Stability of the spreading in small-world network with predictive controller. Physics Letters A 374(13-14), 1560–1564 (2010)
6. Stehle, J., Vorin, N., Barrat, A., et al.: Simulation of an SEIR infectious disease model on the dynamic contact network of conference attendees. BMC Medicine 9(87), 1–15 (2011)
7. Rothman, K.J.: Epidemiology. An introduction. Oxford University Press, Oxford (2002)
8. Isella, L., Stehlé, J., Barrat, A., Cattuto, C., Pinton, J.F., Van den Broeck, W.: What's in a crowd? Analysis of face-to-face behavioral networks. J. Theor. Biol. 271, 166–180 (2010)

Genetic Evolution of Control Systems

Mu-Song Chen[1], Tze-Yee Ho[2], and Chi-Pan Hwang[3]

[1] Da-Yeh University
[2] Feng Chia University
[3] National Changhua University of Education
chenms@mail.dyu.edu.tw, tyho@fcu.edu.tw, cphwang@cc.ncue.edu.tw

Abstract. In this paper, we present to utilize Genetic Algorithms (GAs) as tools to model control processes. Two different crossover operators are combined during evolution to maintain population diversity and to sustain local improvement in the search space. In this manner, a balance between global exploration and local exploitation is reserved during genetic search. To verify the efficiency of the proposed method, the desired control sequences of a given system are solved by the optimal control theory as well as GA with hybrid crossovers to compare their performances. The experimental results showed that the control sequences obtained from the proposed GA with hybrid crossovers are quite consistent with the results of the optimal control.

Keywords: Genetic Algorithms, Hybrid Crossovers, Exploration and Exploitation.

1 Introduction

Classical control methods for dynamic physical systems require mathematical models to predict their behaviors before appropriate decisions can be made. However, there exist many systems that are too complex to be modeled accurately. Over the last few years, the application of artificial intelligence has become a research topic in the domain of process control. Among them, neural networks and fuzzy logic are used in process control extensively. The tuning method of neural network is based on gradient-guided process. If the derivative of the object function cannot be computed because of discontinuous, these methods often fail [1]. Instead, a stochastic optimization method utilizing evolution strategies, such as Genetic algorithm [2][3], can be applied to find optimal solution reliably. Some reports [4][5] also showed that most of the drawbacks of gradient-based learning can be partially alleviated through the use of GA with proper genetic parameters.

In fact, GA is a general-purpose stochastic optimization method for solving search problems. GA solves the problem of finding good chromosomes by manipulating the material in the chromosomes blindly without any knowledge about the domains being solving. The only information they are given is the fitness of chromosomes. The fitness is used to bias the selection of chromosomes so that those with the best fitness tend to reproduce more often than those with bad evaluations. In addition to selection,

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 284–292, 2013.

crossover and mutation are two important genetic operators. Crossover operator recombines genetic materials of two individuals to create the offspring for the next generation. On the other hand, the mutation operator arbitrarily alters the components of the selected individual so as to increase variability of the population. The degree of change is controlled by the mutation probability. Usually, mutation is useful in restoring lost genetic diversity to a converging population. As long as mutation is active, every point in the search space has a nonzero probability of being generated.

To start GA, several control parameters need to be initialized properly. In [6], it was pointed out that the crossover operator plays a crucial role for solving the premature convergence problem. For example, a larger crossover probability allows exploration of the search space and reduces the chances of getting stuck in local optima; but if this value is too large, it always results in the wastage of computation time in exploring unpromising regions. Therefore, the study of crossover operators is quite necessary. In this paper, we utilize two crossover operators with adaptive probabilities in the genetic search to enhance the performance of GA. The rest of the paper is organized as follows. In section 2, the Adaptive Genetic Algorithms (AGA) [7][8] is introduced briefly. The extension of the AGA combined with arithmetic crossover and BLX-α crossover are also illustrated in section 3. Experimental results and performance evaluations are reported in section 4. Finally, conclusions are drawn in section 5.

2 Adaptive Genetic Algorithms, AGA

In the AGA, the crossover probability p_c and mutation probability p_m are adapted to the fitness values of the population. Given the best fitness value f_{best} and the average fitness value f_{avg} of the population, p_c and p_m are determined as

$$\begin{cases} p_c = k_1\left(1-\dfrac{f'-f_{\text{best}}}{f_{\text{avg}}-f_{\text{best}}}\right), & f' \le f_{\text{avg}} \\ p_c = k_2, & f' > f_{\text{avg}} \end{cases} \tag{1}$$

and

$$\begin{cases} p_m = k_3\dfrac{f-f_{\text{best}}}{f_{\text{avg}}-f_{\text{best}}}, & f' \le f_{\text{avg}} \\ p_m = k_4, & f' > f_{\text{avg}} \end{cases} \tag{2}$$

where f' is the larger of the fitness values of the solutions to be crossed. According to Eqs. (1) and (2), individuals with fitness values higher than f_{avg} will be subject to constant values of p_c and p_m to disrupt its structures and to inherit better genes from others. On the other hand, those one with fitness values lower than f_{avg} will be assigned different degrees of p_c and p_m linearly. The AGA has proved to induce schemata with high fitness values, and also cause the fitness of schemata to increase rapidly [7]. Theoretically, the AGA not only improves the convergence rates, but also prevents the GA from getting stuck at local optima. However, from our experiments rapid convergence is still highly possible. One of the situations is when too many individuals obtain no offspring at all, the solutions become trapped in local optima. Under these circumstances, the search becomes inefficient even p_c and p_m are adapted to the fitness values. The main reason of these phenomena is that the crossover operators cannot guarantee the feasible diversity to find novel regions and thus result in a slowing-down in the search stage. In fact, some tools to monitor genetic process are necessary. In the following section, we present two crossover operators with different characteristics to allow suitable levels of exploration and exploitation to be balanced.

3 Crossover Operators

In this section, two different kinds of crossover operators, e.g. arithmetic crossover [9] and BLX-α crossover [10], are introduced to incorporate with adaptive p_c and p_m in promoting the searching efficiency.

3.1 Arithmetic Crossover

The arithmetic crossover takes the weighted average of two individuals as follows. Assuming that $\mathbf{X}=(x_1,x_2,\ldots,x_n)$ and $\mathbf{Y}=(y_1,y_2,\ldots,y_n)$ are two selected individuals, the resulting off-springs after crossover operator are

$$\begin{aligned} \mathbf{Z}^1 &= \lambda_1\mathbf{X}+\lambda_2\mathbf{Y} \\ \mathbf{Z}^2 &= \lambda_1\mathbf{Y}+\lambda_2\mathbf{X} \end{aligned} \tag{3}$$

Usually, λ_i is randomly and uniformly sampled between [0,1]. When the sum of λ_i is equal to unity, it yields the so called convex crossover [11]. In this sense, the two off-springs are the linear combinations of their parents. If λ_1 and λ_2 are 0.5, the offsprings are the average of two individuals. The crossover operator in this case is called average crossover [12]. If λ_i is simply required to be in real space, it yields the so called linear crossover [13]. The geometric explanation of arithmetic operators in two-dimensional space is shown in Fig. 1.

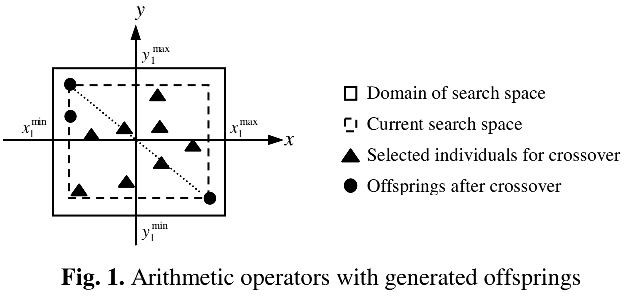


Fig. 1. Arithmetic operators with generated offsprings

The arithmetic crossover has the property in local tuning of the solutions. If the GA can locate the region containing global optima, the population has more information about these zones, then the arithmetic crossover provides efficient exploitation to conduct the offsprings to optimal solutions. In the case of multimodal function, the landscape of the search space is more complex and searching of global points becomes difficult. In particular, when the offspring inherits bad genes from its parents, it can degrade the system performance and may also result in premature convergence. The main reason is due to the improper values of the multipliers λ_i. The problem of premature convergence can be alleviated by creating descendants having the tendency to close to the good individuals. The purpose is to preserve good characteristic of the parents and to speed up convergence. Based on the aforementioned concepts and Eq. (1), we observe that the fitness of the best individuals will have small values of p_c. Therefore, we propose to replace the multipliers λ_i with p_c. Consequently, Eq. (3) is modified as

$$\begin{aligned} \mathbf{Z}^1 &= (1-p_c)\mathbf{X} + p_c\mathbf{Y} \\ \mathbf{Z}^2 &= rand\cdot(1-p_c)\mathbf{X} + p_c\mathbf{Y} \end{aligned} \tag{4}$$

where **X** is the "superior" than **Y** in terms of fitness and *rand* is a random number between [0,1]. In Eq. (4), both $\mathbf{Z}^1$ and $\mathbf{Z}^2$ are between **X** and **Y** but have a higher possibility to situate preferentially in the neighborhood of **X**. As long as the region containing global optimum is found, the suggested method can speed up the local search. On the contrary, when the population diversity is lost as the search proceeds, the BLX-α crossover operator is thus included to investigate new region.

3.2 BLX-α Crossover

When the performance of system degrades, it seems reasonable to imagine the possibility of obtaining good descendants outside the current search region. BLX-α crossover [9] is used to extend the region for exploring the search region in certain extents. Given **X** and **Y**, the offspring with elements z_j^k is randomly chosen in the interval

$$z_j^k \in [\min_j - \alpha \mathrm{R}_j, \max_j + \alpha \mathrm{R}_j], \qquad k = 1,2 \tag{5}$$

where min$_j$=min($x_j^{\min}, y_j^{\min}$), max$_j$=max($x_j^{\max}, y_j^{\max}$), and R$_j$=max$_j$-min$_j$. The typical values of α are chosen between [0,0.5]. The geometric explanation of BLX-α crossover in two-dimensional space is also illustrated in Fig. 2.

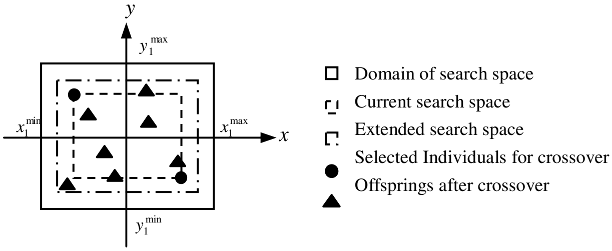


Fig. 2. BLX-α crossover operators with generated offsprings

With the exploration property of BLX-α crossover, the diversity of the population increases and the probability of finding zones with optimal solution also increases. However, too large or too small values of α may also introduce highly uncertainty such that the system diverges or prematurely converges. As α grows, the exploration level is higher and the diversity of population increases. Though, the offspring may be generated far away from their parents, losing their good common properties. If α is small, there are less probabilities to visit new zones in the search space. Therefore, there is no guarantee that the extension caused by BLX-α crossover can conduct the population to the feasible region. In fact, the levels of exploration can be decided by p_c from Eq. (1). If the value of p_c is large, the individual is probably inadequate in the current generation. It makes sense to extend the search region further for exploring suitable descendants. If the value of p_c is small, a local refinement is kept but with a slight extension to increase diversity. In summary, the extension of the search domain is altered proportional to p_c and z_j^k is sampled accordingly in the following interval

$$z_j^k \in \left[\min_j - p_c \mathrm{R}_j, \max_j + p_c \mathrm{R}_j\right], \quad k = 1,2 \tag{6}$$

3.3 Hybrid Crossovers

To decide when to apply arithmetic crossover or BLX-α crossover in the search stage, we observe the system performance in each generation. If the performance is improved in the successive generations, the local search is enhanced with arithmetic crossover to fine-tuning the solutions. On the other hand, BLX-α crossover is employed to probe new regions, which may contain global optimum. As a consequence, a broad search and sufficient refinement are kept in the search stage. It is therefore expected that the efficiency of GA can be promoted and can be verified from our simulations.

4 Experimental Tests

In the section, our goal is intended to solve a given control system involves guiding the states of the control system from given initial states towards target states. The states of the control system are described by a set of state variables. The transition from one state to the next is achieved by a set of control variables. The goal of optimal control is to attain an optimal cost, where cost is measured in terms of the convergence of the states, time, control efforts or some other measures. For a linear discrete system described by

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{A}\mathbf{x}(k)+\mathbf{B}\mathbf{u}(k)\\ \mathbf{y}(k+1) &= \mathbf{C}\mathbf{x}(k)\end{aligned} \tag{7}$$

Our purpose is to determine the control inputs **u**(*k*) that minimize the quadratic cost function described as

$$J(\mathbf{x},\mathbf{u}) = \mathbf{x}^{\mathrm{T}}(N)\mathbf{S}\mathbf{x}(N)+\sum_{k=0}^{N-1}\left(\mathbf{x}^{\mathrm{T}}(k)\mathbf{Q}\mathbf{x}(k)+\mathbf{u}^{\mathrm{T}}(k)\mathbf{R}\mathbf{u}(k)\right) \tag{8}$$

where *N* is the finite state number, **Q**(*k*) is a positive semi-definite matrix, and **R**(*k*) is a positive definite matrix. The weighting matrices are user-specified and define the trade-off between regulation performance and control efforts. The solution of optimal control inputs can be obtained by several different approaches. In the paper, the principle of optimality [14] is employed to obtain the optimal cost. For one-dimensional control system, Eqs. (7) and (8) are simplified as

$$\begin{aligned}x(k+1) &= ax(k)+bu(k)\\ J(\mathbf{x},\mathbf{u}) &= sx^2(N)+\sum_{k=0}^{N-1}\left(sx^2(k)+ru^2(k)\right)\end{aligned} \tag{9}$$

where *a*, *b*, *q*, *s*, and *r* are constants. Based on the parameters as listed in Table 1, the optimal costs *J**(**x**,**u**) for different sets are tabulated as a comparison to the costs obtained by GA.

Table 1. Different parameter sets and the corresponding costs

set	s	r	q	a	b	J^*
0	1	1	1	1	1	$1.618034*10^4$
1	1	1	10	1	1	$1.091608*10^5$
2	1	1	1000	1	1	$1.000999*10^7$
3	1	10	1	1	1	$3.701562*10^4$
4	1	1000	1	1	1	$2.875694*10^5$
5	0	1	1	1	1	$1.618034*10^4$
6	1000	1	1	1	1	$1.618034*10^4$
7	1	1	1	0.01	1	$1.000050*10^4$
8	1	1	1	1	0.01	$4.310041*10^5$
9	1	1	1	1	100	$1.000100*10^4$

In the test of GA, we set the search space for *u*(*k*) in a fixed domain [-200,200] and the population size and number of generations are 30 and 1500, respectively. The corresponding fitness values are calculated by Eq. (9). The results are averaged over 10 trials with different random seeds. In Table 2, simulation results showed that the minimum cost J_2* obtained by GA is very consistent with the theoretic optimal values *J**, in terms of relative error percentage (REP) and the REP is defined as

$$\text{REP} = \frac{J_2^* - J^*}{J^*} * 100\% \tag{10}$$

Table 2. Costs of optimal control compared with GA

set	*J**	J_2*	REP
0	$1.618034*10^4$	$1.618053*10^4$	$1.183*10^{-3}$ %
1	$1.091608*10^5$	$1.091656*10^5$	$4.406*10^{-3}$ %
2	$1.000999*10^7$	$1.001042*10^7$	$4.333*10^{-3}$ %
3	$3.701562*10^4$	$3.701630*10^4$	$1.846*10^{-3}$ %
4	$2.875694*10^5$	$2.875706*10^5$	$4.371*10^{-4}$ %
5	$1.618034*10^4$	$1.618052*10^4$	$1.121*10^{-3}$ %
6	$1.618034*10^4$	$1.625173*10^4$	$4.412*10^{-1}$ %
7	$1.000050*10^4$	$1.000050*10^4$	$8.538*10^{-12}$ %
8	$4.310041*10^5$	$4.310041*10^5$	$1.293*10^{-6}$ %
9	$1.000100*10^4$	$1.006975*10^4$	$6.874*10^{-1}$ %

From the results of Table 2, the costs of optimal control and those obtained by GA are very consistent except for sets 6 and 9. To verify the effects of hybrid crossovers, GA with arithmetic crossover only is simulated to compare the performances. The experimental evidences in Table 3 showed the fact that the GA with hybrid crossovers outperforms the GA with arithmetic crossover only.

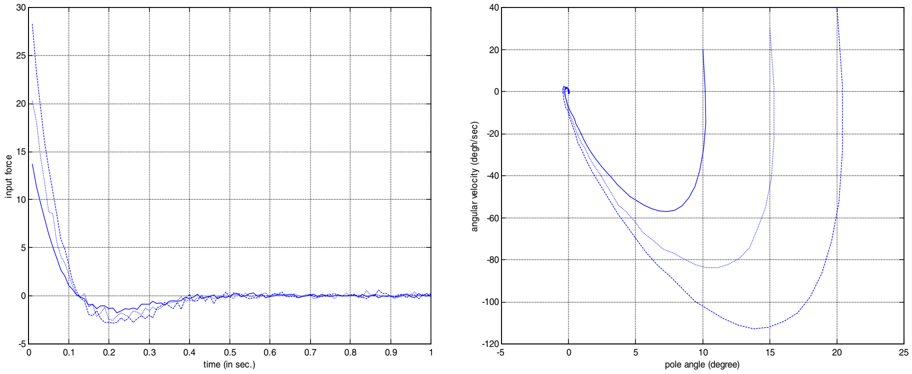
Table 3. Simulation results of GA with arithmetic crossover and hybrid crossovers

set	**GA+hybrid crossover**	**GA+arithmetic crossover**
0	$1.183*10^{-3}$ %	$6.918*10^{-1}$ %
1	$4.406*10^{-3}$ %	6.020 %
2	$4.333*10^{-3}$ %	10.07 %
3	$1.846*10^{-3}$ %	$4.378*10^{-2}$ %
4	$4.371*10^{-4}$ %	$1.291*10^{-1}$ %
5	$1.121*10^{-3}$ %	$7.567*10^{-1}$ %
6	$4.412*10^{-1}$ %	8.305 %
7	$8.538*10^{-12}$ %	$5.196*10^{-5}$ %
8	$1.293*10^{-6}$ %	$1.652*10^{-3}$ %
9	$6.874*10^{-1}$ %	73.67 %

In our second simulations, the goal is to apply hybrid crossovers in controlling the inverted pendulum problem [15]. The inverted pendulum problem involves controlling an inherently unstable system of a cart and a pole that is constrained to move within a vertical plane. The cart can move to the right or left on rails when a force is exerted on it. The state information includes the angle θ of the pole and the angular velocity $\dot{\theta}$ of the pole. The objective is to keep the pole balance, e.g. $(\theta, \dot{\theta})$=(0,0). The error measure used is

$$E=\sum_{k=1}^{N}\theta^2(\mathrm{T}_k)+\sum_{k=0}^{N-1}u^2(\mathrm{T}_k) \tag{11}$$

In Eq. (11), $u(\mathrm{T}_k)$ is the GA's output force. The sampling period T used is 10 ms and N=100. Therefore, the control actions are from t=0 to t=1 second. The initial conditions of the inverted pendulum problem $(\theta,\dot{\theta})$ are (10,20), (15,30), and (20,40), respectively. The length of the pole is 0.5. The population size and number of generations are 30 and 8000, respectively. In Fig. 3(a), solid, dashed, and dotted curves correspond to the control actions for three initial conditions. The state-space plot in Fig. 3(b) shows how the trajectory approaches the origin from three initial points (10,20), (15,30), and (20,40). Fig. 3(a) and Fig. 3(b) reveal the robustness and fault tolerance of our proposed methods with different initial settings.



(a) (b)

Fig. 3. (a) The control actions for inverted pendulum system (solid, dashed, and dotted curves correspond to the initial conditions (10,20), (15,30), and (20,40)). (b) The pole behaviors start from different initial conditions.

5 Conclusions

In the paper, GA combined with hybrid crossovers operators is presented to promote the efficiency of GA. The arithmetic crossover provides efficient exploitation in local improvement of the solutions. As the search proceeds, the population diversity may lose. Then, the BLX-α crossover is applied to extend the current search region properly for exploring suitable descendants. In fact, the complementary effects of arithmetic and BLX-α crossovers allow to maintain population diversity and to sustain local improvement in the search stage. Simulation results in the given control system show that the hybrid approaches outperform the conventional GA or GA with arithmetic crossover only.

Acknowledgments. This research was supported by the National Science Council under contract numbers NSC99-2221-E-212-023 and NSC100-2221-E-212-006.

References

1. Kim, J., Moon, Y., Zeigler, B.P.: Designing fuzzy net controllers using genetic algorithms. IEEE Control Systems 15(3), 66–72 (1995)
2. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Publishing Company Inc. (1989)
3. Davis, L.: Handbook of Genetic Algorithms. Van Nostrand Reinhold, N.Y. (1991)
4. Yoon, B., Holmes, D.J., Langholz, G., Kandel, A.: Efficient genetic algorithms for training layered feedforward neural networks. Information Science 76(1), 67–85 (1994)
5. Zhou, C., Ye, J., Zhu, S.: Fuzzy modeling using fuzzy neural networks with genetic algorithms. In: Proceeding International Conference. on Information and Knowledge Engineering, China, pp. 114–118 (1995)
6. Booker, L.: Improving search in genetic algorithms. In: Davis, L. (ed.) Genetic Algorithms and Simulated Annealing, pp. 61–73. Morgan Kaufmann Publishers, Los Altos (1987)
7. Srinivas, M., Patnaik, L.M.: Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Trans. Systems, Man, and Cybernetics 24(4), 17–26 (1994)
8. Srinivas, M., Patnaik, L.M.: Genetic search:analysis using fitness moments. IEEE Trans. on. Knowledge and Data Engineering 8(1), 120–133 (1996)
9. Herrera, F., Lozano, M., Verdegay, J.L.: Tackling real-coded genetic Algorithms: operators and tools for behavior analysis. Dept. of Compt. Science and Artificial Intelligence, Technical Report DECSAI-95107 (1995)
10. Eshelman, L.J., Schaffer, J.D.: Real-coded genetic algorithms and interval schemata. In: Whitley, L.D. (ed.) Foundation of Genetic Algorithms-2. Morgan Kaufmann Publishers (1993)
11. Michalewicz, Z.: Genetic Algorithm+Data Structure=Evolution Programs. AI Series. Springer, N.Y. (1994)
12. Schwefel, H.: Numerical Optimization of Computer Models. John Wiley & Sons, Chichester (1981)
13. Cheng, R., Gen, M.: Genetic Algorithms and Engineering Design. John Wiley & Sons, N.Y. (1997)
14. Bellman, R.: Adaptive Control Process: A Guided Tour. Princeton University Press, N.J. (1961)
15. Kwakernaak, H., Sivan, R.: Linear optimal control systems. Wiley-Interscience, N.Y. (1972)

An Intelligent Fusion Algorithm for Uncertain Information Processing

Peiyi Zhu, Benlian Xu, and Mingli Lu

School of Electrical and Automation Engineering,
Changshu Institute of technology,
Hushan Road, Changshu, Jiangsu
{China,zpy2000}@126.com

Abstract. With the development of various advanced sensors, and some sensing technologies are not mature, so that measurement information was being uncertain, incomplete. This paper adopts an intelligent fusion algorithm with Rough Set for reduction of the attribute set and target set for the raw data from various sensors. Consequently the noise and redundancy will be reduced in sampling. Then constructs information prediction system of SVM according to the preprocessing information structure, and solves the problem of multisensor data fusion in the situation of small sample and uncertainty. In order to get the optimal fusion accuracy, it uses PSO for fusion parameters. To make operation faster and increase the accuracy of the fusion, a feature selection process with PSO is used in this paper to optimize the fusion accuracy by its superiority of optimal search ability.

Keywords: Uncertain information, Data fusion, RS, SVM.

1 Introduction

Accurately get farmland soil attribute information and its precise geographic distribution information that can placement launch agricultural production materials, and to realize the reasonable utilization of cultivated land resource and the modernization of agriculture precise management. But the traditional laboratory test is typically time consuming and laborious. Hence, rapid measurement and monitoring of apparent soil electrical conductivity (EC_a) is needed to satisfy the precision farming requirements [1, 2]. The existing problem of multi-sensor data fusion and mapping of soil properties with EC_a is increasingly become important in the precision farming requirements, due to its successful implementation, increasing overall reliability [3].

Recently, with the development of artificial intelligent technology, it is possible that this paper puts forward a new algorithm to predict more accurate EC_a [4]. For the EC_a are affected by soil conductivity maximum value, standard deviation, latitude, longitude etc, Rough Sets theory (RS) was adopted for the raw sensory data attribute reduction and target reduction, thus to reduce the sampling noise and redundancy. Meanwhile, using support vector machine (SVM) to fuse the data after reduction that the small sample and uncertain conditions data fusion problem were solved. In order

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 293–300, 2013.

to improve the accuracy of fusion, a feature selection process using Particle Swarm Optimization (PSO) is used in this paper to optimize the fusion accuracy by its the superiority of optimal search ability. Simulation results show that the proposed method has fine fault-tolerance, accuracy and robustness.

2 Data Fusion Based on RS

Rough set (RS) theory was developed by Pawlak in 1982. It is a mathematical tool for dealing with imprecise, uncertain and vague information. It has been applied successfully in such fields such as machine learning, data mining, pattern recognition, fault diagnosis, etc [5]. Since RS theory can analyze and cause uncertainty data, and find the internal data relationship, the theory has an ability of extract feature information and simplifies the information, so it is suitable for processing the problem of multi-sensor data fusion. Data fusion based on RS is commonly used distributed structures, which regard the sensor data collection as an information table. We propose a kind of rough set attribute reduction algorithm to get the minimized attributes (Fig. 1), that is said, analysis the sensor data collection, and get rid of the uncorrelated attributes and from the fusion results.

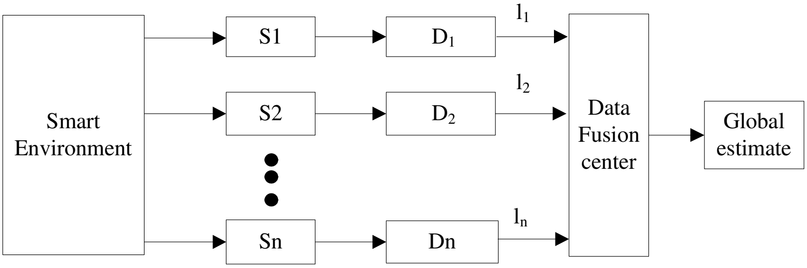


Fig. 1. Distributed data fusion structure

In this structure, $S_1, S_2, ..., S_n$ are sensors, $D_1, D_2, ..., D_n$ are local processors, and $l_1, l_2, ..., l_n$ are local decisions which are decided by the different character of the detected object. In the fusion system, the knowledge of fusion center is expressed by information table in generally. Information table is consisted of object, condition attribute and decision attribute in universe of discourse, where the universe of discourse is measurement information sets of all sensors; condition attribute $C = \{a_1, a_2, ..., a_n\}$ is local decisions sets, that is, local decision result; decision attribute $D = \{d\}$ is final decisions sets from fusion center. There are many attribute reduction algorithms available now, in this work, we adopt many algorithms of attribute reduction to process the dataset, and the reduced attribute subset got by the conditional entropy-based algorithm for reduction of knowledge without core is very reasonable.

3 Data Fusion Based on RS-SVM

Support Vector Machine (SVM) is a class of supervised learning algorithms, as introduced by Vapnik in the late 1960 that analyze data and recognize patterns, used for classification and regression analysis. In addition to performing linear classification, SVM can efficiently perform non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces [6]. This paper will introduce PSO for parameter selection of SVM in order to improve the classification performance of SVM, and then combine RS theory for data fusion.

3.1 Optimal Feature Selection in SVM Based on PSO

It is well known that SVM generalization performance (estimation accuracy) depends on a good setting of hyper parameters C, the kernel function and the kernel parameter. Traditionally, feature selection is performed by experience or trail and error, which might result in a loss of information relevant to classification tasks [7]. Extending previous work on feature selection and classification, this paper proposes PSO for optimal feature selection of SVM. Since the original version of PSO proposed in 1995[8], lots of work have been done to develop it.

SVM based on PSO extracts the input feature subset selection and the SVM kernel parameter setting to optimize SVM classification accuracy. If it is no feature to extract that just estimates the two decision variable C and γ. To feature selection, we need to decide which is selected from the N feature system, and we need N+2 decision variable, where N feature corresponds to decision variable in the range of [0, 1]. If the variable less than or equal to 0.5 that the corresponding feature will be selected, vice versa.

PSO starts with $\boldsymbol{n}$-randomly selected particles and searches for the optimal particle iteratively. Each particle is an $\boldsymbol{m}$-dimensional vector and represents a candidate solution. SVM classifier is built for each candidate solution to evaluate its performance through the cross validation method. PSO algorithm guides the selection of potential subsets that lead to the best prediction accuracy. The algorithm uses the fit particles to contribute to the next generation of $\boldsymbol{n}$-candidate particles. Thus, on average, each successive population of candidate particles fits better than its predecessor. PSO is used to find optimal feature subsets by discovering the best feature combinations as they fly within the problem space from the processed datasets. The procedure describing proposed PSO-SVM approach is shown in Figure 2.

3.2 RS-SVM for Data Fusion

Decision-level fusion is very important in the three levels of image fusion (pixel-level fusion, feature-level fusion, and decision-level fusion), because it aims at the concrete policy-making goal, directly, its fusion results provide evidence for controlling, commanding and decision-making, that is to say, decision-level fusion results influence the level of decision. It is because of the extreme importance of decision-level fusion, the various existing data fusion methods cannot satisfy its requirements.

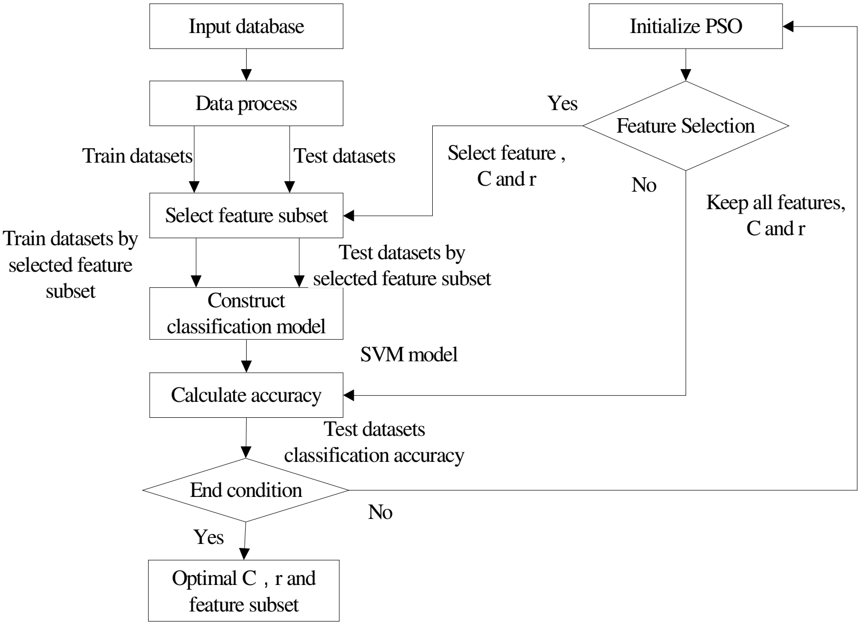

Input database
Data process
Train datasets
Test datasets
Select feature subset
Train datasets by selected feature subset
Test datasets by selected feature subset
Construct classification model
SVM model
Calculate accuracy
Test datasets classification accuracy
End condition
Yes
No
Optimal C , r and feature subset
Initialize PSO
Feature Selection
Yes
Select feature , C and r
No
Keep all features, C and r

Fig. 2. The procedure describing proposed PSO-SVM approach

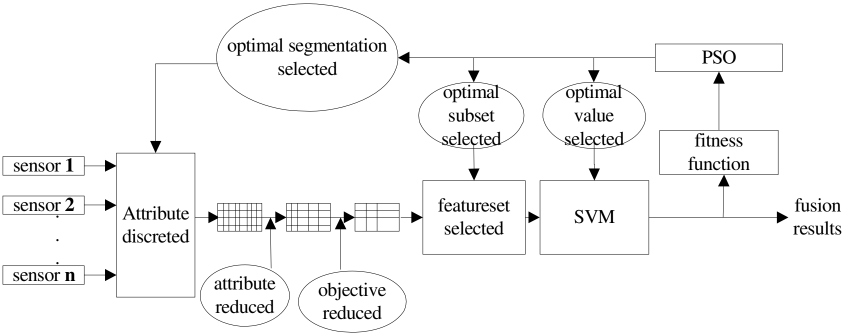


Fig. 3. Multisource Data Fusion Based on Intelligent Algorithm

Aiming at the existing problem of data fusion, a method which simulates human thinking to realize data fusion is presented in this paper (Figure 3).

In this method, multi-source input information is reduced firstly by using the strong qualitative analysis ability of rough set theory, and the noise and redundancy in the samples are removed. On the basis of it, using support vector machine, reduced information is fused. In order to obtain optimum fusing accuracy, particle swarm optimization algorithm is used to optimize fusion parameters.

4 Experiences and Results

In the paper, the study was carried out on a corn field (35°57'29"–37°57'57" N, 118°84'21"–118°85'00" E) located in northeastern Kenli Country in Shandong, eastern coast of China. The soil data in the experimental plot were collected using a regular sampling grid of 15m from an EM38 sensor measurement over 20 days (5 times per day), and get 100 training sample[6]. EC_a is a measurement that correlates with maximum value of EC_a, standard deviation, locating latitude, the longitude, soil temperature, moisture content, pH value, depth, and soil types such as yellow brown soil, paddy soil, tide soil and yellow spongy soils, and some human factors such as measurement error and management decision, which is 14 dimension input variable, that is training sample $l = 100$, and every sample dimension $N = 14$. Firstly these samples are normalized, then construct SVM prediction model, and according to the training sample to establish the optimal model are as follows:

$$\begin{cases} \min\limits_{a,a^*} : -\frac{1}{2}\sum\limits_{i,j=1}^{l}(a_i - a^*_i)(a_j - a_j^*)K(x_i \cdot x_j) - \varepsilon\sum\limits_{i=1}^{l}(a_i - a_i^*) + \sum\limits_{i=1}^{l} y_i(a_i - a_i^*) \\ s.t. \quad 0 \le a_i, a_i^* \le C \\ \qquad \sum\limits_{i=1}^{l}(a_i^* - a_i) = 0 \end{cases} \tag{1}$$

Where $l = 100$, $x_i (i = 1,2,...,l)$ is ith training sample input, $y_i (i = 1,2,...,l)$ is training sample output. RBF is selected as kernel function, $K(x_i, x_j) = \exp(-\frac{|x_i - x_j|^2}{2\gamma^2})$, where $C = 1$, $\gamma^2 = 1$. Minimized equation (1) and solved a_i, a_i^* by PSO, then got the optimal solution as follows:

$$(a, a^*) = (a_1, a_1^*, ..., a_l, a_l^*)^T, \quad \min F(a, a^*) = -0.82561 \tag{2}$$

where $(a_1, a_1^*) = (0.5248,0)$; $(a_6, a_6^*) = (0,0.6027)$; $(a_{11}, a_{11}^*) = (0,0.3027)$; $(a_{18}, a_{18}^*) = (1,0)$ $(a_{22}, a_{22}^*) = (1,0)$; $(a_{35}, a_{35}^*) = (0.3268,0)$; $(a_{40}, a_{40}^*) = (0,0.4095)$; $(a_{47}, a_{47}^*) = (0.1472,0)$: $(a_{51}, a_{51}^*) = (0,1)$; $(a_{65}, a_{65}^*) = (0,1)$; $(a_{66}, a_{66}^*) = (0.6529,0)$; $(a_{72}, a_{72}^*) = (0.2318,0)$; $(a_{84}, a_{84}^*) = (1,0)$; $(a_{86}, a_{86}^*) = (0.2508,0)$; $(a_{87}, a_{87}^*) = (0,0.8761)$; $(a_{95}, a_{95}^*) = (0.2945,0)$; $(a_{98}, a_{98}^*) = (0.3406,0)$; the rest of (a, a^*) equal to zero. The support vectors number is 17, in other words, the first, 6th, 11th, 18th, 22 th, 35 th, 40 th, 47th, 51 th, 65 th, 66 th, 72 th, 84 th, 86 th, 87 th, 95th, and 98 th input training sample are support vectors, and the corresponding Lagrange multiplier $(a_i, a_i^*) \ne (0,0)$. The basic support vector is 18th, 22 th, 51 th, 65 th, 84 th (Lagrange multiplier a_i or $a_i^* = C = 1$). The other (a, a^*) are not support vectors and do not participate in establishing decision-making regression equation. From Eq. 1& Eq.2, we have:

$$\begin{aligned} f(x) = {} & 0.5248\exp(-\left|x-x_1^*\right|^2/2) - 0.6027\exp(-\left|x-x_6^*\right|^2/2) - 0.3027\exp(-\left|x-x_{11}^*\right|^2/2) + \\ & \exp(-\left|x-x_{18}^*\right|^2/2) + \exp(-\left|x-x_{22}^*\right|^2/2) + 0.3268\exp(-\left|x-x_{35}^*\right|^2/2) - \\ & 0.4095\exp(-\left|x-x_{40}^*\right|^2/2) + 0.1472\exp(-\left|x-x_{47}^*\right|^2/2) - \exp(-\left|x-x_{51}^*\right|^2/2) - \\ & \exp(-\left|x-x_{65}^*\right|^2/2) + 0.6529\exp(-\left|x-x_{66}^*\right|^2/2) + 0.2318\exp(-\left|x-x_{72}^*\right|^2/2) + \\ & \exp(-\left|x-x_{84}^*\right|^2/2) + 0.2508\exp(-\left|x-x_{86}^*\right|^2/2) - 0.8761\exp(-\left|x-x_{87}^*\right|^2/2) + \\ & 0.2954\exp(-\left|x-x_{95}^*\right|^2/2) + 0.3406\exp(-\left|x-x_{98}^*\right|^2/2) \end{aligned} \tag{3}$$

We can take predicted sample input x to decision regression equation $f(x)$, and then obtain the predicted value. Using De-normalization processing, we can get the real EC$_a$ value. In Figure 4 SVM predicted EC$_a$ value in 20days.

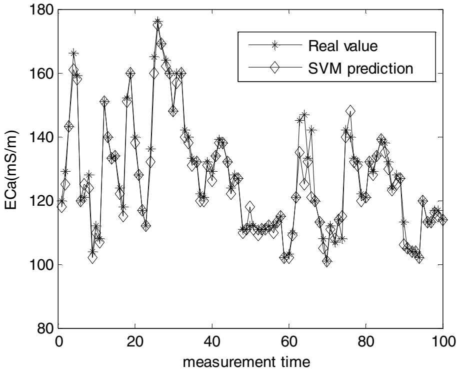


Fig. 4. The prediction of EC$_a$ in 20 days (x in days)

However, the simple SVM cannot fully reflect the nonlinear characteristics of all samples when the training samples are various, which reduces the performance of SVM. So we can reduce the EC$_a$ prediction factors by RS theory, and select the input samples which have the same character with EC$_a$ prediction value to construct a prediction model based on SVM. EC$_a$ value correlates with 14 factors which have relations with EC$_a$. We can describe these data as an information sheet $L=<U,R,v,f>$ by RS theory, where U is influencing factor set, C & D is condition and decision attributes, respectively ($C, D \subseteq R$). Firstly, discretizing this 100 sample set, we can construct information sheet T_0 by historical data which contains a characteristic parameter. Next, build up the new information sheet T_1 by the minimum attribute. If the condition attributes is C_2 at this time, D_2 is decision attributes, thus $C_2 \subseteq C_1$, $D_2 \subseteq D_1$. After reducing attributes based on RS, minimum attribute includes: maximum value of EC$_a$, standard deviation, locating latitude, the longitude, soil temperature, moisture content, pH value and depth. The number of attribute was reduced from 1400 to 800, the rest all deleted. Thus we have define this reduced attribute as new learning sample set of SVM, according to the minimum condition attribute set and the corresponding original data we can get test data again. After optimizing C and γ^2 in SVM by PSO, we learn and train SVM until meet the termination conditions. We can get the final prediction results in Figure 5.

As can be seen from figure 5, the performance of RS-SVM is better than the performance of normal SVM while using only 57% of the features. One of the main reasons we deleted some redundancy attribute after RS attribute reduction, which did not lose any effective information. And PSO-SVM can adjust penalty coefficient and width coefficient as adaptable, which can effectively avoid the problems of "owe learning" and "over learning" in normal SVM, so as to greatly reduce the data dimension, reduced the complexity of the classification process. The results are summarized in Tab. 1 and Tab. 2.

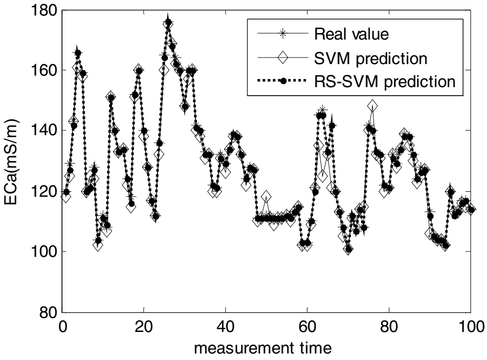


Fig. 5. The prediction of EC_a with two different algorithms(x in days)

Table 1. The mean-square deviation of predictions

Test point	training set		testing set	
	SVM	Our method	SVM	Our method
1	0.048	0.035	0.043	0.031
2	0.059	0.046	0.042	0.025
3	0.047	0.041	0.032	0.015
4	0.034	0.038	0.043	0.016
5	0.072	0.049	0.038	0.033
6	0.038	0.025	0.047	0.036

Table 1 show that our method achieves higher accuracy rates compared with the normal SVM. However, the normal SVM cannot simplify input information space dimension, so the sample training time will be added when the input sample set dimension is high. Furthermore, after RS attribute reduction we deleted some redundancy attribute, so our method had a faster prediction speed which was shown in Table2.

Table 2. The time of simulation

Test point	simulation time(the units: S)	
	SVM	Our method
1	25.0	6.0
2	28.1	5.8
3	19.2	4.5
4	17.3	3.9
5	22.7	4.8
6	26.5	5.1

5 Conclusions

In this paper, we have presented an intelligent method for uncertain information processing. Moreover, we related the proposed framework with RS-SVM and provided for its use as a data fusion method. Experiments showed that our method had a well the maximum MSE value and the average, and which show that the proposed method has fine fault-tolerance, robustness and accuracy.

Acknowledgments. It is a project supported by a grant from the natural science fundamental research program of higher education colleges in Jiangsu province (07KJB510001), Suzhou Municipal Science and Technology Plan Project No.SYN201109.

References

1. Corwin, D.L., Lesch, S.M.: Application of soil electrical conductivity to precision agriculture: theory, principles, andguidelines. Agron.J 95(3), 455–471 (2003)
2. Corwin, D.L., Lesch, S.M.: Characterizing soil spatial variability with apparent soil electrical conductivity: I Survey protocols. Comp. Electron. Agric. 46, 103–133 (2005)
3. Corwin, D.L., Lesch, S.M.: Apparent soil electrical conductivit measurements in agriculture. Computers and Electronics in Agriculture 46, 11–43 (2005)
4. Zhu, P., Xiong, W., Qin, N., Xu, B.: D-S Theory Based on an Improved PSO for Data Fusion. Journal of Networks 7(2), 270–276 (2012)
5. Pawlak, Z., et al.: Rough sets. Communications of the ACM 38(11), 88–95 (1995)
6. Zhu, P., Xu, B.: Fusion of ECa Data using SVM and Rough Sets Theory Augmented by PSO. Journal of Computational Information Systems 7(1), 295–302 (2011)
7. Zhao, W.-Q., Zhu, Y.-L., Jiang, B.: A classification model based on SVM and rough set theory. Journal of Communication and Computer 39(5), 42–45 (2008)
8. Zhu, P., Xiong, W., Xu, B.: A Sensor Management Method Based on an Improved PSO Algorithm. International Journal of Advancements in Computing Technology 4(9), 259–265 (2012)

A New Target Tracking Algorithm Based on Online Adaboost

Zhuowen Lv, Kejun Wang, and Tao Yan

College of Automation, Harbin Engineering University, 150001Harbin, China
wangkejun@hrbeu.edu.cn

Abstract. In order to overcome the effect of blocking in process of target tracking under stationary camera, a target tracking algorithm based on online Adaboost was presented. Codebook model was set up to detect moving target in YUV color space; in process of tracking, feature of online Adaboost fused texture contours and color, then accurate target location was obtained. The experimental results show that, the detecting algorithm in this paper has good detecting results, which provides assistance to tracking. The proposed tracking algorithm is effective for the targets having blocking, even a large area of blocking in more complex scenes.

Keywords: moving target tracking, codebook model, online Adaboost, feature selection.

1 Introduction

In the field of computer vision, The real-time detection and track of targets is an important research topic[1]. In recent years, statistical learning method gradually becomes one of the mainstream technology in the field of pattern recognition, and it has successfully been applied in many classic issues, for example target tracking. Adaboost algorithm[2] based on Haar feature is a successful application in face detection[3]. [4] applied Adaboost algorithm to the field of target tracking, the linear combination of R, G and B with integer coeffcients was used to generate the candidate features, and the result of this method had achieved better tracking results. Unlike offline Adaboost algorithm[2], the training samples of online Adaboost algorithm are several real-time data. Using this algorithm can better adapt to the changes of target features, however, large areas of blocking easily lead to classification errors in complex scenes[5], then the tracking target will be lost.

This paper proposed a target tracking algorithm based on online Adaboost. In the process of detecting, codebook model was set up to detect target in YUV color space [6], the shadow problems has been solved; In the process of tracking, the problem of blocking has been solved. The feature of online Adaboost algorithm fused texture contours and color. With the change of the pixels in each frame tracking, the classifications were updated and eliminated to reflect the real-time new features.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 301–307, 2013.

2 Target Detection Based on Codebook

Kim[7] set up the codebook background model by using quantification and clustering techniques. Each pixel has a codebook which includes several code words in RGB space. The algorithm is made up of training and testing. During the training phase, fixed number of frames for each pixel are trained firstly, and cluster the pixels according to intervals which pixels fall into; learn and update according to the value and time information of pixels. After training, code words are excluded according to the hit numbers, invalid code words generated from interference factors are eliminated. During the testing phase, if the pixel value falls into the allowable range of certain code word, test result is background, else is foreground.

Codebook model of RGB space[7] retaines multimode characteristics of background, memory occupancy is small and computing complexity is very low. Most original videos from video capturing equipment are based on YUV color space, if the motion detection is done in RGB or HSV space, it is necessary to converse RGB or HSV space to YUV space by using floating-point mathematical operation[8], this amount of computation is large； what is more, YUV space or HSV space is fit for the characteristics of separate luminance and chrominance in codebook algorithm, shadow illumination model has larger changes only for the luminance components, according to this point, the two color spaces can better solve the detection shadow problem. S (saturation value) of HSV color space corresponds white, when the scene contains lots of white information, background is easily mistaken for the foreground. The paper build codebook model in the YUV color space [6].

3 Tracking Process of Online Adaboost Algorithm

3.1 Feature Selection

Good features and good tracking algorithms are equally important[9], features directly determine the accuracy of weak classifiers. Good features not only reduce the number of learning samples, but also reduce the number of weak classifiers[10]. The features in this paper select color (RGB) and local direction histogram (LOH). LOH feature is not sensitive to light and expresses contours indirectly. Generally, when the colors of the targets are similar to the background, their textures are not similar. LOH features:

$$E(R) = \sum_{(x,y)\in R} \psi_K(x,y) \cdot \tag{1}$$

R is the processing region. $K = 8$. In order to adapt feature changes during tracking accurately and comprehensively, three color channels of RGB are added to LOH feature vector. There is $K+3$ dimensional features for every pixel. These feature values are substituted into weak classifiers, which are not only meeting the few samples training, but also improve detection accuracy.

3.2 Initializing the Weak Classifiers

The proposed tracking algorithm makes the first frame of the targets detected by codebook as the starting tracking frame. After detecting targets by using codebook, the weak classifiers of online Adaboost algorithm must be firstly initialized during tracking process.

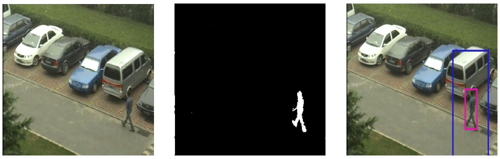


Fig. 1. Target position initialization, from left to right are respectively (a) a frame of a video, (b) the binary image of the detecting result and (c) the process of the proposed tracking algorithm

Fig. 1, the result of target detection is pink rectangle, making this rectangle as the center to construct interested region (blue rectangle, length and width are three times larger than the pink rectangle), the pixels are marked $y_i = +1$, which are positive samples in the pink rectangle region; the pixels are marked $y_i = -1$, which are negative samples in the other area of blue rectangle. On the basis of this, initialize the weak classifiers:

Input: a frame of detected target image.

1) According to detection results， mark the N pixels $\{x_i\}_{i=1}^{N}$ of the interested region;
2) initialize the weights of samples: $w_1 = w_2 = \ldots = w_N = 1/N$;
3) for the M weak classifiers, cycle the following steps:
   a) Calculate $K+3$ dimensional eigenvectors of LOH and RGB for each pixel, K is feature dimension of LOH;
   b) Train the weak classifiers h_t ($t = 1, 2, \ldots, M$);
   c) Calculate the error rate: $err = \sum_{i=1}^{N} w_i \left| h_t(x_i) - y_i \right|$;
   d) Calculate the weights of classifier: $\alpha_t = \frac{1}{2} \ln(1 - err/err)$;
   e) Update and normalize the weights of sample points: $w_i = w_i \cdot e^{(\alpha_t |h_t(x_i) - y_i|)}$, $\sum w_i = 1$;
4) Strong classifier: $H(x) = sign\left(\sum_{t=1}^{M} \alpha_t \cdot h_t(x)\right)$.

The number of weak classifiers is $M = K + 3 = 11$, we use least square method to train the weak classifiers $h_t(x) : x_i \to \{-1, 1\}$, then update the weights $\{w_i\}_{i=1,\ldots,N}$ of the sample points. Cycle M times like this, we can complete the initialization of the weak classifiers.

3.3 Tracking Process

The distance between adjacent frame of targets will not be too far, in the new frame, the interested region is the same as the former frame. In order to adapt to the target appearance changes caused by light, angle, occlusion, weak classifiers of the former frame need to be updated. According to the error rate of weak classifiers in each frame, the weak classifiers of larger error rates should be eliminated, weights of the $M-k$ retained weak classifiers should be updated and calculate error rate. The process of tracking algorithm:

Input: a new frame I_j and initialized weak classifiers $\{h_1,h_2,\dots h_M\}$

1) Extract samples $\{x_i\}_{i=1}^N$ of the interested region, give a group of initialized weights $\{w_i\}_{i=1}^N$; mark the interested region as $y_i=\{-1,+1\}$;
2) Remove k classifiers of high error rate from the original weak classifiers $\{h_1,h_2,\dots h_M\}$; for the rest of weak classifiers $(l=k+1,\cdots,M)$:
   a) Calculate error rate, select weak classifier h_t of the lowest error rate from the remaining weak classifiers;
   b) Update weights α_t of weak classifiers and $\{w_i\}_{i=1}^N$ of sample points according to the new features;
   c) Add h_t to the new classifier group, for the rest of the classifiers, cycle from a);
3) Add k new classifiers:
   a) Select weak classifier h_t ($t=1,2,\dots,M$) for sample points of the interested region;
   b) Calculate the error rate err and weights α_t ;
   c) Update the weights $\{w_i\}_{i=1}^N$ of sample points•and ensure $\sum w_i=1$;
   d) Get strong classifier $H(x)$ according to the trained M weak classifiers.

The process of tracking is combining online Adaboost initialization of Weak classifiers with the above tracking steps. The removed classifiers of high error rate is $k=3$.

4 Process of Algorithm and Analysis of Experimental Results

4.1 Process of Detection and Tracking Algorithm

1) Train background model and obtain codebook background model;
2) Detect whether input pixels match codebook models and obtain foreground target;
3) According to the detecting results, the targets are tracked by online Adaboost algorithm in the interested region.
4) Turn to 2. Begin the next round of tracking.

4.2 Analysis of the Experimental Results

The first experimental video in this paper is from PETS (IEEE International Workshop on Performance Evaluation of Tracking and Surveillance) 2001 Dataset 1 Camera 1 Image, which is a typical video sequence having global and local changes of illumination. What is more, there are processes of targets becoming into background. The second experimental video is from Video Library of Chinese Academy of Sciences[11].Two videos have 2689 frames and 253 frames. All experimental test platform is CPU of Intel (Pentium) dual-core 3.17GHz, 2GB RAM, Windows XP operating system, the test codes are compiled and run on VS2008 by using C++ and OpenCv.

The results of codebook detection algorithm are shown as Fig. 2, in which the car is from moving state to stationary state (Fig. 2(b)). The car becomes background gradually (Fig. 2(c), (d)). The proposed algorithm achieved better detection effect.

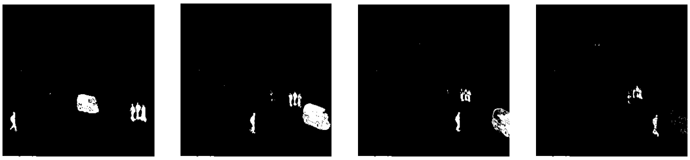


Fig. 2. Detection result of scene 1, from left to right are respectively the frame of (a) 839th, (b) 944th, (c) 997th, (d) 1041th.

Fig. 3 compare codebook detection algorithm with the other algorithms. Fig. 3(b) has poor ability of processing shadows, and the detected foreground region is much larger than the actual; though Fig. 3(c) restrains parts of shadows, it does not detects all the foreground targets, the small amounts of targets detected are also incomplete. Fig. 3(d) has poor ability of restraining noises, and the foreground targets detected have large areas of voids. The codebook detection algorithm can restrain most noises, shadows and obtain more complete foreground targets (Fig. 3(e)).

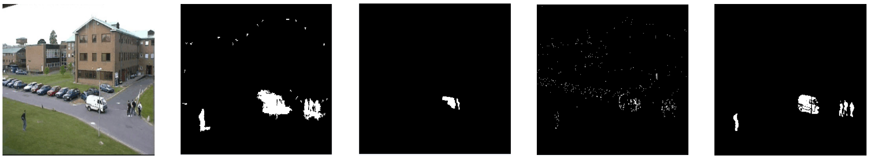


Fig. 3. Detection result of different algorithms, from left to right are respectively (a) a frame of scene 1, (b) detection result of mixture Gaussian model, (c) detection result of Bayesian decision, (d) detection result of three frame subtraction, (e)detection result of codebook.

Fig. 4, Fig. 5 and Fig. 6 are tracking results of CamShift algorithm, particle filter and the proposed algorithm, where the target has blocking and even large areas of blocking in the case of having other interference target. The experiment makes the first target entering into the monitoring area as the tracking target. In order to show the process of tracking clearly, the blue rectangle (interested region) is removed, only the rectangle of target position is left.

The tracking result of CamShift algorithm is not good, the target is blocked by interference target for a long time, tracking rectangle includes interference target (Fig. 4(b), (c)), after the 364th frames, tracking fails (Fig. 4 (d)).

Particle filter predicts the target state by using the random distribution of many particles. When blocking happens, only a few particles which have large weights can be distributed in unconcluded parts. However, the particles of blocking parts have small weights, the estimation of the target state is mainly decided by the particles with larger weights, particles appear degradation phenomenon, which leads to the tracking deviation eventually. Particle filter algorithm can not continuously track the target when blocking happens, instantaneous or intermittent frames will lose target (Fig.5 (b)). After the 439th frames, the target moves fast, the tracking fails (Fig.5 (d)).

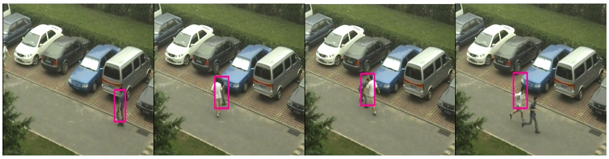


Fig. 4. Tracking result of the target having large blocking area by CamShift algorithm, from left to right are respectively the frame of (a) 80th, (b) 311th, (c) 362th, (d) 452th

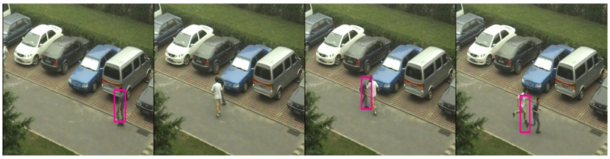


Fig. 5. Tracking result of the target having large blocking area by particle filter algorithm, from left to right are respectively the frame of (a) 80th, (b) 311th, (c) 362th, (d) 452th

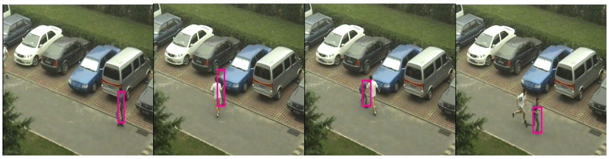


Fig. 6. Tracking result of the target having large blocking area by proposed algorithm, from left to right are respectively the frame of (a) 80th, (b) 311th, (c) 362th, (d)452th

The proposed online Adaboost algorithm fuses RGB and LOH features, which makes the target not interfered by the color and avoids the effect of blocking (Fig.6 (a), (b), (c), (d)). This experiment shows that the suitability of the proposed algorithm in the condition of blocking, even larger area of blocking.

5 Conclusion and Future Works

This paper introduces codebook detection algorithm in YUV space and online Adaboost tracking algorithm, the combination of the two methods has the following advantages: low computational complexity; most of the noise and shadows are suppressed, complete targets are extracted; large areas of blocking are handled. In the future, we will try to use offline Adaboost algorithm before using online Adaboost algorithm to track, which can also increase the tracking accuracy in theory.

References

1. Hou, Z.Q., Han, C.Z.: A Survey of Visual Tracking. Acta Automatica Sinica 32, 603–617 (2006)
2. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: 13th International Conference on Machine Learning, pp. 148–156. American Association for Artificial Intelligence, New York (1996)
3. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 511–518. IEEE Press, Los Alamitos (2001)
4. Jia, J.P., Zhang, F.Z., Chai, Y.M.: Adaboost Object Tracking Algorithm. Pattern Recognition and Artificial Intelligence 22, 475–479 (2009)
5. Freund, Y.: An adaptive version of the boost by majority algorithm. Machine Learning 43, 293–318 (2001)
6. Qi, M.B., Yang, A.L., Jiang, J.G.: A Vehicles Detection and Tracking Algorithm Based on Improved Codebook. Journal of Image and Graphics 16, 406–412 (2011)
7. Kim, K., Chalidabhongse, T.H., Harwood, D., et al.: Background Modeling and Subtraction by Codebook Construction. In: International Conference on Image Processing, pp. 3061–3064. Institute of Electrical and Electronics Engineers Computer Society, Piscataway (2004)
8. Xu, C., Tian, Z., Li, R.F.: A Fast Motion Detection Method Based on Improved Codebook Model. Journal of Computer Research and Development 47, 2149–2156 (2010)
9. Shi, J.B., Tomasi, C.: Good Features to Track. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 593–600. IEEE Press, Los Alamitos (1994)
10. Levi, K., Weiss, Y.: Learning Object Detection From a Small Number of Examples: the Importance of Good Features. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 53–60. IEEE Press, Los Alamitos (2004)
11. Huang, K., et al.: View Independent Human Behavior Analysis. IEEE Transactions on Systems, Man and Cybernetics 39, 1028–1035 (2009)

False Data Attacks Judgment Based on Consistency Loop Model in Wireless Sensor Networks*

Ping Li[1,2], Limin Sun[1,2], Wu Yang[1,**], Qing Fang[1], Jinyang Xie[1], and Kui Ma[3]

[1] Computer and Communication Engineering Institute,
Changsha University of Science and Technology, Hunan, 410114, China
[2] State Key Lab. of Information Security, Institute of Information Engineering,
Chinese Academy of Sciences, Beijing, 100093, China
[3] Wuxi Internet of Things Industry Research Institute,
Jiangsu, 214000, China
{lping9188,fangqing1012}@163.com, sunlimin@iie.ac.cn,
{1013568049,505354246}@qq.com, makui@wsn.cn

Abstract. Wireless sensor networks are usually deployed in complex environments; an attacker can easily inject false data by capturing nodes, causing serious consequences. The main work of this paper is as follows. Firstly, the logical loop model is created based on the estimated value of the source events of every wireless sensor network node. Secondly, each node based on RSSI find neighbor nodes by establishing consistency loop model. Finally, the malicious node is determined by comparing the similarities and differences of the nodes between the two loop models. The simulation shows that this mechanism is effective to inhibit the infringement of malicious nodes to the network, and improve network security performance.

Keywords: malicious node, WSN Logical loop model, Consistency loop model.

1 Introduction

WSN is very susceptible and vulnerable to many kinds of attacks either from inside or outside of the network. It is also clear that most of the security solution adopted for MANET cannot be directly used for WSNs for the same reasons [1]. To protect WSNs against different kinds of vulnerabilities, preventive mechanisms like cryptography and authentication can be applied to prevent some types of attacks. This kind of preventive mechanisms formed the first defense line for WSNs. However, some attacks like wormholes, sinkhole, could not be detected using this kind of preventive mechanisms. In addition, these mechanisms are only effective to prevent from outside attacks and failed to guarantee the prevention of intruders from inside the network.

* This research was supported by the Key Technology R&D Project of Jiangsu Province (Grant No. BE2010026).

** Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 308–315, 2013.

Trust management[2] has now become an additional means to cryptography-based security measures, which can identify selfish and malicious nodes efficiently and solve the security problems for node failure or capture in WSNs. Trust management also can deal with this problem efficiently and enhance the security, reliability and impartiality of the system. But, there are some insurmountable problems provided by current trust management, such as a normal node misjudgment malicious node. Resulting in the entire network node misjudgment, disrupt the entire network order.

Compared with the traditional false data attack essay, the judgment accuracy of the node behavior is higher to false data attacks judgment based on consistency loop model in wireless sensor networks. We propose sensor network model has no fixed requirements to the position of the source event. And the mechanism can be good to find out the wireless sensor network internal malicious nodes compare with traditional password system. Most of all compared with the trust management system, this mechanism hardly has misjudgment to benign node. It improves the safety performance of the entire wireless sensor networks.

2 Related Work

A statistics-based malicious node detection scheme is proposed by Ana Paula R. da Silva etc. in reference [3]. In such a scheme, a series of regulations are predefined to describe the normal behaviors of nodes and further judge the anomaly behaviors of nodes. And the rate of false alarming is quite high because there is no interaction among nodes.

Data trust refers to the trust assessment of the fault tolerance and consistency of data. The trust model presented by Josang [4] was used to deal with uncertainties of data stream in WSNs. Krasniewski put forward a fault-tolerant system TIBFIT based on trust in order to compute the trust value of node in WSNs with the structure of cluster. And Hur [5] presented a security data fusion algorithm based on trust which calculated the trust value of data fusion by examining the consistency of the data. Reference combine above mentioned methods to develop a simplified method of calculating the data trust value.

M.C.Vuran proposes a spatial correlation model for wireless sensor network. This model uses an energy index model to calculate the node between observations and information [6]. Every node in the domain of source event will be encode its observed temperature values and then transmit to the Sink over the network. The Sink receives information about each node decoding, to create logical loop model $H_1, H_2, ..., H_k$, the distance between the loops are equation.

In this paper we put forward the logical loop model and the number of nodes to different loops is equation. Then through RSSI linear segment threshold gradually find consistency loop model. Finally, query the difference of the logical loop model and the consistency loop model to identify malicious node.

3 Network Model and Parameter Optimization

There are some bottlenecks in wireless sensor networks such as limited data transmission limited power, due to the characteristics of the wireless sensor node. In this paper, we propose the logic loop model and consistency loop model of wireless sensor networks to query malicious nodes. We assumed that the event source is modeled as a temperature source. All wireless sensor network nodes will transfer its perception of event source temperature information to the Sink including some malicious node will also tampered with temperature information gradually transferred to the Sink. Finally the Sink through the consistency loop model judgment algorithm to identify malicious nodes. There is some definition as follows:

$-D_{ij}$: The distance between H_i and H_j $D_{ij} = \left| d[H_i] - d[H_j] \right|$.

–CONNECTIVITY : In the wireless sensor networks the signals sent by the node a_i, if the node a_j receives the RSSI value is greater than a specific threshold value P, we called Node a_i and Node a_j are 'Connectivity'. You can also think that $d_{ij} \le p \cdot$

–RING CONNECTED GRAPH : wireless sensor node is specified to the vertex of plane connected graph G (V, E), if the graph is a convex polygon, it says that this G (V, E) is a 'RING CONNECTED GRAPH'.

3.1 Network Model - Logical Ring Model and Consistent Loop Model

In this paper, we put the event source simulation for a temperature source. When the temperature source event occurs, every node near the event source will be encode the observed temperature values and then transmitted over the network to the Sink. The Sink receives all of temperature information and decodes temperature information about each node, using formula 1, divide the perception of temperature information of all the nodes from largest to smallest into m collection to create logical loop model $H_1, H_2, ..., H_m$.

$$Z = Ce^{-\alpha d^{\theta}}, \theta > 0 \tag{1}$$

The main parameters expressed as follows: C : The events source of energy. d : The distance from the source node and event. Z : Energy observations of the wireless sensor node. θ: Type of energy. α : The parameters of control the speed of energy attenuation.

Based on the basic principles of RSSI ranging, When the node a_i issued a ranging signal, and if the RSSI value of a node a_j in the set Q_i is greater than the linear segments reception threshold value p, called the node a_j and the node a_i 'CONNECTIVITY'.

Considering that logical loop model may exist some false information about the malicious nodes. In this paper, consistency loop model based on ‘RING CONNECTED GRAPH’ provide a good solution to this problem. Firstly, query H_i in the logic loop model, then through the basic principles of the RSSI linear segment, and gradually find ‘RING CONNECTED GRAPH’ corresponding with a_i. Finally, H_i in logical loop model in contrast with the consistent loop model to identify malicious nodes.

3.2 Parameter Optimization

As a result of the same as the number of nodes in every loops within the logical loop model, the span of different loops are inconsistent, Based on RSSI linear segment threshold p have a greater impact to find the consistent loop model, affecting the accuracy of the algorithm. This essay Section V will do a detailed analysis.

4 Algorithm Details

Malicious node for false temperature information, the algorithm achieved through the establishment of a logical annular space model and consistent with the annular space model to remove the malicious nodes.

4.1 The Establishment of a Logical Loop Model

As is shown in Figure 1, create a logical loop model are as follows:

1. When the temperature source occurs an events, all nodes which can perceive the temperature source consists of the set Q.
2. All nodes which can perceive the temperature source transmit its observations data $T_1, T_2, ..., T_n$ to the Sink.
3. The sink take observed experimental data from largest to smallest arranged to $T_n', T_{n-1}', ..., T_1'$
4. The sink equally distributes $T_n', T_{n-1}', ..., T_1'$ into m collection of nodes, and each collection precisely contains K nodes to form $Q_1, Q_2, ..., Q_m$ $\{T_n', T_{n-1}', ..., T_{n-k+1}'\} \subseteq Q_1, ..., \{T_k', T_{k-1}', ..., T_1'\} \subseteq Q_m$
5. $Q_1, Q_2, ..., Q_m$ form $H_1, H_2, ..., H_i, ..., H_m$ and $\{H_i \subset S[Q_{i+1}] - S[Q_i]\}$.

4.2 Consistency Loop Model and Malicious Node Judgment

As is shown in Figure 1, create a consistency loop model are as follows:

1. Query a_i belongs to a set of Q_i and adds the node a_i in to the set Q_i'.
2. Query S_i, R_i is the intersection of the intersection of the set Q_i and the set S_i.

3. Find the node a_j in the set R_i when the distance between the node a_i and the node a_j is minimum, repeating the above three, if the new node is a_i then end. In the case all the nodes in the Q_i ' form consistency loop model.

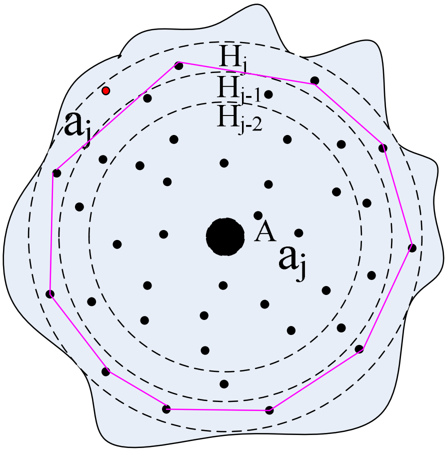


Fig. 1. Logical loop model contains malicious node and consistency ring model

Table 1. Symbols and definition of the algorithm

Character	Explanation
Q	Collection of nodes within the specified range
T_i	Node A perception of temperature values
d_{ij}	The distance between nodes a_i and a_j
S_i	The set of nodes with the distance of the node a_i is less than L
p	The threshold of RSSI linear segment
m	The number of rings of the logical ring Model
$Q_{i\max}$	The maximum value of the set Q_i
$d[H_i]$	The distance between H_i and the event source.
$S[Q_i]$	A point which the distance between it to A is less than $Q_{i\max}$ belong to the regional

4. Compare with the set $Q_i^{'}$ and the set Q_i, if there is a certain one or more nodes contained in the set Q_i does not include in the set $Q_i^{'}$, it is determined that such nodes are malicious nodes which are hoax temperature information.

5 Simulation and Analysis

In this paper the malicious nodes judgment in allusion to monitoring applications for wireless sensor networks. The area inside a radius of 500 * 500 randomly deploy 200 sensor nodes, and randomly generated n (n<=10) malicious nodes (false temperature data node).Our sensor network model have no fixed requirements to the position of the event source, in order to facilitate the simulation we assume that the event source in the regional centers. On a fixed network topology, this experiment simulated a temperature source, then according to the logic loop model and consistency loop model comparison to identify whether the behavior of every node is accurate.

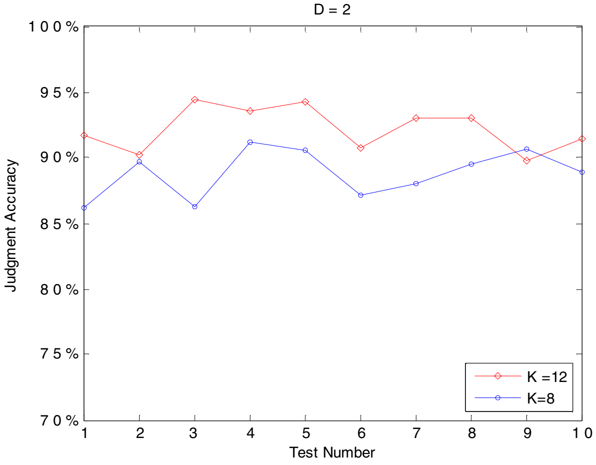


Fig. 2. Relationship between K and accuracy when D=2

Mat lab is the tool for this simulation experiment. As a result of the number of nodes K of each loop in the logical loop model into account play a decisive role in span between loops. However, the spans between the logical loops have a great influence on the threshold of RSSI linear segment (The largest distance d_{ij} of the node connectivity) in the consistent loop model. We use several typical number of logical ring nodes K (8, 12) and a few typical d_{ij} values (2, 5, 8) of this wireless sensor network random simulation. The simulation results are shown in Figure 2, Figure 3 and Figure 4.

Although volatility of each curve was relatively large, but the impact of each variable for the accuracy of the overall judgment can still draw the following conclusion can be seen from the above four graphs. As is shown in Figure 2, When D(D=2) is too small and K is too larger the judgment accuracy of the node behavior not high, its reason is that when D is too small corresponding to the span between the loops is too small. In this case malicious node judgment within the loop is not very successful, and In this case

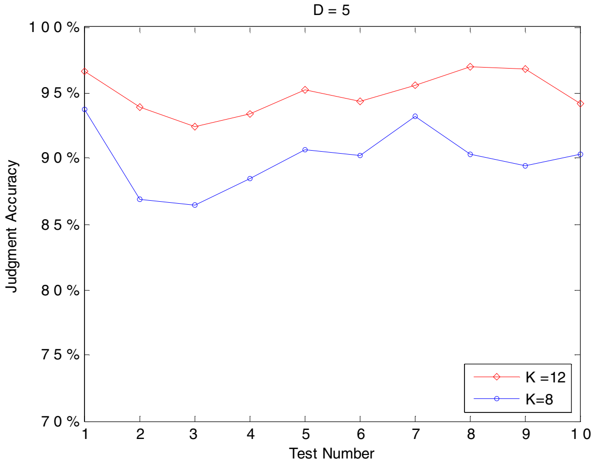


Fig. 3. Relationship between K and accuracy when D=5

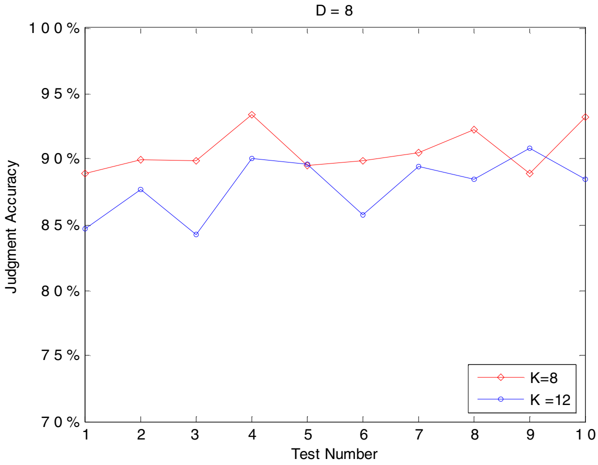


Fig. 4. Relationship between K and accuracy when D=8

The D is more difficult to find a suitable value. We can see from the results in Figure 4 When D(D=8) is too large and K also larger, the threshold value of RSSI is hard to find, the consistency ring model is likely to close away from the event source, even to the internal ring, leading to the entire network are all connected, the entire algorithm almost failure. Our algorithm judge the accuracy of the internal malicious nodes in wireless sensor networks goes down obviously. We can know from Figure 4 the judgment accuracy of the node behavior is higher than the result of the Figure 3, the reason is that find a suitable K and D have a great influence on the judgment accuracy of the node behavior. So to find a suitable K and D is the top priority of the algorithm.

After repeatedly test K and D we found relatively perfect results. And we can know that the accuracy of the entire network is very stable and more than 95% and it meet our requirements.

6 Conclusion

There are a number of security issues in WSN. The traditional method of password-based system does not recognize the internal malicious nodes, at the same time, the method based on trust management are easy to misjudgment some benign nodes. In this paper, we propose the malicious node judgment algorithm based on the logical loop model and consistency loop model are very suitable for detect the internal malicious nodes of WSNs. Experiments show that our algorithm has a higher accurately to the behavior of nodes, it can make up for the above shortcomings. All in all, the system can accurately filter out false information, to improve the network information security.

References

1. Lindsey, S., Raghavendra, C., Sivalingam, K.M.: Data gathering a1gorithms in sensor network using energy metrics. IEEE Trans on Parallel and Distributed Systems 13(9), 924–935 (2002)
2. Yuan, L., Qu, G.: Design Space Exploration for Energy-Efficient Secure Sensor Network. In: ASAP (2002)
3. Silva, A.R.D., Martins, M., Rocha, B.: Decentralized Intrusion Detection in Wireless Sensor Networks. In: Proceedings of the 1st ACM International Workshop on Quality of Service & Security in Wireless and Mobile Networks (Q2SWinet 2005), pp. 16–22 (2005)
4. Josang, A., Ismail, R., Boyd, C.: A Survey of Trust and Reputation Systems for Online Service Provision. Decision Support Systems, 618–644 (2007)
5. Hur, J., Lee, Y., Hong, S.M., Yoon, H.: Trust Management for Resilient Wireless Sensor Networks. In: Proceedings of the 8th International Conference on Information Security and Cryptology, pp. 56–68 (2005)
6. Vuran, M.C., Akyildiz, I.F.: Spatiotemporal correlation theory and applications for wireless sensor networks. Computer Networks Journal 45(3), 245–261 (2004)

Multi-cell Interaction Tracking Algorithm for Colliding and Dividing Cell Dynamic Analysis

Mingli Lu[1,2], Benlian Xu[2], Andong Sheng[1], and Peiyi Zhu[2]

[1] School of Automation, Nanjing University of Science & Technology,
210094 Nanjing, China
[2] School of Electrical & Automatic Engineering, Changshu Institute of Technology,
215500 Changshu, China
luxiaowenwp@sohu.com, xu_benlian@yahoo.com.cn,
shengandong@mail.njust.edu.cn, zpy2000@126.com

Abstract. Cell motion analysis contributes to research the mechanism of the inflammatory process and to the development of anti-inflammatory drugs. This paper aims to develop an accurate and robust algorithm to track multiple colliding cells and further characterize the dynamics of each cell. First, a hybrid cell detection algorithm is proposed to obtain reliable measurements in cell collision images. Second, a variant of interacting multiple models particle filter is designed for analysis of cell motion behaviors. The simulation results show that our algorithm could obtain favorable performance compared with other methods.

Keywords: Cell Segmentation, Cell Tracking, Interacting Multiple Models Particle Filter.

1 Introduction

Over the past decade, a number of cell tracking algorithms have been developed. These algorithms concentrate on a variety of cell types and are based on different tracking methods. These cell tracking approaches in the literature can be categorized into three categories, namely, tracking based on detection and segmentation [1], tracking based on evolving model and tracking based on probabilistic approach. In terms of the first category, cell dynamic movement analysis generally consists of cell detection and cell correspondence. This approach is computationally efficient and robust when cell density is low. In the second tracking approaches, the features of cell such as shape, position, boundaries are initialized in the first frame, and then the features of cell evolve frame by frame. Active contour [2], level set [3] and Mean-shift [4] are the example of this type of approaches. Tracking methods based on probabilistic framework have been presented in [5]. These approaches are more robust to under low resolution and SNR scenarios.

In this paper, we employ a mixture scheme to combine the detection-segmentation-based approach with the Bayesian probabilistic approach. First, a hybrid cell detection segmentation algorithm is employed to maximize detection rate and minimize missing

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 316–324, 2013.

rate. To match different cell motion features, an object interaction based interacting multiple models tracking algorithm, coupled with modes of non-interacting, colliding and dividing, is proposed to analysis cell motion behaviors.

2 Methods

Our work starts by processing the input images sequentially, and the output is the cell trajectories, cell migration velocities and turn rate.

2.1 A Hybrid Cell Detection Algorithm for Image Sequences

The cell detection is a challenging job due to a high noise level in time-lapse microscopy images and wide ranging in intensity and shape. Image enhancement removes blurring and noise, and increases contrast, etc. After the process of image enhancement, "A Hybrid Cell Detection Algorithm" is used to segment overlapping or adhesion cells. This combination method consists of threshold processing, holes filling, noise removal, image dilation and shape and boundary constraint. The description of threshold process can be briefly presented as Table 1.

Table 1. The description of threshold process

Step (1): An initial threshold (T) is chosen, and this can be done randomly or according to any other methods desired.

Step (2): creating two sets

$G_1 = \{ f(m,n)$[1]$: f(m,n) > T \}$ (Object pixels)

$G_2 = \left\{ f\left(m,n\right) : f\left(m,n\right) \le T \right\}$ (Background pixels)

Step (3): compute the average intensity of each set

$m_1 =$ Average intensity value of G_1

$m_2 =$ Average intensity value of G_2

Step (4): create the new threshold

$T^{'} = (m_1 + m_2) / 2$.

Step (5): Go back to **Step (2)** until convergence is reached

The overview of the proposed detection method is given in Fig.1.

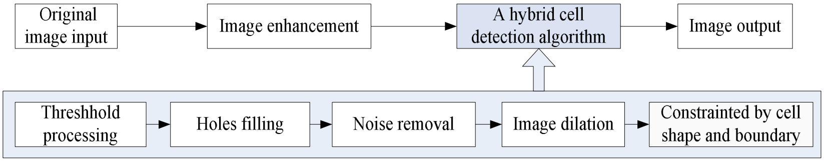


Fig. 1. The overview of the proposed cell detection method

[1] Note, $f\left(m,n\right)$ is the value of the pixel located in the m^{th} column, n^{th} row.

2.2 Cell Interaction Based interacting Multiple Models Tracking Algorithm

In the problem of cell tracking, however, since cell exhibits various behaviors, such as moving random fluctuation, collision in different frames. Therefore, constant models are not competitive and robust to accurate tracking. Through observations of cell motion from several image sequences, we find that cells move to other direction dramatically when they collide and appear merged with other cells, i.e. motion speed, motion direction and area of cells vary over time. So, Cell Interaction based interacting multiple models tracking algorithm with random turn rate variable is developed. The cell motion system is as follows.

$$X_k = F_{k,r} X_{k-1} + G_{k,r} w_{k,r} \tag{1}$$

$$Z_k = H_k X_k + v_k \tag{2}$$

To catch up with cell motion uncertainties, we augment the standard state in general IMMPF by the unknown cell turn rate ω_k and cell area s_k, resulting in state vector $X_k = (x_k, y_k, \dot{x}_k, \dot{y}_k, \omega_k, s_k)^T$. Z_k is the detection measurement vector, $w_{k,r}$ and v_k are the mode-dependent process and measurement random noise sequences, which assumed to be mutually independent with covariance $Q_{k,r}$ and R_k, respectively. The random parameter r is model index. The successful implementation of our proposed algorithm relies on two aspects, namely, the determination of turn rate ω_k and the way of modeling individual modes. $\omega_{k,m}$ is predicted at the end of previous $k-1$ scan upon the base of last value $\omega_{k-1,m}$. Another key issue is to determine the evolvement of cell state at each mode. As we observe in many cell tracking scenarios, three cases often happen, namely, one cell evolves independently without any interaction with other cell; one cell collides with other cell for one or more frames; and one cell or merged cell divides into two or more individual cells in the next frame. Without loss of generality, therefore, we adopt three modes for cell tracking, namely, augmented constant velocity (ACV), an augmented variable coordinate turn models for cell collision (ACT1), and an augmented variable coordinate turn models for cell division (ACT2).

Mode 1 (ACV): If a given cell does not undergo collision or division, it moves smoothly with minimum appearance change, the following model is adopted as below.

where $w_{k,1}$ is the system process Gaussian white noise at the sampling index k, assumed to be zero mean with 6×6 covariance.

$$F_k^{(1)} = \begin{bmatrix} 1 & 0 & T & 0 & 0 & 0 \\ 0 & 1 & 0 & T & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$Q_{k,1} = diag(\sigma^2_{x,k,1}, \sigma^2_{y,k,1}, \sigma^2_{\dot{x},k,1}, \sigma^2_{\dot{y},,k1}, \sigma^2_{\omega,k,1}, \sigma^2_{s,k,1}) \tag{4}$$

$$G_{k,1} = \begin{bmatrix} \frac{1}{2}T^2 & 0 & T & 0 & 0 & 0 \\ 0 & \frac{1}{2}T^2 & 0 & T & 0 & 0 \end{bmatrix}^T . \tag{5}$$

Mode 2 (ACT1): If one cell collides with the other one in a given frame, its motion speed, direction and area are assumed to vary accordingly. Meanwhile, cell area will increase as well, but it is not exceed the sum of two colliding cells. Thus, the following state evolvement is proposed as

$$F_k^{(2)} \triangleq \begin{bmatrix} 1 & 0 & \frac{\sin\omega_{k,1}T}{\omega_{k,1}} & \frac{\cos\omega_{k,1}T}{\omega_{k,1}} & f_{11,k} & 0 \\ 0 & 1 & \frac{1-\cos\omega_{k,1}T}{\omega_{k,1}} & \frac{\sin\omega_{k,1}T}{\omega_{k,1}} & f_{21,k} & 0 \\ 0 & 0 & \cos\omega_{k,1}T & -\sin\omega_{k,1}T & f_{31,k} & 0 \\ 0 & 0 & \sin\omega_{k,1}T & \cos\omega_{k,1}T & f_{41,k} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha \end{bmatrix} \tag{6}$$

where control variable α considers the effect of collision on the resulting area of merged cell, we usually take $1 < \alpha \leq 2$ according to the above assumption; the non-linear system Jacobi term $M_{1,k} \triangleq [f_{11,k}, f_{21,k}, f_{31,k}, f_{41,k}]^T$ is defined as [6]

$$f_{11,k} = \frac{\dot{x}_k(\omega_{k,1}T\cos\omega_{k,1}T - \sin\omega_{k,1}T)}{\omega^2_{k,1}} - \frac{\dot{y}_k(\omega_{k,1}T\sin\omega_{k,1}T - 1 + \cos\omega_{k,1}T)}{\omega^2_{k,1}} \tag{7}$$

$$f_{21,k} = \frac{\dot{x}_k(\omega_{k,1}T\sin\omega_{k,1}T + \cos\omega_{k,1}T - 1)}{\omega^2_{k,1}} - \frac{\dot{y}_k(\omega_{k,1}T\cos\omega_{k,1}T - \sin\omega_{k,1}T)}{\omega^2_{k,1}} \tag{8}$$

$$f_{31,k} = -T(\dot{x}_k \sin\omega_{k,1}T + \dot{y}_k \cos\omega_{k,1}T) \tag{9}$$

$$f_{41,k} = T(\dot{x}_k \cos\omega_{k,1}T - \dot{y}_k \sin\omega_{k,1}T) \tag{10}$$

In experiment, we take $\omega_{k,1} = \hat{\omega}_{k-1,1}$, $G_{k,2} = G_{k,1}$ and $Q_{k,2} = Q_{k,1}$.

Mode 3 (ACT2): At time k, if colliding cells split away from one another, each moves away and appears as an individual cell. The motion speed, direction and area of interested cell change from the previous time to current time. Following the same rule as ACT1, we define the state transition mode as

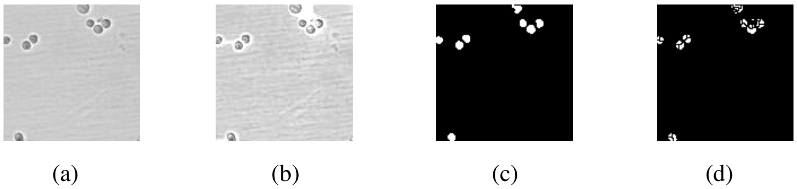
$$F_k^{(3)} \triangleq \begin{bmatrix} 1 & 0 & \frac{\sin\omega_{k,2}T}{\omega_{k,2}} & \frac{\cos\omega_{k,2}T}{\omega_{k,2}} & f_{12,k} & 0 \\ 0 & 1 & \frac{1-\cos\omega_{k,2}T}{\omega_{k,2}} & \frac{\sin\omega_{k,2}T}{\omega_{k,2}} & f_{22,k} & 0 \\ 0 & 0 & \cos\omega_{k,2}T & -\sin\omega_{k,2}T & f_{32,k} & 0 \\ 0 & 0 & \sin\omega_{k,2}T & \cos\omega_{k,2}T & f_{42,k} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \beta \end{bmatrix} \tag{11}$$

where control variable β takes into account the effect of cell splitting on the resulting cell area from the previous time to the current time, and usually is smaller than the previous one. Therefore, β lies in the range of $(0,1)$. In terms of the non-linear system Jacobi term $M_{2,k} \triangleq [f_{12,k}, f_{22,k}, f_{32,k}, f_{42,k}]^T$, we follow the same formula as Eqs. (7) to (10) as long as the turn rate $\omega_{k,1}$ in each corresponding component is replaced by $\omega_{k,2}$. Similarly, we have $\omega_{k,2} = \hat{\omega}_{k-1,2}$, $G_{k,3} = G_{k,1}$ and $Q_{k,3} = Q_{k,1}$.

If a cell moves in a smooth way, the cell's distance difference between two consecutive frames less than threshold δ_1 is considered to be the same cell. If a cell undergoes colliding, it is associated in the case of both distance difference less than threshold δ_1 and area difference less than threshold δ_2.

3 Experiments

In this section, experiments of our algorithm on low-contrast cell image sequences are conducted to verify the effectiveness of the proposed method, both for detection and track. These experiment data include various challenging scenarios, such as cells adhesion, different cell dynamics, cells colliding or varying number of cells in different frames.

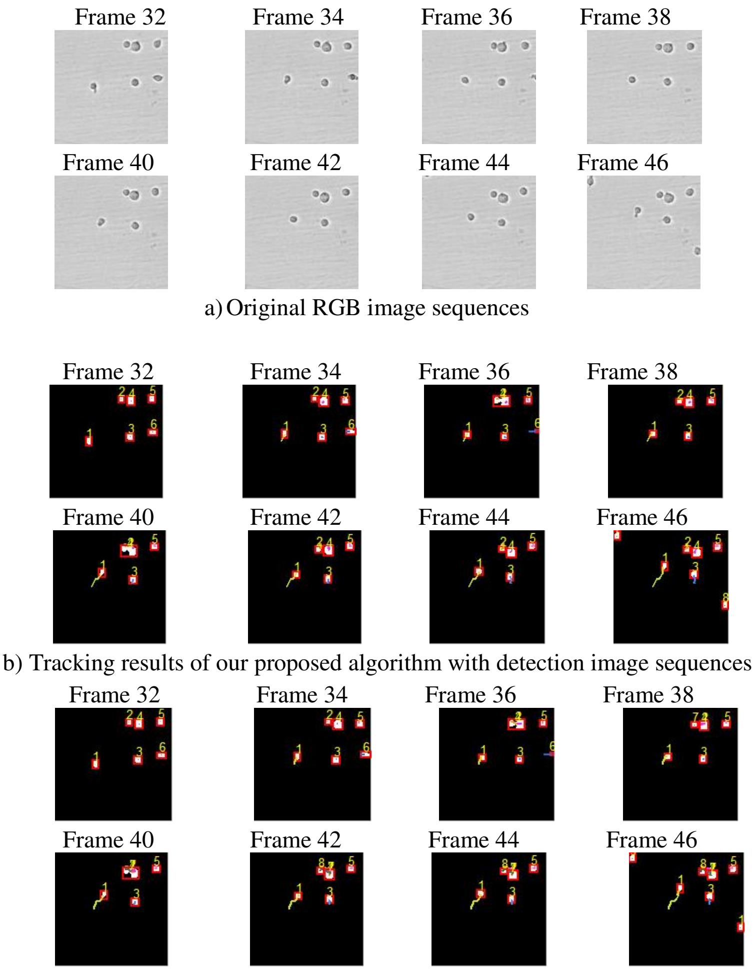


(a) (b) (c) (d)

Fig. 2. Cell detection. From left to right, the original image, the enhanced image, image, the result image of proposed method, the watershed segmentation image.

As illustrated in Figs. 2 (c), our proposed hybrid cell detection algorithm gives a set of well separated cell detections after the enhanced step, the results of which are plotted in Figs.2 (b). Figs.2 (d) presents the resulting detections using the watershed algorithm, but the phenomenon of oversegmentation occurs.

In the cell tracking experiment, the targets are modeled by rectangular blobs. Three modes in our proposed cell interaction IMMPF are adopt, namely, ACV, ACT1 and ACT2. The Markov transition matrix between three modes is assumed constant and set empirically as $M = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$ .The initial state of cell i is represented as $\boldsymbol{X}_0^i = [x_0^i, y_0^i, \dot{x}_0^i, \dot{y}_0^i, \omega_0^i, s_0^i]^T$, $\dot{x}_0^i = 0.3\mu m / s$, $\dot{y}_0^i = 0.3\mu m / s$, $\omega_0^i = 0.8 rad / s$, s_0^i takes the value of cell i detection area. Other parameters are listed below: for simple $Q_{k,r} = diag(30, 30, 0.1, 0.1, 0.01, 0.01)(r = 1, 2, 3)$; $R = diag(1, 1, 1)$; $\delta_1 = 20$; $\delta_2 = 7$; $\alpha = 1.9$; $\beta = 0.5$; $T = 1s$, $N = 500$. In IMMPF algorithm, all parameters are taken the same values as our algorithm except using a fixed turn rate constant $\omega = 0.8 rad / s$. Fig.3 (a) illustrates the original RGB cell image sequences, and an example of successful tracking with our proposed method is shown in Fig.3 (b). According to the tracking results, our algorithm could tackle the following challenging cases: cell 2 collides with cell 4 in frame 36 and splits as shown in frame 38; cell 2 collides again with cell 4 in frame 40 and splits in frame 42; cell 6 moves right, partially leaves the field of view in frames 36, and fully leaves the field of view in frame 38; and new cells 7 and 8 enter the field of view in frame 46. The performance is degraded when the general IMMPF is used, as shown in Fig.3(c). A new cell 7 is wrongly initiated when the detected two cells split in frame 38, and the original cells 2 and 4 are merged together as a cell. Afterwards, the similar results occur in frames 40 and 42. Lack of velocity ground truth, we evaluated the performance of our algorithm by comparison with manual tracking results. Instant velocity estimate of cell 1 per frame for our proposed algorithm versus IMMPF and manual tracking is shown in Fig.4. It is obvious that the instant velocity curves of our proposed algorithm are closer to the truth than that of the IMMPF. The difference between our proposed algorithm and manual tracking is very small. The results indicate that our algorithm may replace laborious manual procedures. Estimate of turn rate of selected cell 4 using our proposed algorithm is plotted in Fig.5. It can be shown that the cell turn rate changes over time. Fig. 6 illustrates the mode probability of cell 4. The change of the mode probabilities in Fig. 6 is the same as the change of the turn rate of the system described in Fig. 6. It is obvious that the algorithm works well and the mode switching is right.

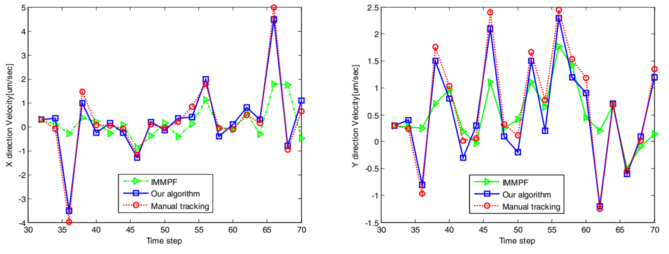


a) Original RGB image sequences

b) Tracking results of our proposed algorithm with detection image sequences

c) Tracking results of IMMPF with detection image sequences

Fig. 3. Multiple cells tracking with colliding and varying number of cells in different frames



a) Cell 1 in x -direction b) Cell 1 in y -direction

Fig. 4. Instant velocity estimate per time step using various methods

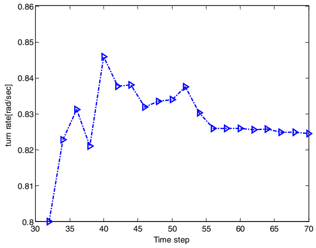


Fig. 5. Turn rate estimate of cell 4

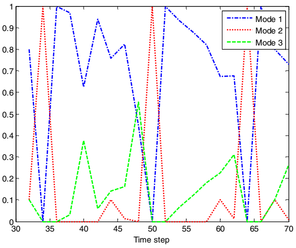


Fig. 6. Mode probability of cell 4

4 Conclusions

In this paper, we present a method for reliably detecting overlapping or adhesion cells and tracking multiple cells with collision or varying number in different frames. After image enhancement, a hybrid cell detection algorithm is used to cell segmentation. Cell interaction based interacting multiple models tracking algorithm is developed to analysize the motion behaviors. The simulation results are shown that our proposed algorithm could track simultaneously multiple cells of colliding or cells of entering and/or leaving field of view, etc.. Furthermore, it could provide accurate dynamic estimate of each cell, such as velocity.

Acknowledgments. This work is supported by national natural science foundation of China (No.61273312), national natural science foundation of Jiangsu province (No.BK2010261), and partially supported by national natural science foundation of China (No. 61273076).

References

1. Yang, L., Qiu, Z., Lu, W.: A New Framework for Particle Detection in Low-SNR Fluorescence Live-Cell Images and Its Application for Improved Particle Tracking. IEEE Trans. Biomed. Eng. 59(7), 2040–2050 (2012)
2. Nilanjan, R., Acton, S.T., Ley, K.: Tracking leukocytes in vivo with shape and size constrained active contours. IEEE Trans. Med. Imag. 21(10), 1222–1235 (2002)

3. Mukherjee, D.P., Ray, N., Acton, S.T.: Level set analysis for leukocyte detection and tracking. IEEE Trans. Image. Process. 13(4), 562–572 (2004)
4. Debeir, O., Ham, P.V., Kiss, R., et al.: Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes. IEEE Trans. Med. Imag. 24(6), 697–711 (2005)
5. Smal, I., Niessen, W., Meijering, E.: Bayesian tracking for fluorescence microscopic imaging. In: 3rd IEEE International Symposium on Biomedical Imaging: Macro to Nano, pp. 550–553 (2006)
6. Semerdjiev, E., Mihaylova, L., Li, X.R.: Variable- and fixed-sructure augmented IMM algorithms using coordinate turn model. In: Proceedings of the Third International Conference on Information Fusion, vol. 1, pp. 25–32 (2000)

An Study of Indoor Localization Algorithm Based on Imperfect Signal Coverage in Wireless Networks*

Ping Li[1,2], Limin Sun[1,2], Qing Fang[1,**], Jinyang Xie[1], and Wu Yang[1], and Kui Ma[3]

[1] Computer and Communication Engineering Institute, Changsha University of Science and Technology, Hunan, 410114, China
[2] State Key Lab. of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100093, China
[3] Wuxi Internet of Things Industry Research Institute, Jiangsu, 214000, China

{lping9188,fangqing1012}@163.com, sunlimin@iie.ac.cn, {505354246,1013568049}@qq.com, makui@wsn.cn

Abstract. Existing localization algorithms didn't consider the important factor of the antenna measuring angles. And most wireless indoor localization algorithms require a site survey process which is time-consuming and labor-intensive. This paper presents a featured region localization algorithm without site survey and discusses the measured angle in different intervals. According to the relationship among the fingerprints sets of the same angle interval, our proposed algorithm is used to find the featured points within the region. Regions of points are determined by calculating Euclidean distance between the points and APs (Access Points). Experiments are conducted in an 8m by 8m laboratory, and results show that this algorithm has superior performance compared with existing algorithms.

Keywords: localization, measured angles, site survey, featured points.

1 Introduction

With the development of wireless communication technology, wireless location services are also increasingly receiving widespread attention. In most of applications such as pervasive medicare, smart space, wireless sensor surveillance, mobile peer-to-peer computing, etc., location is one of the most essential issues. Existing location algorithm has two major categories: range-based algorithm and range-free algorithm [1]. Range-based localization used the wireless signal propagation model, but the parameters of this model are related in environment. To determine the parameters of environment, we must carry out a large number of site surveys.

* This research was supported by the Key Technology R&D Project of Jiangsu Province (Grant No. BE2010026).

** Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 325–333, 2013.

Parameters changed with the dynamic environment. This localization algorithm is also time-consuming, labor-intensive and non-adaptive. Range-free algorithm has lower cost and lower energy consumption compared with Range-based algorithm, but its positioning accuracy is poor. Meanwhile, in actual measurement, the measured fingerprints have large errors due to the non-omni direction antenna. Propagation model would cause serious deviation from the true value of distance.

In this study, we proposed a new range-free algorithm named featured region localization algorithm. By exploiting fingerprint characteristics of featured points and different angles, we successfully remove the site survey process of traditional approaches. The vectors which can characterize featured points are addressed. Localization service is achieved by compare the RSSI (Received Signal Strength Indication) value of the same angle. Experimental results show that the accuracy of the indoor localization algorithm can achieve about 90%, which is competitive to existing solutions.

The rest of the paper is organized as follows. We discuss the existing approaches of localization within an indoor environment in Section 2. Section 3 presents our design framework. In Section 4 summarizes the entire working process of this algorithm. The prototype implementation and experiments are discussed in Section 5. Finally, in Section 6 we render our conclusion.

2 Related Work

Location information is essential for a wide range of pervasive and mobile applications, such as wireless sensor networks, mobile social networks, location-based services, smart space, etc., especially indoor localization. Many techniques have been proposed in the past two decades.

RSSI-based algorithm [2] calculates the distances between APs and points through measured RSSI values. This method demand large numbers of anchor nodes. And the irregular RF propagation characteristics, such as indoor multi reflection, non-line-of-sight, etc., cause localization accuracy decreased. Time of arrival (TOA) algorithm [3] calculates the distances by measuring the signal transmit time. Time difference of arrival (TDOA) algorithm [4] installed ultrasonic transducer and RF transceiver on nodes. Anchor nodes simultaneously transmit ultrasonic waves and electromagnetic waves, and receiving nodes calculate the distance between two points by calculating the arrival time difference of the two signals. Angle of arrival (AOA) algorithm [5] relies on antenna array to obtain angles information and using angles information to calculate distances. Range-free algorithm estimate distances or coordinates from nodes to anchor nodes through the network connectivity. APIT (Approximate Point-in-triangulation) algorithm [6]: Target nodes select three adjacent anchor nodes arbitrary. Then test target nodes whether located in the triangle which composed by the three nodes, and used different anchor nodes combination to test until exhausted all combinations or achieve the required localization accuracy. DV-Hop (Distance Vector-HOP) algorithm [7] collects information of adjacent nodes from the network. It calculates the shortest path between non-adjacent nodes, and used the coordinates of known nodes to estimate jump distance. Amorphous Positioning algorithm [8] uses offline jump distance for estimating. Like DV-Hop, Estimated value is improved by exchange information of adjacent nodes. N-Hop multilateration algorithm [9] uses Kalman filter for cycle refinement. Self-positioning algorithm [10] (SPA) established a local coordinate system on the basis of each node's communication range, and also established a global coordinate system through exchange information of nodes. Nodes can calculate its own position in the coordinate system which established by the node with N hops apart.

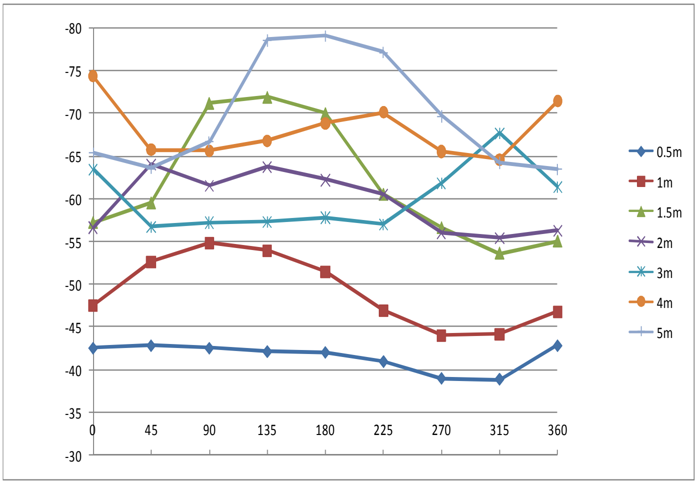


Fig. 1. Measured fingerprints in different distances and angles

However, in actual measurement, RSSI value has great relationship with measured angle. As shown in Fig. 1, changed the measure angle would cause a great change of RSSI values when the distance is constant. The amplitude reached about 10dB when the distance is far, this would seriously affect the positioning accuracy.

3 Framework

Without site survey, the key challenge of localization is how to figure out the locations through measured RSSI values. In this section, we provide a matching based technique that finds a perfect match between logical featured vectors and the truth featured points. To identify logical featured vectors through training fingerprints, and correct the featured vectors through the relationship that the angles should satisfied. Because of there are many measuring angles, we divided 360° into several intervals, and each interval is represented by an angle. Related terms are shown in Table 1.

3.1 Relational Mapping

Assuming that there are t featured points $a_1, a_2, ..., a_t$ in location area. Given a vector $f = \{s_1, s_2, ..., s_k, \alpha\}$, if f satisfies the judgment conditions $r(s_1, s_2, ..., s_{k'}) = true$, then Equation (1) is hold,

$$Location(f) = a_i, i = 1, 2, ..., t \tag{1}$$

Equation (1) indicates that f is a featured vector of the featured point a_i in the featured direction α .

If vectors $f_1, f_2, ..., f_j$ satisfy the judgment conditions $R(f_1, f_2, ..., f_j) = true$, then $f_1, f_2, ..., f_j$ are the featured vectors in j directions of featured points.

Table 1. Related terms

Symbol	Explanation
s_{ij}	The i-th measured RSSI value of j-th AP
α_i	The angle of i-th measuring
F_i	All the measured fingerprints in the direction i , $i = 1, 2, 3, 4$
ξ	The threshold of RSSI measurement error in the same conditions
σ	The threshold of angle measuring error
a_i	Featured points
f	Training set vectors
d	Euclidean distance
r, R	The judgment conditions of the featured points

3.2 Determining Logical Featured Vectors of Featured Points

APs are generally distributed in a grid pattern. Four APs constitute a square region, when access to any regions, the four corresponding APs are easy be found as they have the strongest signal strength by measured RSSI values. We consider four directions of the central featured point.

Property 1: Suppose there are n training fingerprints $f_i = \{s_{i1}, s_{i2}, s_{i3}, s_{i4}, \alpha_i\}, i = 1, 2, ..., n$, for any fingerprint f_j , if it satisfies the *numerical criterion* i , it can be classified into direction i fingerprints set.

$$\textit{Numerical criterion 1:}\ \left|s_{j1} - s_{j2}\right| \le \xi, \left|s_{j3} - s_{j4}\right| \le \xi, s_{j1}, s_{j2} > s_{j3}, s_{j4} \tag{2}$$

$$\textit{Numerical criterion 2:}\ \left|s_{j2} - s_{j3}\right| \le \xi, \left|s_{j4} - s_{j1}\right| \le \xi, s_{j2}, s_{j3} > s_{j1}, s_{j4} \tag{3}$$

$$\textit{Numerical criterion 3: } \left|s_{j3}-s_{j4}\right|\le\xi,\left|s_{j1}-s_{j2}\right|\le\xi,s_{j3},s_{j4}>s_{j1},s_{j2} \qquad (4)$$

$$\textit{Numerical criterion 4: } \left|s_{j1}-s_{j4}\right|\le\xi,\left|s_{j2}-s_{j3}\right|\le\xi,s_{j1},s_{j4}>s_{j2},s_{j3} \qquad (5)$$

Suppose that F_1,F_2,F_3,F_4 denote the fingerprints set which satisfy *numerical criterion* 1, 2, 3, 4, respectively. Arbitrary select a vector in F_1,F_2,F_3,F_4 respectively and named f_n,f_w,f_s,f_e.

If f_n,f_w,f_s,f_e satisfy formula (6), the four vectors represent the logical vectors in the four directions of the central featured point. Assume that $\alpha_n,\alpha_w,\alpha_s,\alpha_e$ represent the angle of direction 1 to direction 4, respectively.

$$\begin{cases}\left|s_{n2}\text{-}s_{w2}\right|\le\xi,\left|s_{w3}\text{-}s_{s3}\right|\le\xi,\\ \left|s_{s4}\text{-}s_{e4}\right|\le\xi,\left|s_{e1}\text{-}s_{n1}\right|\le\xi,\\ \left|s_{n3}\text{-}s_{e3}\right|\le\xi,\left|s_{n4}\text{-}s_{w4}\right|\le\xi,\\ \left|s_{w1}\text{-}s_{s1}\right|\le\xi,\left|s_{e2}\text{-}s_{w2}\right|\le\xi.\end{cases} \qquad (6)$$

3.3 Fingerprints Correction of Central Point

Due to measurement errors, the four fingerprints f_n,f_w,f_s,f_e are determined by the previous section may cause the result not unique. Select a vector in each direction, when the angle of the vector satisfies *angle criterion 1*, then the four vectors are the fingerprints of central featured point.

$$\textit{Angle criterion 1: } \begin{cases}\left|\alpha_w-\alpha_n-90\right|\le\sigma,\\ \left|\alpha_s-\alpha_w-90\right|\le\sigma,\\ \left|\alpha_e-\alpha_s-90\right|\le\sigma.\end{cases} \qquad (7)$$

If the vectors satisfy the *angle criterion 1* are still not unique, then vectors are randomly selected as featured vectors of central featured point in localization.

3.4 Divided Angle Interval

We study the algorithm in four directions, but in actual measurement the angle changed between 0-360. So angle need to be divided into different intervals, and determine the measured angle at which interval in location services. In localization phase, if the measured angle $\alpha\in(\alpha_i\text{-}45^\circ,\alpha_i+45^\circ)$, then using the fingerprint in the direction α_i as the representative of this interval.

4 Processing

4.1 Fingerprints Training

Assuming that APs are omni-directional and known their coordinates. Because of RSSI values obtained have a great relationship with the antenna directions, the RSSI values have a difference of about 10dB when they obtained in different angles at the same location. Smartphones integrated many sensor, such as gyroscope, it can measure the direction and we can use this characteristic to obtain the antenna direction of mobile phone when measuring fingerprints. In fingerprints training phase, the users with smartphones walking indoor with their job requires and periodically record the measured RSSI value of each AP and record the direction by the gyroscope, denoted $f_1, f_2, \ldots, f_n$.

4.2 Location Services

Use above methods to find the central point of the rule quadrilateral formed by four APs. The central point characterized by the measured RSSI values of the four APs in four directions. In location services phase, location steps as follows:

1. Users with smartphones enter the area covered by four APs. Determined the user is located in which quadrilateral through measuring RSSI values;

2. Measure RSSI values and angle of the four APs by smartphones, they can be written as: $f_c = \{s_{c1}, s_{c2}, s_{c3}, s_{c4}, \alpha_c\}$;

3. According to the measured value f_c, we can determine the angle interval.

4. Assume that the fingerprints of the four APs and the featured point in this interval are f_1, f_2, f_3, f_4 f_z , then calculated the Euclidean distance between f_c and f_1, f_2, f_3, f_4, f_z: $d_i = \sqrt{\sum_{n=1}^{4}(s_{cn} - s_{in})^2}$, where d_i $i = 1, 2, 3, 4, z$.

5. Compare the five Euclidean distance, if $d_j = \min(d_i)$, then the point is in the circular area which its center is j .

5 Experimentation and Evaluation

The experiments are conducted in the laboratory which is 8m by 8m. The four APs are distributed in the four corners. The indoor environment is divided into nine regions.

In these experiments, we compared the localization accuracy of angular-based and angular-free localization algorithm. Each algorithm conduct 10 groups experiments and the number of training fingerprints are 1000, and each group are 100 statistics. As shown in Fig. 2, the accuracy of angular-based is higher than angular-free due to the mobile antenna omni-direction insufficient.

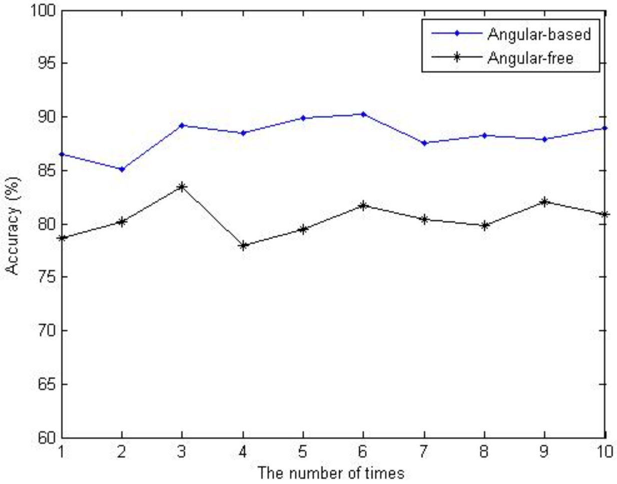


Fig. 2. The localization accuracy of angular-based and angular-free

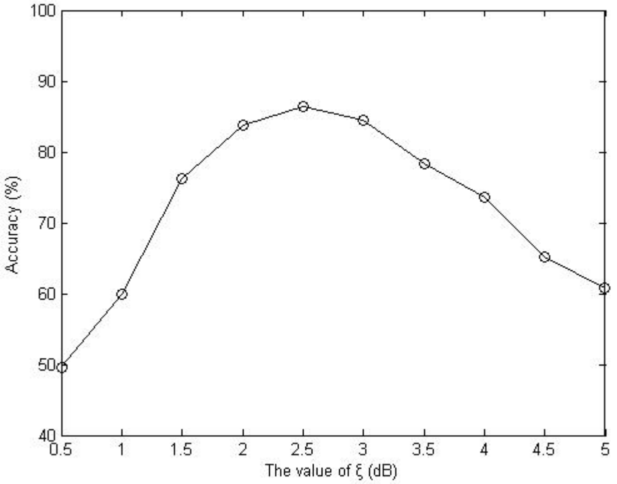


Fig. 3. Relationship between ξ and accuracy

As Fig. 3 illustrate, this algorithm has the best performance when the measurement error threshold ξ is approximately 2.5dB. When locating central featured point, if ξ is too small we can't obtain a good discrimination degree. On the other hand, lager ξ indicates that more samples of central featured point are appropriate, leading to higher difficulty for location.

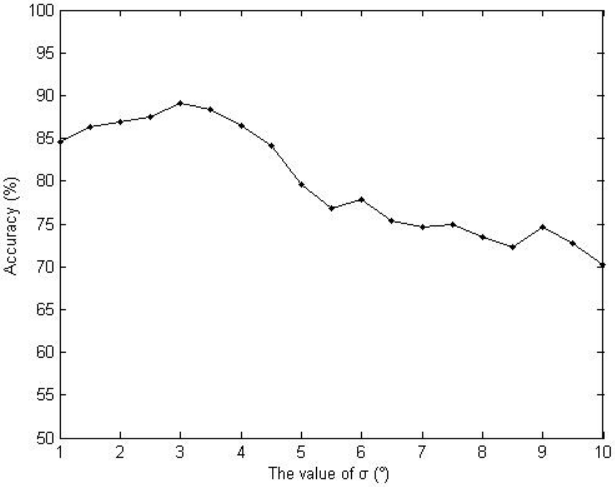


Fig. 4. Relationship between *σ* and accuracy

As Fig. 4 illustrate, it has balance performance when measuring angular error threshold σ change from 1° to 4°, it has the best performance when σ is approximately 3°. When the value is too small, some accurate fingerprints may be eliminated by correcting, but the whole performance is better. When σ is greater than 4°, the accuracy rate drops quickly. When the value is too large, some error fingerprints may also be selected in correction phase.

6 Conclusion

Previous indoor localization approaches mostly rely on site survey which is labor-intensive and time-consuming over every location. This paper analyzes the problem that the impact of antenna measure angle. The algorithm reduces the site survey of labor and time consuming and discuss the angle as the main factors. Found the fingerprints on different directions of the featured point by measuring fingerprints and discussed it in different angles. Angular relationship is used for fingerprints correction. Due to the measured angle range changed from 0 to 360, we divided the angle into several intervals. We implement experiment in a laboratory and it achieves accuracy of 90%. Through experiments, we obtained the optimum value of the measurement error threshold ξ and the measured angle error threshold σ. We believe this algorithm demonstrates its advantage on low human cost, a long-standing and universal will in wireless indoor localization.

References

1. Mert, B., Liu, M., Shen, W.M.: Localization in cooperative wireless sensor networks: A review, 438–443 (2009)
2. Girod, L., Byehovskiy, V., Elson, J.: Localization tiny sensors in time and space: A case study. In: Proceedings of the 2002 IEEE International Conference on Computer Design, pp. 214–219 (2002)
3. Harter, A., Hopper, A., Steggles, P.: The anatomy of a context-aware application. In: Proceeding of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, pp. 59–68. ACM Press (1999)
4. Girod, L., Estrin, D.: Robust range estimation using acoustic and multimodal sensing. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2001), pp. 1312–1320. IEEE Robotics and Automation Society (2001)
5. Nieulescu, D., Nath, B.: Ad hoc positioning system (APS) using AoA. In: Proceedings of the IEEE INFOCOM, San Franeisco, pp. 1734–1743 (2003)
6. He, T., Huang, C., Blum, B.M.: Rang-free localization schemes for large scale sensor networks. In: MobiCom 2003, pp. 81–95 (2003)
7. Niculescu, D., Nath, B.: DV based positioning in ad hoc networks. Journal of Telecommunication Systems 22(1/4), 267–280 (2003)
8. Agpal, R.N., Shrobe, H., Bachrach, J.: Organizing a global coordinate system from Local Information on an Ad Hoc sensor network. In: Proc. of Int. (2003)
9. Savvides, A., Park, H., Srivastava, M.B.: The n-hop multilateration primitive for node localization problems. Mobile Networks and Applications 8, 443–451 (2003)
10. Acpkun, S., Hamdi, M., Hubaux, J.P.: GPS-Free positioning in mobile ad-hoc networks. Cluster Computing 5(2), 157–167 (2002)

Group-Based Overhead Limiting for Stability Routing in Ad Hoc Networks

Xi Hu[1], Cong Wang[1], Siwei Zhao[2], and Xin Wang[3]

[1] Northestern University at Qinhuangdao, 066004, Hebei, China
[2] School of Electronic and Information Engineering, Beijing Jiaotong University, 100044, Beijing, China
[3] Information and Control Engineering Faculty, Shenyang Jianzhu University, 110168, Shenyang, China
{huxi214,congw1981}@gmail.com

Abstract. Two critical issues of designing scalable routing algorithm in ad hoc networks are that it should be robust to frequent path disruptions caused by node mobility and limit routing overhead. This paper argues grouping nodes with the motion direction of source to limit the range of RREQ broadcast. This kind of grouping also ensures that nodes, belonging to the same group, are more likely to establish longtime existent paths as they have similar motions. During route discovery, only the nodes which are in same group with source are allowed to rebroadcast RREQ. So the destination can select a most stable route to reply. The performance is evaluated through computer simulation with NS2. Simulation results indicate the proposed routing algorithm can limit routing overhead and enhance route stability effectively.

Keywords: ad hoc networks, overhead, stability, grouping.

1 Introduction

One important character of ad hoc network is the mobility of nodes which can cause radio links to break frequently. When any link of a path breaks, this path needs to be either repaired or replaced with a newly found path. This rerouting operation degrades network performance.

To reduce rerouting operation, selecting a longtime existent path in such networks should be considered. The stability of a path can be evaluated by the lifetime of this path, and then longer lifetime means better stability. Moreover the stability of a path depends on the stability of all links constituting it. Neighbor Stability Routing (NSR)[1] algorithm selects the most historically and accumulatively stable mobile nodes to form a path between the source node and destination node. The relative stability is then propagated from the collective data by all nodes along a path. Papers[2-5] use LET[6] to assess the stability of link which is based on GPS information, and the route expiration time is the minimum LET on route. These routing protocols integrate the evaluation of LET into on-demand routing algorithms, such as DSR[7] or AODV[8], for discovering stable route. For every node can obtain GPS information

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 334–341, 2013.

by itself, no periodic message exchange needed, which can reduce much control overhead. In [9], a self-adaptive and mobility-aware path selection in mobile ad-hoc networks is proposed. To aware mobility of node, Doppler value is calculated based on the Doppler shift which can be obtained through the forwarding of route request packet like DSR for assessing the stability.

In our studies, we find that to discovery stable routes, it needs rebroadcast more RREQ by in-between nodes, which increases routing overhead and induce the broadcast storm problem[10]. To mitigate the impact of the broadcast storm problem, several broadcast schemes have been proposed. These schemes are broadly categorized into two main approaches: deterministic and probabilistic. Probabilistic approaches[11, 12] requires each node to rebroadcast the packet to its neighbors with a given forwarding probability. Deterministic approaches[13-14] in contrast, predetermine and select the neighboring nodes that forward broadcast packets.

In this paper, we propose a group-based stable routing algorithm, which aims at construct stable paths and limit routing overhead. Constructing stable route can reduce broadcast frequency and grouping can reduce the number of redundant rebroadcasts.

2 Group-Based RREQ Rebroadcast

In common on-demand routing algorithms, such as AODV and DSR etc., when some node has data to send and there isn't an available route can be used, the node (source) will broadcast a route request (RREQ) packet to its neighbors which locates in its communication radius. Other in-between nodes which receive this RREQ for the first time will rebroadcast it to their neighbors too. This rebroadcast process will terminate until the maximum hop is reached. Then it will lead to the broadcast storm, which makes the performance of ad hoc network worse.

To reduce these unexpected impacts, we propose a grouping-based route discovery for on demand routing algorithms in ad hoc networks. This scheme can group nodes according to the direction of the source node dynamic, and only the nodes in the same group are admitted to rebroadcast the RREQ received firstly.

The basic grouping rule is that, the motion direction of source is defined as a reference direction (RD), as shown in Fig. 1. Then other nodes of which directions belong to $[RD-\alpha, RD+\alpha]$ is considered to be in the same group with the source.

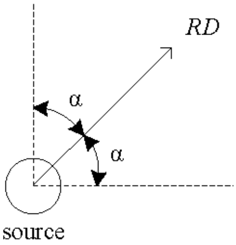


Fig. 1. Grouping rule

Based on the grouping rule, we propose a group-based RREQ rebroadcast scheme, which aims at limiting the range of broadcast. The details of this scheme are described as follow: as shown in Fig. 2, the arrow on node indicates its motion direction. The source (node 1) broadcasts a RREQ in an on-demand manner which is similar to DSR except for adding its motion information (i.e. coordinate, speed and direction) into RREQ's motion information field, which is a new field recorded in RREQ. Then the neighbors {6, 7, 8, 9} of node 1 receive this RREQ, and group themselves according to the grouping rule. Only neighbors {7, 9} upgrade motion information field with their motion information respectively and rebroadcast the RREQ, for their directions are contained in $[RD-\alpha, RD+\alpha]$. After node 7 rebroadcasts the RREQ, its neighbors {1, 4, 5, 6, 8} receive the RREQ, but only neighbors {4, 5} are the first time to receive the RREQ, moreover, only neighbor 5's direction satisfied the group rule, so only neighbor 5 upgrade and rebroadcast the RREQ. Fig. 3 gives the group-based RREQ rebroadcast algorithm.

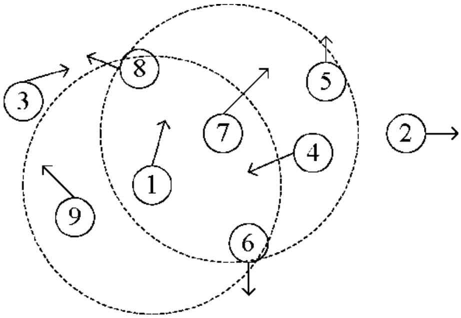


Fig. 2. A simple ad hoc network for example

```
Algorithm 1. Group-based RREQ rebroadcast
source s generates a RREQ and broadcast it
some in-between node receives this RREQ
IF the RREQ is a duplicate one THEN
        drops the RREQ
ELSE
        IF its direction satisfied the grouping rule THEN
              upgrade and rebroadcasts the RREQ
        ELSE
              drops the RREQ
        ENDIF
ENDIF
```

Fig. 3. Group-based RREQ rebroadcast algorithm

Grouping can limit the broadcast of RREQs, but it also reduces the hit rate of route discovery. So it needs a supplementary scheme: When a route discovery fails, the routing algorithm restarts a new route discovery with a bigger α_{new} , i.e. $\alpha_{new} = \alpha_{old} + \Delta\alpha$.

3 Stability Metric Evaluation

For every node is equipped with GPS or other equipments from which the motion information can be gotten, we adopt LET[6] as link stability metric to evaluate the availability of link, and take the minimum LET on path as route stability metric in this paper.

4 Algorithm Description

The proposed routing algorithm is gotten based on DSR algorithm. So we only give the route discovery of proposed algorithm in Fig. 4, which is different from DSR.

5 Simulation and Results

5.1 Simulation Environment

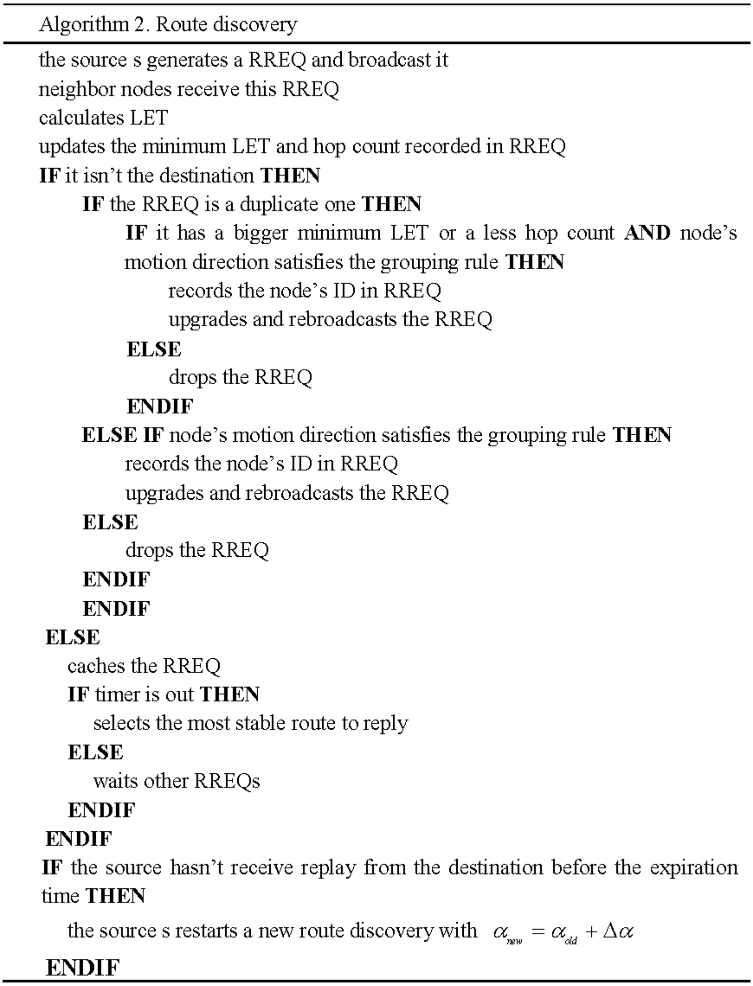
We study the proposed routing algorithm by simulation, and NS-2.29 is used here as the simulation tool.

The simulation scene is 1000m × 1000m, and 100 nodes are placed random in it. Nodes move according to RWP mobility model, of which the minimum speed is 1m/s, the maximum speed increases from 10m/s to 50m/s with the step 10m/s, and the pause time is 0s. We all build up 30 CBR flows for each simulation. The CBR packet is 512 bytes and the sending rate is 4packets/s. The simulation time is 600s.

The radio transmission model is two-ray ground reflection, and the communication radius is 250m.

5.2 Performance Metrics

1. Packet delivery ratio (PDR) is the ratio of the data received by destinations to the data sent by sources during simulation.
2. Routing overhead (RO) is the number of control packets which are used to send a CBR packet.
3. Delay is the average time which is experienced by data from source to destination.



```
Algorithm 2. Route discovery
the source s generates a RREQ and broadcast it
neighbor nodes receive this RREQ
calculates LET
updates the minimum LET and hop count recorded in RREQ
IF it isn't the destination THEN
    IF the RREQ is a duplicate one THEN
        IF it has a bigger minimum LET or a less hop count AND node's
        motion direction satisfies the grouping rule THEN
            records the node's ID in RREQ
            upgrades and rebroadcasts the RREQ
        ELSE
            drops the RREQ
        ENDIF
    ELSE IF node's motion direction satisfies the grouping rule THEN
        records the node's ID in RREQ
        upgrades and rebroadcasts the RREQ
    ELSE
        drops the RREQ
    ENDIF
    ENDIF
ELSE
  caches the RREQ
  IF timer is out THEN
    selects the most stable route to reply
  ELSE
    waits other RREQs
  ENDIF
ENDIF
IF the source hasn't receive replay from the destination before the expiration
time THEN
  the source s restarts a new route discovery with $\alpha_{new} = \alpha_{old} + \Delta\alpha$
ENDIF
```

Fig. 4. Route discovery of proposed routing algorithm

5.3 Results

We compare the performance of proposed routing algorithm, which is named Group-based for short, with DSR algorithm and ROMSP routing algorithm[15] which uses LET to establish stable route and is without routing overhead limiting scheme.

In simulations, we just simply set the origin value of α to $\pi/4$ and the value of $\Delta\alpha$ to $\pi/8$ for the proposed group-based routing algorithm. The results shown are the average value of ten simulations.

1. Packet delivery ratio (PDR)

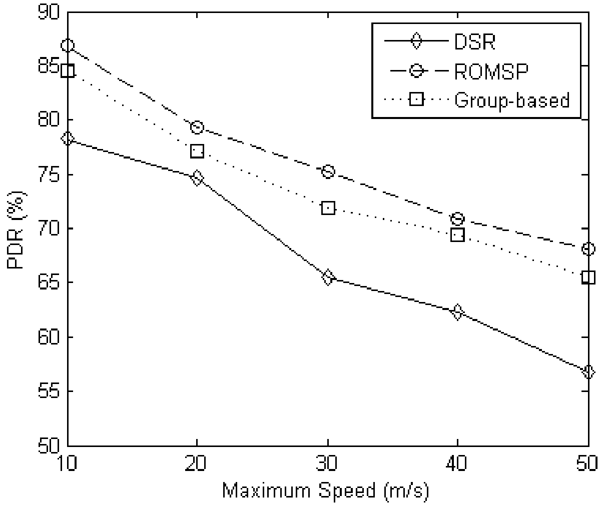


Fig. 5. PDR vs. Maximum speed

As shown in Fig. 5, The PDRs of ROMSP routing algorithm and our group-based routing algorithm are better than that of DSR algorithm. The reason is that they consider the LET of route based on which routes can be used longer than DSR. Also, we can see ROMSP algorithm is better than group-based algorithm in PDR, it illustrates that grouping may reduce the hit rate of discovering route which can lead to data loss.

2. Routing overhead (RO)

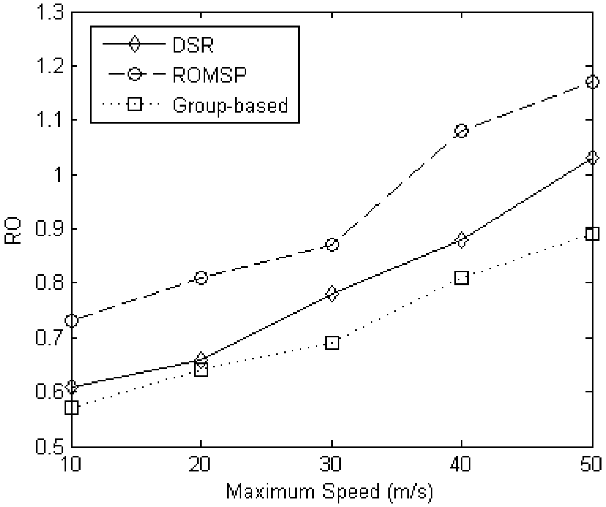


Fig. 6. RO vs. Maximum speed

As shown in Fig. 6, our group-based routing algorithm has the lowest routing overhead among three algorithms, and ROMSP routing algorithm has the highest routing overhead. The reasons are that in DSR and ROMSP routing algorithms, the RREQ will broadcast throughout the whole network. Furthermore, to discover more stable paths, in-between nodes which operate ROMSP forward more RREQs than DSR. But in group-based routing algorithm, the broadcast of RREQ is limited, i.e. only the nodes which are in the same group are allowed to rebroadcast.

3. Delay

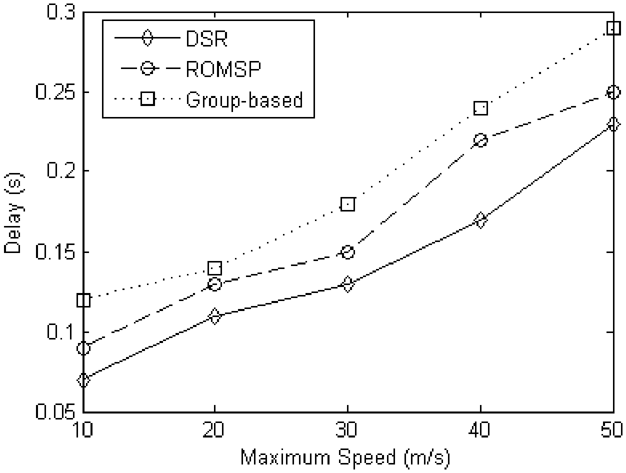


Fig. 7. Delay vs. Maximum speed

As shown in Fig. 7, DSR has the lowest delay among three algorithms, and group-based algorithm has the biggest delay, the reason is that to discover more stable routes, destination nodes must wait a few seconds. Furthermore, group-based algorithm may not discover available route successfully within one round RREQ broadcasting, so data must be cached in nodes longer before successfully establishing an available route.

6 Conclusion

In this paper, we first propose a group-based RREQ rebroadcast algorithm to limit the broadcast range of RREQs. Then we propose a routing algorithm to establish stable route based on this group-based RREQ rebroadcast algorithm. The basic idea of proposed routing algorithm is that it employs the group to limit the range of the broadcast of RREQ during the route discovery process, and then following the rebroadcast of RREQ, a stable path can also be discovered. The simulation results shows that the proposed routing algorithm can discover a more stable route with a light overhead.

Acknowledgments. This work was supported by The Central University Fundamental Research Foundation, under Grant. N110323007, N110323009.

References

1. Chen, L., Lee, C.W.: Neighbor Stability Routing in Ad Hoc Networks. In: IEEE Wireless Communications and Networking Conference 2005, vol. 4, pp. 1964–1969. IEEE Press, New York (2005)
2. Yang, P., Huang, B.: QoS Routing Protocol based on Link Stability with Dynamic Delay Prediction in Ad Hoc Networks. In: IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, pp. 515–518. IEEE Press, New York (2008)
3. Wang, N.C., Huang, Y.F., Chen, J.C.: A Stable Weight-based On-demand Routing Protocol for Mobile Ad Hoc Networks. Information Sciences 177(24), 5522–5537 (2007)
4. Taleb, T., Sakhaee, E., Jamalipour, A.: A Stable Routing Protocol to Support ITS Services in VANET Networks. IEEE Trans. Vehicular Technology 56(6), 3337–3347 (2007)
5. Wang, N.C., Chang, S.W.: A Reliable On-demand Routing Protocol for Mobile Ad Hoc Networks with Mobility Prediction. Computer Communications 29(1), 123–135 (2005)
6. Rappaport, T.S.: Wireless Communications: Principles and Practice. Prentice-Hall, Upper Saddle River (1995)
7. Johnson, D.B., Maltz, D.A., Broch, J.: DSR: The Dynamic Source Routing Protocol for Multi-hop Wireless Ad hoc Networks. Ad hoc Networking. Addison Wesley (2001)
8. Perkins, C., Belding-Royer, E., Das, S.: Ad Hoc On-Demand Distance Vector (AODV) Routing. IETF RFC 3561, http://www.ietf.org/rfc/rfc3561.txt
9. Sakhaee, E., Leibnitz, K., Wakamiya, N.: Self-Adaptive and Mobility-Aware Path Selection in Mobile Ad-Hoc Networks. In: 3rd International Conference on Bio-Inspired Models of Network, Information and Computing Sytems, vol. (41). ACM Press, New York (2008)
10. Tseng, Y.C., Ni, S.Y., Chen, Y.S.: The Broadcast Storm Problem in a Mobile Ad Hoc Network. Wireless Networks 8, 153–167 (2002)
11. Zhang, Q., Agrawal, D.P.: Dynamic Probabilistic Broadcasting in Ad Hoc Networks. Journal of Parallel and Distributed Computing 65(2), 220–233 (2005)
12. Bani-Yassein, M., Ould-Khaoua, M., Mackenzie, L.M.: Improving the Performance of Probabilistic Flooding in Ad Hoc Networks. In: International Workshop on Wireless Adhoc Networks (2005)
13. Wu, J., Dai, F.: Broadcasting in Ad Hoc Networks based on Self-pruning. In: 22nd International Conference on Computer Communications, pp. 2240–2250. IEEE Press, New York (2003)
14. Peng, W., Lu, X.: AHBP: An Efficient Broadcast Protocol for Mobile Ad Hoc Networks. Journal of Computer Science and Technology 16(2), 114–125 (2001)
15. Sakhaee, E., Taleb, T., Jamalipour, A.: A Novel Scheme to Reduce Control Overhead and Increase Link Duration in Highly Mobile Ad Hoc Networks. In: IEEE Wireless Communications and Networking Conference 2007, pp. 3972–3977. IEEE Press, New York (2007)

Path Planning in RoboCup Soccer Simulation 3D Using Evolutionary Artificial Neural Network

Saleha Raza and Sajjad Haider

Artificial Intelligence Lab, Faculty of Computer Science,
Institute of Business Administration,
Karachi, Pakistan
{saleha.raza,sajjad.haider}@khi.iba.edu.pk

Abstract. RoboCup Soccer offers a challenging platform for intelligent soccer agents to continuously perceive their environment and make smart decisions autonomously. During a soccer match, once a robot takes possession of the ball, the most important decision it has to make is to plan a route from its current location to opponents' goal. This paper presents an artificial neural network based approach for path planning. The proposed approach takes the current state of the environment as an input and provides the best path to be followed as an output. The weights of the neural network have been optimized using three computational intelligence based techniques, namely evolutionary algorithms (EA), particle swarm optimization (PSO), and artificial immune system (AIS). To assess the performance of these approaches, a baseline search mechanism has been suggested that works on discrete points in the solution space of all possible paths. The performance of the base line and the neural networks based approach(es) is compared on a synthetic dataset. The results suggest that the neural network evolved via PSO based approach performs better than the other variations of neural networks as well as the baseline approach.

Keywords: RoboCup Soccer, Artificial neural networks, Path planning, Particle Swarm Optimization, Evolutionary algorithms, Artificial immune systems.

1 Introduction

RoboCup Soccer [1] provides an exciting platform for the advancement of research in artificial intelligence and related areas with the stated goal that by the mid of 21st century, a team of fully autonomous robots would defeat humans in soccer. The competition has several leagues with separate classes of robots and the corresponding underlying challenges.

One of the most challenging problems in RoboCup Soccer is path planning that deals with the ability of an autonomous soccer agent to plan a route from its initial point to its destination without colliding with obstacles. Several efforts have been made to perform effective path planning in soccer agents. Many approaches [2-11] have addressed the problem of online path planning by modeling the environment in the form of grid maps and applying various search techniques to identify the optimal

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 342–350, 2013.

path. They, however, come across the problem of excessive online computation that may become a problem in a rapidly changing soccer environment. Similarly, the generation of sophisticated plans to be executed by the team of autonomous soccer agents is not very feasible in a soccer game where situation is being changed very rapidly and there is a limited communication mechanism among players. Calaiselv et al[5] applies genetic algorithms to determine a collision-free path between two points while Salomon et al. [4] considers the domain of RoboCup middle-sized league and uses genetic algorithm to come up with an optimal path between two points. Carpin et al. [9] presents a survey of research conducted to address the problem of motion planning in humanoid robots. However, most of the reported work, in RoboCup Soccer domain, is focused towards Simulation 2D league which does not deal with locomotion related challenges. On the contrary, the situation is much complicated in Simulation 3D league as any path planning algorithm is constrained by the locomotion skill set of the underlying humanoid robots. Different teams have to adapt their strategies in accordance with their locomotive capabilities so that they can make effective use of their strength while intelligently dealing with their weaknesses.

Traditionally, decision making in situations like these is handled by ‘if –then-else’ rules. With large number of decision variables, however, it becomes extremely challenging to come up with a comprehensive set of rules that effectively caters to all possible situations that may arise during a game. This paper presents an artificial neural network based approach to determine the best path taken by a soccer agent in RoboCup Soccer environment while maximizing possession of the ball and the chance of scoring a goal. The current state of the game is provided as an input to the neural network and the waypoint is produced as an output. The overall path starts from the current position of the soccer agent and ends at opponent’s goal via an intermediate point suggested by the neural network. Thus, determining the best path in the current context is equivalent to determining the best intermediate point between the agent’s current position and opponent’s goal. Due to continuously changing soccer environment, only one intermediate way point is considered, rather than a multi-step long-term plan. By the time agent is approaching that point, the state of the game is likely to change enough to call for re-computation of a new waypoint. Fig. 1 shows different game situations with the home player (in blue) and many opponent players (in red). The line represents the direction in which the ball should be taken.

Since the training data in the form the best path to be followed by the soccer robot in different game situations is not available, the back propagation is not a choice to train the neural network. The paper has applied the techniques of Computational Intelligence (CI) to evolve the weights of the neural network based on a given fitness function. In order to gain an insight into different classes of computational intelligence techniques and perform a comparison, the weights have been evolved using three different techniques of CI: evolutionary algorithms (EA), particle swarm optimization (PSO), and artificial immune systems (AIS).

The rest of the paper is organized as follows. Section 2 explains the presented approach of path planning using evolutionary neural networks while experimental design and results are presented in Section 3. Finally, Section 4 concludes the paper and provides directions for future research.

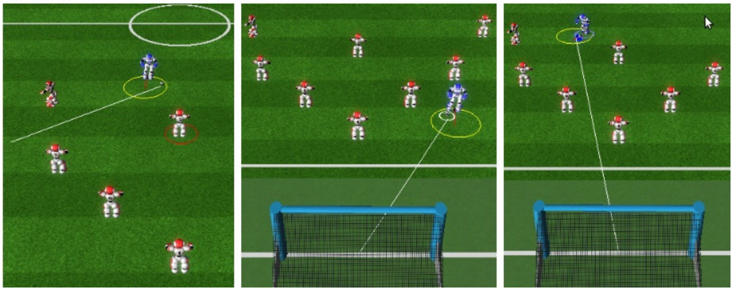


Fig. 1. Different game situations and suggested target points

2 Path Planning Using Evolved Neural Network (PPENN)

In this paper, we have used three different computational intelligence techniques, namely evolutionary algorithms, particle swarm optimization, and artificial immune systems to evolve weights of a neural network. A benchmark dataset has been generated and is used to compare the fitness of neural networks whose weights are evolved via EA, PSO and AIS. A baseline search mechanism that explores discrete points in search space has also been suggested and tested on the generated dataset. The results obtained via this baseline search mechanism are used as benchmark to compare the performance of the three neural networks mentioned above. The whole process can be divided into the several steps which are further elaborated in the subsequent sections:

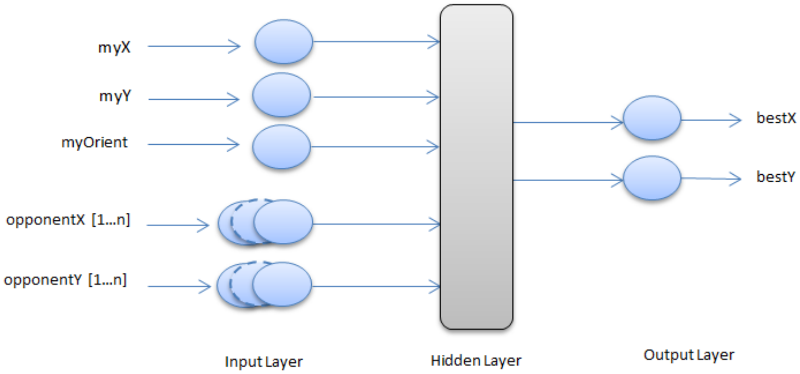


Fig. 2. Structure of neural network

2.1 Devising Network Topology

The current state of the game is provided as an input to the neural network. The game state includes position and orientation of the player in possession of the ball, positions of other teammates and opponents, velocity of players and balls. For the sake of simplicity, this paper considers the position and orientation of the player in possession of ball and the positions of opponents. Fig. 2 shows the architecture of the neural network. All input parameters are normalized and then fed to a fully connected 3-layer neural network. The output (x,y) generated by the neural network suggests the best point for the soccer agent to aim for.

2.2 Generating Baseline Data

To facilitate the evolution of neural network, a synthetic dataset has been generated for fitness computation. This dataset is formed by generating different game situations; each comprising of the position and orientation of the soccer agent in possession of the ball, and position of three opponent players closest to the ball. A baseline search mechanism has been implemented to determine a reasonably good target point in the given situation. This baseline method considers the current position of the agent possessing the ball and generates some candidate points in its vicinity at discrete intervals. The algorithm then iterates through each of the candidate point and computes their cost. The parameters used in the computation of cost function are described below:

- Distance of target point from the opponent goal
- Distance of target point from the soccer agent’s current position
- Radians to move for the soccer agent to turn towards that point
- Radians to move from the target point to the goal
- Collision with other players

The cost computation method is applied to all candidate points and the point with minimum cost is considered the best in a given situation. This method is repeated for each game situation in the baseline data and the corresponding target point along with its cost is stored in the dataset. This dataset serves as a baseline to compute the fitness of neural network during evolution. Table 1 provides a formal description of the procedure of setting up this baseline data.

Fig. 3 provides partial visualization of the search space by considering 2 variables at a time and plot the cost function on the third axis. The rest of the parameters are fixed for the purpose of analyzing a specific view of the search space. Since the goal is to minimize the cost function, we are interested in finding a local minimum.

Table 1. Setting up baseline dataset

```
Given N, X
Where N  =  no of test records, X = No. of candidate points in baseline search
Let D = baseline dataset to be generated
For i = 1 to N
    -  Randomly generate current position (D.myPosition) and orientation
       (D.Orientation)of soccer agent in the soccer field
    -  Randomly generate positions of three opponents (D.Opp1, D.Opp2,D.Opp3) in
       front of the soccer agent in a near vicinity.
    -  Generate X candidate points in the vicinity of D.myPosition
    -  Let minCost = 100000
    -  For each candidate point P ∈ X
           Cost = ComputeCost(P)
           If  Cost < minCost
               minCost = Cost, baselinePoint = P
       End For
    -  D.baseline =  baselinePoint
End For
```

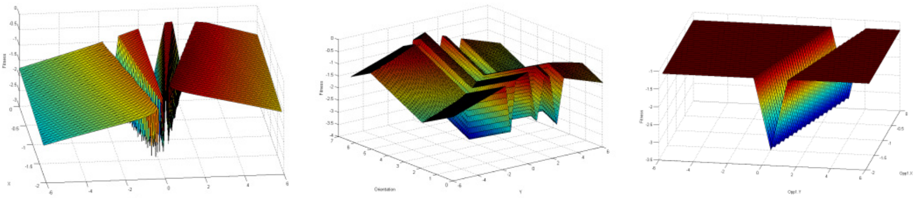


Fig. 3. a) Fitness of a point with changing X and Y positions, b) Fitness of point against varying Y and orientation of player c) Fitness of points against varying positions of opponent1

2.3 Evolving Neural Network

The weights of the neural networks are evolved via EA, PSO and AIS. There are 165 (9 x 15 + 15 x 2) weights as per the architecture shown in Fig. 2. Thus, the length of chromosomes/particles/cells is set to 165 in this experiment. During the evolution of NN, each game situation in the baseline dataset is passed through it and the cost of the output point is calculated using the method described in section 2.2. As described in Table 2, the average cost of all points in the data set is computed and is compared against the average cost obtained through the baseline approach. The difference between the two costs becomes the fitness of the corresponding individual. The process continues for a fixed number of generations while minimizing the average cost of the data set.

Table 2. Evolving ANN

```
Let N = No. of training records
Initialize the whole population with random weights.
While desired optimality criteria is not achieved
    For each individual I in population
        Simulate ANN with weights I.weights.
        totalANNCost= 0, totalbaselineCost = 0
        For each training record D in baseline dataset
            Pass D through neural network and obtain its output O.
            outputCost = ComputeCost(O)
            totalANNCost = totalANNCost + outputCost
            baselineCost = ComputeCost(D.baselinePoint)
            totalbaselineCost = totalbaselineCost + baselineCost
          End For
    End For
      I.fitness = (totalANNCost/N ) – ( totalbaselineCost/N)
      Obtain new generation through evolutionary process.
End While
Return weights of the best individual in the population.
```

2.4 Applying Evolved Neural Network

Once the evolutionary process is finished and the optimized weights are obtained, the respective neural networks are tested in a soccer match. Inputs, comprising of the current game situation, are passed through the neural network and the best path suggested by the network is followed by the soccer agents.

3 Experimentation and Results

The experiment involves evolving three neural networks via EA, PSO and AIS, respectively, using the baseline dataset and comparing their performance on the test dataset. The population, representing weights of the corresponding NN, consists of 100 individuals and the evolutionary process is continued till 1000 generations. For EA, 'binary tournament' has been used as the parent selection while 'truncation' is used as the survivor selection mechanism. Baseline dataset, comprising of 1000 different game situations, is used to compute the fitness of the individuals and the best-so-far and average values in each generation are recorded for evaluation purposes. The entire process is repeated 10 times to reduce the chance factor and to average out the average and the best-so-far fitness values. PSO and AIS have a general tendency of early and late convergence, respectively, which is also evident from the graphs shown in Fig. 4. During the training, EA has shown better convergence than the other two techniques.

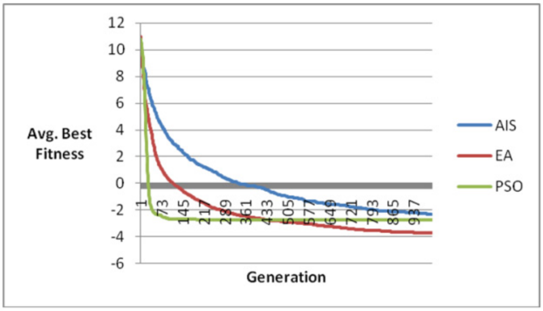


Fig. 4. Average best fitness curves for AIS, EA, and PSO

The negative fitness values indicate that the CI techniques have performed better than the baseline approach and the average cost suggested by them are less than that of the baseline search. Once a neural network has been trained, its performance is tested on another randomly generated dataset of 100 records to ensure that it has not been over-learned and is performing well on the other datasets too. Each record in the test dataset is passed through the baseline approach and through each of the evolved neural network. Evolution is performed for various numbers of generations (100, 300, 500, and 1000) and the weights obtained at the end of that generation are used to obtain target point for each test record. The difference between average cost of all the target points obtained via a neural network and those of the baseline approach are recorded and are shown in Table 3. For instance, the second row and third column has a value of -0.23 which states that the target points generated via NN have, on average, lower cost values. The average difference is -0.23 when weights of the NN were evolved for 300 generations via AIS. Similarly, the fourth row and second column has a value of -0.17. This also states that the NN performed better than the baseline approach and the difference in the average cost value over all test point is -0.17 when weights of the NN were evolved via PSO for 100 generations.

Table 3. Comparison of EA, PSO, and AIS on test dataset

No. of generations	100	300	500	1000
AIS	0.14	-0.23	-0.35	-0.43
EA	-0.006	-0.081	-0.13	-0.13
PSO	-0.17	-0.55	-0.55	-0.55

Overall, it can be seen from Table 3 that NN-PSO performed better than NN-EA, NN-AIS and the baseline approach. This is to be noted that EA, which has shown best convergence during the training, has not performed that well on the test dataset. Due to early convergence of PSO, it has obtained optimal weights even before 100 generations and is consistent afterwards. AIS and EA, on the other hand, kept improving the weights of the corresponding neural networks as the number of generations were increased. At the end of 1000 generations, however, PSO still performed better than the other two approaches.

4 Conclusion

This paper presented a neural network based approach for path planning in RoboCup Soccer Simulation 3D league. The neural networks were evolved using three different computational intelligence techniques, namely Evolutionary Algorithms (EA), Particle Swarm Optimization (PSO), and Artificial Immune System. An elementary search mechanism is implemented and is used as baseline to assess the performance of three evolved neural networks. The performances of these neural networks are tested and compared on a test dataset and all of them performed better than the baseline approach. The one evolved with PSO, however, outperformed the other two on the test data. The results suggest that machine learning techniques provide a good alternative to devise and learn optimal behavior of soccer agents which otherwise becomes too difficult to be handled using conventional if-then-else structures. Even though the focus of the paper is on RoboCup Soccer Simulation 3D league, the presented approach is extensible to the other leagues of RoboCup too. The work presented in this paper focuses on an individual player and how it can take decisions independently without coordinating the moves of other teammates. The possibility of doing a coordinated path planning by considering the positions and role of other teammates will enable a formation of players to move in harmony and is a direction of future research.

Acknowledgments. The work was partially supported by Higher Education Commission, Pakistan research grant 20-1815/RCD/105032.

References

1. RoboCup, `http://www.robocup.org/`
2. Nieuwenhuisen, M., Steffens, R., Behnke, S.: Local Multiresolution Path Planning in Soccer Games Based on Projected Intentions. In: Röfer, T., Mayer, N.M., Savage, J., Saranlı, U. (eds.) RoboCup 2011. LNCS, vol. 7416, pp. 495–506. Springer, Heidelberg (2012)
3. Behnke, S., Stückler, J.: Hierarchical Reactive Control for Humanoid Soccer Robots (2008)
4. Burchardt, H., Salomon, R.: Implementation of Path Planning using Genetic Algorithms on Mobile Robots, pp. 1831–1836 (2006)
5. Calaiselvy, C., Yong, F.T., Ping, L.W.: A genetic algorithm for robot navigation. In: Liew, K.-M., Shen, H., See, S., Cai, W. (eds.) PDCAT 2004. LNCS, vol. 3320, pp. 314–317. Springer, Heidelberg (2004)
6. Nagasaka, Y., Murakami, K., Naruse, T., Takahashi, T., Mori, Y.: Potential Field Approach to Short Term Action Planning in RoboCup F180 League. In: Stone, P., Balch, T., Kraetzschmar, G.K. (eds.) RoboCup 2000. LNCS (LNAI), vol. 2019, pp. 345–350. Springer, Heidelberg (2001)
7. Li, H., Yang, S.X., Seto, M.L.: Neural-Network-Based Path Planning for a Multirobot System With Moving Obstacles. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 39(4), 410–419 (2009)

8. Qixin, C., Yanwen, H., Jingliang, Z.: An Evolutionary Artificial Potential Field Algorithm for Dynamic Path Planning of Mobile Robot, pp. 3331–3336 (2006)
9. Carpin, S., Kallmann, M., Pagello, E.: The Challenge of Motion Planning for Soccer Playing Humanoid Robots. International Journal of Humanoid Robotics 5(3), 481 (2008)
10. Thrun, S., Burgard, W., Fox, D.: Probabilistic Robotics (September 2005)
11. Konar, A.: Computational Intelligence: Principles, Techniques and Applications, 2005th edn. Springer (2005)

Solving Hamilton Path Problem with P System

Laisheng Xiang and Jie Xue

School of Management Science and Engineering, Shandong Normal University,
Jinan 250014 Shandong, China
{xLs3366,xiaozhuzhu1113}@163.com

Abstract. P systems are biologically inspired theoretical models of distributed and parallel computing. Hamilton path is a classical NP problem, recently, there are lots of methods to solve it. Today we give a new and efficient algorithm to this classic. This paper uses the improved P system with priority and promoters/inhibitors to give an efficient solution to Hamilton path problem. We give two examples to illustrate our method's feasibility. We discuss future research problems also.

Keywords: P system, Hamilton Path, Priority, Promoter, Inhibitor, Polynomial Time.

1 Introduction

Membrane computing is a new branch of natural computing which is initiated by Păun at the end of 1998. The advantage of these methods is the huge inherent parallelism, it has drawn great attention from the scientific community so far [1]. In recent years, many different models of P systems have been proposed, such as cell-like P systems, tissue-like P systems, spiking neural P systems.

Membrane computing has been applied in many fields, such as biology, computer graphics, cryptography, etc. One of the topics in this field is the study of the computational power and efficiency of P systems. In particular, different models of these cell-like P systems have been successfully used for designing solutions to NP-complete problems in polynomial time [2].

A Hamiltonian path is a path that visits each vertex exactly once, given starting and ending vertex beforehand. The Hamiltonian path problem is to decide for any given graph with specified start and end vertices whether a Hamiltonian path exists or not. So it is a decision problem.

There have been many algorithms to deal with the Hamiltonian path problem, just like greedy algorithm, dynamic planning algorithm, divide and conquer algorithm. However, it seemed that there were any efficient methods in solving it. In the early 1970s, it was shown to be "NP complete." Until 1994, Adleman used DNA computation to find the Hamilton path in a seven vertex directed graph, which proved to be a much powerful algorithm. But there have not been anybody using other efficient algorithm--membrane computing solving this problem.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 351–359, 2013.

In this paper we present another thought on solving problems based on the cell-like P systems. The parallel ability and potential of solving combinatorial problem of P systems are employed in this study. We propose the basic idea of using P systems with promoters and inhibitors and P systems with priority to solve Hamilton path problem. We also discuss the time complexities of our new algorithm and compare it with other ways in solving Hamilton path problem.

2 Preliminaries

A cell-like P system with symport and antiport is a tuple[3][4][5]:

$$\Pi_n = \left(V, \mu, w_0, w_1 \ldots\ldots w_n, E, R_0, R_1 \ldots\ldots R_n\right)$$

(i) V is the alphabet of objects;
(ii) μ is a membrane structure with n membranes; (in the present paper we will have only nested membrane structures;)
(iii) strings w_i, $1 \le i \le n$, represent the multisets over V associated with the regions of μ;
(iv) $E \in V$ is the set of objects which are supposed to appear in the environment in arbitrarily many copies; (in one of our models the environment will be absent)
(v) R_i, $1 \le i \le m$, are finite sets of symport and antiport rules(u,out;v,in) over V associated with the membranes of μ; or rewriting rules, u→ v.
(vi) i_0 is the label of an elementary membrane of μ (the output membrane). In our models the output membrane is missing, since we read the result from the whole configuration.

2.1 P Systems with Priority and Promoters / Inhibitors

Our P system was generated by [6] and did some improvement to adapt to solve our shortest Hamilton path problems. In [7] firstly introduced the idea of sets of symport/antiport rules endowed with priority relations. Priority relations on rules have been used for P systems with rewriting rules. Throughout the rest of the paper, we use the notion of priority on rules of objects change.

Let r be a symport or an antiport rule. Let $p \in V$ be a distinguished symbol object, and $q \in V$ another one. We call p a promoter for rule r and we denote this by r|p, if rule r is active only in the presence of p. If rule r is active in the presence of p, but inactive if q is also simultaneously present, we say that q inhibitors p, and we denote this rule by $r|_p^q$.

A Simple P systems with priority and promoters / inhibitors is tuple: $\Pi_n = \left(V, \mu, w_0, w_1 \ldots\ldots w_n, E, R_1 \ldots\ldots R_n\right)$ such that: μ = [n−1…[2[1[0]0]1]2 …]n−1]is a structure of n -1 nested membranes (n ≥ 2),the innermost one labeled with 1;the objects and rules associated to each membrane are the following:

(n-1) p v_{n-1} R_{n-1} =(q,out), (n-2) v_{n-2} R_{n-2} =(p,in)>$(v_{n-1}, in; v_{n-2}, out)$ >(q,out) i=1,2,3…n-1... (j) v_j $R_j = R_{n-2}$ ...(2) v_2 $R_2 = R_{n-2}$,(1)q R_1 = {(p, in) > (q, out)|p}.

In our model we have one promoter p in the skin membrane and one inhibit q in the innermost membrane. The presence of p in membrane n-2 will activate the rule (p, in) $\in R_{n-2}$, and the rule will bring the p from n-1 into membrane n-2 ,the two integers represented in membranes n-1 and n -2 will be exchanged, the sequence in the membrane will be recorded . At the second transition, p will travel into membrane n-3, attracted by the rule (p, in) $\in R_{n-3}$. Here, the presence of p will activate the exchange rules, and the two integers represented in membranes n-2 and n -3 will be exchanged. At the next transition, p will travel into membrane n - 4 attracted by the rule(p, in) $\in R_{n-4}$. Here, it activates the communication rules which will change the integers placed in membranes n -3 and n-4. There is no q present yet, so the last rule of R_{n-2} has nothing to feed on. From now on, until p reaches the innermost membrane, things evolve in a standard manner.

Finally, at the (n-2)th transition, the promoter p enters membrane 1,it activates the rule (q, out)|p(which cannot act in p absence), and q is sent out to membrane 2.then q will go out to the n-1 membrane, and active the rule(q, out).q was sent out to the environment, the computation halted.

2.2 Sorting P System with Priority

The sorting P system with n nested membranes (n ≥ 2) are as follows[6][8]:

$$\Pi_n = (V, \mu, w_0, w_1 \ldots\ldots w_n, E, R_1 \ldots\ldots R_n)$$

such that:– μ = [n[n−1⋯[2[1]1]2 ⋯]n−1]n is a structure of n nested membranes, labeled from the innermost one towards the outermost one with 1…n; the objects and rules associated to each membrane are the following:

(1)p a^{X_1} R_1={(p, in)>(a, in; a, out)|p>(a, in)|p>(p, out)},(2) a^{X_2} R_2 ={(a, in; a, out)|p > (a, in)|p}…(2k + 1)p $a^{X_{2k+1}}$ $R_{2k+1} = R_2$,(2k + 2) $a^{X_{2k+2}}$ $R_{2k+2} = R_2$ …(n) a^{X_n} R_n =∅.

We note first that every membrane contains an integer xi represented as a^{x_i} ,for every i = 1… n. Further, there are differences between the odd and the even membranes. All the odd membranes contain a promoter p and a totally ordered set of four symport and antiport rules, R={(p, in)>(a, in; a, out)|p>(a, in)|p>(p, out)}.All the even membranes do not contain the promoter, and their set of rules is R_2 ={(a, in; a, out)|p>(a, in)|p}.The last membrane n, independently of n being odd or even, will not contain the promoter, and will have no rules, i.e., R_n =∅.In the next configuration, Since there is no promoter p in the even membranes, none of their rules will be active. In the odd membranes, the first rule which tries to act is (p, in), but since there is no p in the external membrane, the rule has nothing to feed on. The pair of rules {(a, in; a, out)|p > (a, in)|p} will become active, because of the presence of p, and thus the comparison will be at work between each odd membrane 2k+1, and its immediate upper neighbor, membrane 2k+2. Working between the two rules will order the two integers by placing the smaller one in the outer membrane, and the bigger in the inner membrane, and the configuration reached will be stable. Thus, the last rule, (p,out), will be allowed to act, and the result will be the sending of the promoter p into the outer even membrane. So, in the next configuration of the P system Π_n , the rules attached to the membranes are the same, the 'integers' might have "changed places", every consecutive pair $X_{2k+1} \leq X_{2k+2}$ being now ordered, and the promoters have travelled from the odd membranes to the even ones.

3 New Solution for the Shortest Hamilton Path Problem

3.1 Algorithm Description

In the mathematical field of graph theory, a Hamiltonian path is a path in an undirected graph that visits each vertex exactly once.(usually given the beginning and ending vertex) Determining whether such paths exist in graphs is the Hamiltonian path problem, which is NP-complete. Next, we will see that the Hamiltonian path problem can be solved in polynomial time by our P system.

Let us consider a graph G = (V,E), where V ={ V_i |1≤ i ≤ n} is the set of vertices and E ⊆ {{ V_i , V_j }|1 ≤ i < j ≤ n} is the set of edges. If { V_i , V_j } $\not\subseteq$ E then we denote V_{ij} ={λ}.

We will address the resolution via a brute force algorithm, in the framework of Change P systems with priority and promoters / inhibitors and sorting P system with priority, which consists in the following phases:

Generation Stage : Here we use the thought of cycle shift method to traverse nodes(except for v_1 and v_n) of the given graph and get all of the possible hamilton path（we assign v_1 as the first vertex and v_n is the last vertex）

Using change P systems with priority and promoters / inhibitors

$\Pi_n = (V, \mu, w_0, w_1 \ldots\ldots w_n, E, R_1 \ldots\ldots R_n)$

(n-1)p v_{n-1} R_{n-1} =(q,out), (n–2) v_{n-2} R_{n-2} =(p,in)>(v_n ,in; v_{n-1} ,out)>(q,out) i=1,2,3…n-1...(j) v_j $R_j = R_{n-2}$...(2) v_2 $R_2 = R_{n-2}$(1)q R_0 ={(p, in) > (q, out)|p}.

After all of the n nodes having done the exchange in our membrane, we read the order of all the vertex V and mark them as M_i ,using symport rule (M_i , in) to put M_i into the membrane γ which we gave before. After transitions , we get the candidate sets.

In this process ,we get 2 alphabets' full array firstly, then put them into membrane and insert the third alphabet into the skin membrane, communicate them between membranes parallel, record all the squences after every exchange. The algorithm has so strong parallelism that it can reduce time complexity.

In this stage we can also use another thought to let the traverse come true. Now we give an improved change P system to accomplish this task. For explaining clearly, we divide the P system into 2 sections.

Initial $\Pi_n = (V, \mu, w_0, w_1 \ldots\ldots w_n, E, R_1 \ldots\ldots R_n)$

Section one: (n)p q v_n R_n =∅,(n-1) v_{n-1} R_{n-1} ={(p,in)>(v_i ,in; v_{i-1} , out)|p>(q,in)|q’>(v_i ,in; v_{i-1} ,out)|q>(q'in;q,out)},(n-2) v_{n-2} R_{n-2} ={(p,in)>(v_i ,in; v_{i-1} , out)|p>(q’,out)>(q’,in)}….(j) v_j R_j = R_{n-2} …(2) v_2 R_2 = R_{n-2} ,(1)q’ v_1 $R_1 = R_{n-2}$.

Section two: (n)q v_n R_n ={(v_n ,in; v_{n-1} ,out) |p >(q,in) |p},(n-1) v_{n-1} R_{n-1} = {(v_i ,in; v_{i-1} ,out)|p > (p,out) >(q,in) >(q,out)}, (n-2) v_{n-2} R_{n-2} = R_{n-1} …(j) v_j R_j = R_{n-1} …(2) v_2 R_2 ={(v_i ,in; v_{i-1} ,out)|p >(p,out) >(q,in)>(v_i ,in; v_{i-1} , out)|q’>(q’,in; q,out)}, (1)p q’ v_1 R_1 ={(p,out)|q’>(q,in)>(q’,out)|q}

In this P system i stands for the membrane, v_i is the vertex in membrane i .we carry on doing section one and section two alternately until getting all the possible combination.

Checking Stage :we do some screening work by using rewriting rules to get all the Hamiltonian path the graph given.we design three rewriting rules Q_1, Q_2, Q_3, Q_4 to change those edges who do not belong to E ⊆ {{ V_i , V_j }|1≤i<j≤n}into λ, then change M_i which includes λ into λ

Q_1 : a $M_i \rightarrow v_1$ M_i （let M_i be the catalyst ,change a into v_1 , M_i , M_i = $v_m v_s \ldots\ldots v_p$ ） , Q_2 : v_1 M_i b→ v_1 M_i v_n , Q_3 : v_1 M_i v_n → v_1 $v_m v_s \lambda \ldots\ldots v_p v_n$, Q_4 : v_1 $v_m v_s \lambda \ldots\ldots v_p v_n$ → λ

At the end of this stage, we get all the Hamilton path in the given graph .Then, we calculate the weights of these paths and mark them as w_i .

Sorting stage: we use sorting P system with priority to sort our paths. We use $x_1 \ldots x_n$ to represent the numbers of a, and this number is also equal to w_i .

Initial $\Pi_n = (V, \mu, w_0, w_1 \ldots\ldots w_n, E, R_1 \ldots\ldots R_n)$

(1) p a^{X_1} R_1 = {(p, in) > (a, in; a, out)|p > (a, in)|p > (p, out)},(2) a^{X_2} R_2 = {(a, in; a, out)|p > (a, in)|p}...(2k + 1)p $a^{X_{2k+1}}$ R_{2k+1} = R_1 ,(2k + 2) $a^{X_{2k+2}}$ R_{2k+2} = R_2 ... (n) a^{X_n} R_n = ∅.

Output Stage: After the sorting stage, we get the orders of our paths. we know that the smallest path we wanted is in the skin membrane .

Our new algorithm reduce time complexity, we use (n+1)（n-1）/2 steps to get the candidate sets In the checking stage, we just use four steps to clear those path which are not satisfied our demand. In the end, finding the shortest path only used n steps. So we only need $\mathrm{N}=\frac{1}{2}(n^2-1)+4+n=\frac{1}{2}n^2+n+3\frac{1}{2}$ to achieve our goals. It means that we deal with this problem in polynomial time.

3.2 Examples

Example One

In this example, we will use thought of cycle shift method to get the candidate sets.

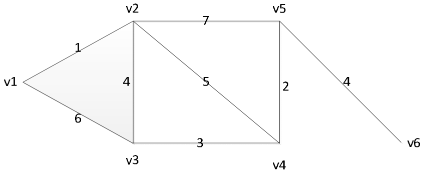


Fig. 1.

According to the graph (Fig.1), we establish a structure of six nested membranes and we give two initial string{ a_3,a_2 }and{ b_2,b_3 }.(In order to distinguish those sequences, we use different alphabet which is belong to v to represent them, but p and q work on all of them)

Generating stage:

Initial Π_6= $(\mathrm{V},\mu,\mathrm{w}_0,\mathrm{w}_1,\ \mathrm{w}_2,\mathrm{w}_3,\mathrm{E},\ \mathrm{R}_1,\mathrm{R}_2,\mathrm{R}_3,\mathrm{R}_4,\mathrm{R}_5,\mathrm{R}_6)$

(6)p, R_6 =(q,out) ,(5) a_4,b_4 R_5 =(p,in)>(v_i ,in; v_{i-1} ,out) >(q,out) i=1,2,3…n-1($v_i \neq \emptyset$), (4) a_3,b_2 R_4 = R_5 ,(3) a_2,b_3 R_3 = R_5 ,(2)∅ R_2 = R_5, (1)q R_1 = {(p, in) > (q, out)|p}.

After every exchange, we will record the new sequence, so at last, we can get string s_0={432,423,342,243,324,234}. s_0 is the full array of our nodes.

When all qs move into environment, we clear the strings in the membrane and put string which are in s_0 to P system as follows:

Second $\Pi_6=(\mathrm{V},\mu,\mathrm{w}_0,\mathrm{w}_1,\ \mathrm{w}_2,\mathrm{w}_3,\mathrm{E},\ \mathrm{R}_1,\mathrm{R}_2,\mathrm{R}_3,\mathrm{R}_4,\mathrm{R}_5,\mathrm{R}_6)$

(6)p R_6=(q, out), (5) a_5,b_5,c_5,d_5,e_5,f_5 R_5 =(p, in)> (v_i ,in; v_{i-1} ,out)>(q, out)i=1,2,3…n-1(v_i ≠∅),(4) a_4,b_4,c_3,d_2 ,e_3 ,f_2 R_4 = R_5 ,(3) a_3,b_2,c_4,d_4 ,e_2 ,f_3 R_3= R_5,(2) a_2,b_3, c_2, d_3, e_4 , f_4 R_2= R_5, (1)q R_1 = {(p, in)>(q, out)|p}.

After this step, we get s_1 ={5432, 4532, 4352, 4325, 5423, 4523, 4253,4235,5342,3542,3452, 3425, 5243, 2543, 2453, 2435, 5324, 3524, 3254,3245,5234,2534,2354,2345}, we mark them as M_1 to M_{24} and put them into membrane γ.

Checking Stage: we use rewriting rules to get this graph's Hamilton path .

Q_1 : a $M_i \rightarrow v_1$ M_i （let M_i be the catalyst ,change a into v_1 , M_i , $M_i = v_m v_s \ldots\ldots v_p$ ） , Q_2 : v_1 M_i b→ v_1 M_i v_n , Q_3 : v_1 M_i v_n → v_1 $v_m v_s \lambda \ldots\ldots v_p v_n$,Q_4 : v_1 $v_m v_s \lambda \ldots\ldots v_p v_n \rightarrow \lambda$.

Q_1 an Q_2 make M_i become $v_1 M_i v_n$ ($v_1 M_i v_n$ ={154326,145326,143526,143256,154236,145236, 142536, 142356, 153426,135426,134526, 134256, 152436, 125436, 124536, 124356, 132456,123456}) Q_3 helps us to identify paths which do not exist , Q_4 put these paths to λ. Now all the Hamilton paths we wanted are in membrane γ.(they are{ $v_1v_3v_4v_2v_5v_6$, $v_1v_3v_2v_4v_5v_6$, $v_1v_2v_3v_4v_5v_6$ })

Then, we calculate the weights of these paths and mark them as w_i : w_1=25, w_2=21, w_3=14.

Sorting stage: we use sorting P system with priority to sort our paths. we use 25,21,14 to represent the numbers of a, and this number is equal to w_1, w_2, w_3.

Π3 = (V,μ, w_1, w_2, w_3 , w_n, R_1, R_2,R_3)

(1) p a^{25} R_1 = {(p, in) > (a, in; a, out)|p > (a, in)|p > (p, out)}, (2) a^{21} R_2 ={(a, in; a, out)|p > (a, in)|p}, (3) a^{14} R_3 = ∅.

Output Stage: we know $v_1v_2v_3v_4v_5v_6$ stands for the shortest path in the skin membrane and the second short path is $v_1v_3v_2v_4v_5v_6$ in skin's neighbor membrane, the longest path is $v_1v_3v_4v_2v_5v_6$.

Example Two

Now we will use the improved change P system to get the candidate sets.

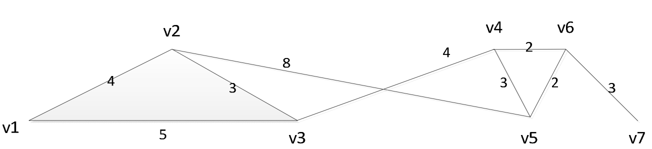


Fig. 2.

Initial Π_5 = (V, μ, w_0,w_1, w_2,w_3, w_4,E, R_1,R_2,R_3,R_4,R_5)

Section one: (5) p q v_5 R_5 =∅,(4) v_4 R_4 ={(p,in)> (v_i ,in; v_{i-1} ,out) |p>(q,in)|q'>(v_i ,in; v_{i-1} ,out)|q>(q'in;q,out) },(3) v_3 R_3 ={(p,in)> (v_i ,in; v_{i-1} ,out) |p >(q',out) >(q'in)}, (2) v_2 $R_2 = R_3$,(1)q' v_1 $R_1 = R_3$

After section one ,the string S={ $v_5v_1v_2v_4v_3$ }

Section two: (5)q v_3 R_5 ={(v_i ,in; v_{i-1} ,out)|p>(q,in) |p}, (4) v_4 R_4 ={(v_i ,in; v_{i-1} ,out) |p > (p,out) >(q,in) >(q,out)}, (3) v_2 $R_3 = R_4$,(2) v_1 R_2 ={(v_i ,in; v_{i-1} ,out) |p > (p,out) >(q,in)> (v_i ,in; v_{i-1} ,out) |q'>(q',in;q,out)}, (1) p q' v_5 R_1 ={(p,out)|q'>(q,in)>(q',out)|q}

After section two, the string S={ $v_2v_1v_4v_3v_5$ } ,then, we will go on section one and section b alternately until we get all the possible combination.

Afterwards, we will conduct the checking stage, sorting stage and output stage. At last, getting the result: the shortest Hamilton path is $v_1v_2v_3v_4v_5v_6v_7$.

4 Discussion and Conclusion

Our algorithm is different with other algorithms carrying out by silicon computers, we introduce the biology thought to find Hamilton path, we use the improved sorting P system to propose a new solution to an NP-complete problem, the shortest Hamilton path problem, According to the parallelism of membrane computing, we reduce the time complexity and get all the possible Hamilton path simultaneously. In our algorithm, we use $\mathrm{N} = \frac{1}{2}(n^2 - 1) + 4 + n = \frac{1}{2}n^2 + n + 3\frac{1}{2}$ to achieve our goals, get the shortest path. The time complexity is O(n^2).

This algorithm based on DNA computing technique has the advantages of the conventional algorithm and membrane computing. This method is suitable for the complicated graph .It also can be used as a scheme for designing solutions to other NP-complete problems from graph theory.

The main benefit of using membrane computing techniques to solve complex problems is that different possible solutions are created parallel. We gave the strategy of the new algorithm in section 3.1, In Section 3.2 we gave two instances. One is to prove the feasibility of our idea and the second is to show this algorithm is suitable for the little complicated graphs. Finally, we discuss the time of the complexities. It is found that our approach is faster than the traditional algorithms because of its parallel characters.

Acknowledgements. Research is supported by the Natural Science Foundation of China(No.61170038 and No.60873058), the Natural Science Foundation of Shandong province(No .ZR2001FM001 and No .2010RKMA2005).

References

1. Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Pérez-Jiménez, M.J., Riscos Núñez, A.: Linear–time Tissue P System Based Solutionfor the 3-coloring Problem. ENTCS 17, 181–193 (2007)
2. Pan, L., Mario, J.: Pérez-Jiménez: Computational complexity of tissue-like P systems. Journal of Complexity 26, 296–315 (2010)
3. Păun, G., Rozenberg, G., Salomaa, A.: Membrane Computing. Oxford University Press (2008)
4. Păun, G.: A quick introduction to membrane computing. The Journal of Logic and Algebraic Programming 79, 291–294 (2010)
5. P systems web page, `http://psystems.disco.unimib.it/`
6. Ceterchi, R., Martin-Vide, C.: P Systems with Communication for Static Sorting. GRLMC Report Rovira I Virgili University 101-117 (2003)
7. Păun, G.: The power of Communication: P Systems with Symport/Antiport. New Generation Computers 20(3), 295–306 (2002)
8. Arulanandham, J.J.: Implementing Bead-Sort with P Systems. In: Calude, C.S., Dinneen, M.J., Peper, F. (eds.) UMC 2002. LNCS, vol. 2509, pp. 115–125. Springer, Heidelberg (2002)

Dynamic Comprehensive Evaluation of Manufacturing Capability for a Job Shop

Huachen Liu[1,2], Sijin Xin[1,2], Wenjun Xu[1,2], and Yuanyuan Zhao[1,2]

[1] Key Lab. of Fiber Optic Sensing Technology and Information Processing
of Ministry of Education
[2] School of Information Engineering, Wuhan University of Technology,
Wuhan, 430070, China
{l.huachen,xinsj,xuwenjun,wing_zyy}@whut.edu.cn

Abstract. With the development of information technology, many new manufacturing models, e.g. virtual manufacturing, cloud manufacturing, have emerged to enhance interoperability and collaboration among the manufacturing enterprises. As a dominate factor for production performance, manufacturing capability has attract great attention and needs to be well analyzed and measured to assist the decision making in manufacturing process. A number of works have been carried out for measuring manufacturing capability. However, most of them modeled manufacturing capability in a static manner, without considering the dynamic characteristics as well as the online and real-time collected manufacturing operation data. In this paper, we analyze the components of manufacturing capability and propose a dynamic comprehensive evaluation method of manufacturing capability for a job shop, using subjective and objective analysis. A case study is presented and the results demonstrate that it is effective and flexible to evaluate the manufacturing capability for a job shop.

Keywords: manufacturing capability, job shop, dynamic evaluation, multi-criteria.

1 Introduction

With the rapid development of information technology and computer network, globalization has been a growing trend in manufacturing field. More and more products are the outcome of cooperating companies throughout the whole world. How to survive and occupy a large market share has been a problem that every manufacture must be faced with under the increasingly fierce competition. Besides, new manufacturing models, such as virtual manufacturing [1], manufacturing grid [2] have been proposed driven by advanced manufacturing technology as well as networking and information technology. Recently, a new manufacturing model - cloud manufacturing was proposed in [3]. Among the new manufacturing models, they all emphasize the cooperation and win-win situation among the participant companies. Then, how to select the best cooperative partner or manufacturing services providers (in Cloud Manufacturing) has become an important issue.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 360–367, 2013.

In the large enterprises and group companies, the manufacturing operation data can be collected on line and shared in the whole enterprise, which provides the basic data support for decision making. In order to enhance the interoperation and collaboration, the manufacturing strategy must be carried out on the basis of thorough understanding the manufacturing capability for each job shop. As a result, an effective and accurate evaluation mechanism is urgently needed in order to quantify the manufacturing capability for each job shop so as to carry out the optimal task scheduling when faced with a new manufacturing task.

Most of earlier researchers focused on how to make a comprehensive and overall assessment in a static manner, which is strongly restricted by the evaluation time. However, the manufacturing capability varies with manufacturing process and massive manufacturing operation data is accumulated as time passes. A novel dynamic comprehensive evaluation method is proposed to measure the capability under multiple criteria. Through fully analyzing the historical manufacturing operation data, we consider them with different time weight according to their influence to the present evaluation using the information entropy. Then the analytical hierarchy process (AHP) [4] is used to rank the multiple criteria for manufacturing capability for a job shop. Combining objective weighing in time scale and subjective weighting method AHP, the dynamic comprehensive evaluation can be made in measuring manufacturing capability. As a result, the manufacturing capability for a job shop with proposed evaluation mechanism can be quantified more accurate and reliable due to the fully utilizing manufacturing operation data.

2 Related Works

The research on manufacturing capability is originally attribute to skinner [5], who initially proposes the concept of manufacturing capability, and suggests that the key factors in manufacturing capability including quality, cost, delivery time. Hayes and Wheelwright [6] developed the elements related to manufacturing capability. They suggest that to build superior manufacturing capability, the quality of labor force, management, manufacturing engineering are quite essential in enterprise development. Later, Cleveland [7] raised the question that whether the manufacturing capability can be measured, and present a positive answer to it. To measure the manufacturing capability, he defined nine aspects including adaptive manufacturing, delivery performance, logistics, production, etc. Avella added the environmental protection competence in measuring manufacturing capabilities, which is consistent with the research on manufacturing objectives [8].

For the research of measuring manufacturing capability, Cleveland is the pioneer to quantify the manufacturing capability. He chose a company which assembles marine and industrial cranes to illustrate how the measuring process works in four steps. A genetic technique for evaluation of the capability of manufacturing system was proposed, using fuzzy relations and analytic hierarchy process techniques to calculate the manufacturing indices. Every research manufacturing system was quantified to a value, after which, the ranking list can be built up and the manufacturer can choose the best corporate partners [9]. With regard to comprehensive evaluation method, great process has been made in the earlier research. Data envelopment analysis (DEA) which is proposed by Farell performs well when multiple input and out exits [10]. By evaluating the decision making unit, Wei proposed that the relative efficiency can be acquired through the deviation measurement between decision making unit and data envelopment analysis [11]. These methods have achieved great success in comprehensive evaluation. However, most of them analyze and evaluate the system strongly restricted by the time scale. The manufacturing capability vary with the time and manufacturing process, and massive data could be cumulated in the development of manufacturer while real-time data can be acquired with the new technology in manufacturing. The manufacturing capability of the job shop should be evaluated dynamically so as to provide the most timely and precise information to the manager. Precise and regular evaluation of manufacturing capability for a job shop would encourage the job shop to maintain superior manufacturing performance.

3 Multiple Criteria and Modeling of Manufacturing Capability

3.1 Define the Criteria

In order to understand and measure manufacturing capability, this paper tries to make comprehensive evaluation of capability for a job shop. Using the top-down method, we consider five aspects to evaluate the manufacturing capability, and gradually develop specific criteria for each aspect. We claim that the product quality, cost control capability, delivery capability, manufacturing flexibility and environmental protection capability would affect the manufacturing capability of the job shop. Product quality is the foremost pursuit for the manufacturer in competitive market, which could help the manufacturer to establish the unique dominate position with first class products. Delivery capability and cost control capability are the powerful weapons to keep a long-time good relationship with the customers to ensure the product can be recognize by the market. To handle the variation in manufacturing process and changes in order, manufacturing flexibility is of the essence for building up superior manufacturing capability. Finally, the environmental protection capability is the needs of harmony between human and nature, which could also win the support of government and relevant policies as so to acquire greater economic support. The specific aspects and sub-criteria are shown in Table 1.

3.2 Modeling of Manufacturing Capability for a Job Shop

Let $A = \{a_1, a_2, ..., a_i, ..., a_m\}$ be the set of job shops to be evaluated in the enterprise while the criteria set that would build the manufacturing capability is marked as $\Theta = \{c_1, c_2, ..., c_j, ..., c_n\}$. $w = \{w_1, w_2, ..., w_j, ..., w_n\}$ is the weight vector for the multiple criteria. For weight vector, it should be subjected to $\sum w_j = 1 \; and \; 0 \le w_j \le 1$. $T = (t_1, t_2, ..., t_k, ..., t_K)$ stands for the time series sample, which is the historical performance in manufacturing process. $\lambda = \{\lambda_1, \lambda_2, ..., \lambda_k, ..., \lambda_K\}$ stands for the time weight, which is subject to

$\sum \lambda_k = 1 \; and \; 0 \le \lambda_k \le 1$ .From the model above, a_{ij}^k is the evaluation level of job shop a_i under the c_j criteria at evaluation time k . Then according to weighted principle, at each time series sample k , we can get the comprehensive evaluation level of job shop a_i at time k

$$D_i^k = \sum_{j=1}^{n} w_j a_{ij}^k \tag{1}$$

The comprehensive evaluation level of job shop under multiple criteria is

$$D_i = \sum_{k=1}^{K} \lambda_k D_i^k = \sum_{k=1}^{K} \sum_{j=1}^{n} \lambda_k w_j a_{ij}^k \tag{2}$$

Table 1. Specific criteria in evaluating manufacturing capability for a job shop

Factors	Specific sub-criteria for each considering aspect	
Product Quality	Performance Index	Precision machining
		Precision retaining ability
		Product reliability
		Product performance quality
		Product durability
	Nonperformance Index	Product appearance
		Product safety
Delivery	Supply speed of raw material	
	Just-in-time delivery of final product	
	Speed and reliability of processing the customer order	
	Optimal schedule in manufacturing process	
Cost Control	Product research and development cost	
	Cost summation in product manufacturing	
	Cost in product maintenance phase	
Flexibility	Product Range	
	The ability to cope with massive change product mix	
	The ability to handle changes in product volume flexibility	
	Process flexibility in manufacturing	
Environmental Protection	Green design of product	
	The material choice in production manufacturing	
	Friendliness with environment in the packing of products	

4 Dynamic Comprehensive Evaluation Method

4.1 Determine the Time Weight Vector

In dynamic evaluation of manufacturing capability, the difficulty lies in how to determine the importance of historical data and the latest acquired data and also the extreme earlier data and somewhat earlier data also should be regard discriminatingly.

Intuitively, the farther the data acquired from now, the less importance should be endowed. Let the function $\lambda(t)$ be importance degrades with time elapse, and the following properties should be possessed for $\lambda(t)$.

For $\forall\, t \in R, \lambda(t) \geq 0$ or $\exists\, t_0 \in R, \forall\, t \in R \; and \; t > t_0, \lambda(t) \geq 0$

P1. $\lambda(t) \geq 0$. P2. The first derivative of $\lambda(t)$ exist, and $\lambda'(t) \geq 0$.

The property of P1 is to ensure the weight vector to be non-negative and under the P2 condition, the time weight vector increases with the time elapse. Inspired by the information entropy, the smaller probability of the event occurs, the more information it can provide. In the determination of time vector, the newer of acquired data, the more information it can provide to evaluate the manufacturing capability, which should be considered more effective for concise evaluation.

Let H be the information influence for the present evaluation, we define the entropy as follows

$$H = (t - t_0)^{\alpha} \log_a (t - t_0) \tag{3}$$

In which, t is the evaluation time of the job shop manufacturing capability, and t_0 stands for the starting time of acquired data used to evaluate capability. α is the scale factor to adjust the significance of the recent received manufacturing operation data. The scale factor is to adjust the importance degree for recent manufacturing operation data. In evaluation, the α can be set according to the number of sample series.

In the evaluation experiment, the sample time is always discretized. Let $T = (t_1, t_2, ..., t_k, ..., t_K)$ be the time series sample, From Equ.3, for each sample time, we get $H = \{H_1, H_2, ..., H_k, ..., H_K\}$. After normalization, the jth criteria weight at sample time k is:

$$w_j^k = \frac{H_k}{\sum_{k=1}^{K} H_k} = \frac{(t_k - t_0)^{\alpha} \log_a (t_k - t_0)}{\sum_{k=1}^{K} (t_k - t_0)^{\alpha} \log_a (t_k - t_0)} \tag{4}$$

From Equ.4, the time weight vector can be computed objectively.

4.2 Determine the Weight of Multiple Criteria

In determining the weight of multiple criteria, we would follow the routine of AHP. Firstly, the standard comparison scale “criteria 1-9” is used in AHP to make the pair-wise comparison matrix $P = [p_{ij}]$, which is consistent with the suggestions $p_{ji} = {}^{1}\!/_{p_{ij}}$ proposed in [12]. Let γ be the largest eigenvalue of matrix P, follow the consistency index CI proposed by Satty $CI = (r - n)/(n - 1)$. The consistency of pare-wise comparisons can be measured. Then check whether the value of CI corresponds to acceptable degree of inconsistency using a random consistency index RI. The consistency ratio CR

can be calculated as $CR = CI / RI$. According to the maximum eigenvalue, the relative weight of each criterion is acquired and the hierarchy of the evaluation system can be decomposed step by step.

5 Case Study

In a group company of heavy machine manufacturing, the manufacturing goal of the second half year of 2012 need to be drafted according to the manufacturing operation data in the first half year. There are a number of job shops attached to the group company, and the manufacturing operation data can be shared in the whole company. Here, we choose three assembling job shops to demonstrate the proposed dynamic comprehensive evaluation method. The raw data of the three job shops has been collected analyzed by the month. And each job shop is evaluated under the multiple criteria of product quality, cost control capability, delivery capability, flexibility and environmental protection capability.

Table 2. Capability factors and their values for the three assembling job shops

Job shops	Factor	Jan.	Feb.	Mar.	Apr.	May.	Jun.
Js1	Quality	17.85	18.46	16.27	15.64	16.47	16.58
	Cost	2564	2451	2872	2698	2758	2956
	Delivery	1.65	1.68	1.72	1.63	1.56	1.52
	Flexibility	5.42	5.64	5.48	5.4	5.75	5.48
	Environment	3.65	3.58	3.68	3.57	3.52	3.54
Js2	Quality	15.96	15.69	15.38	16.52	17.35	16.34
	Cost	2687	2589	2845	2383	2598	2571
	Delivery	1.54	1.57	1.6	1.58	1.69	1.62
	Flexibility	5.48	5.71	5.64	5.62	5.68	5.71
	Environment	3.52	3.45	3.48	3.56	3.58	3.62
Js3	Quality	16.34	17.47	15.29	17.36	15.97	17.34
	Cost	2681	2734	2852	2831	2943	2896
	Delivery	1.68	1.57	1.58	1.72	1.71	1.68
	Flexibility	5.69	5.48	5.26	5.52	5.57	5.51
	Environment	3.51	3.47	3.45	3.43	3.51	3.43

Under space constraints, every criteria value is the composite result of sub-criteria using the AHP method. On the basis of metrics listed in Table 1, the AHP is used to compute the relative scores. From Equ.2, we can deduce that the two step summation is accordant with commutative law. In the following evaluation, we present the main criteria of manufacturing capability with different weight from AHP within the six month, which is shown in Table 2.

In the determination of time weight scale, from Equ.4, Let $t_0 = 0$, $\alpha = 0.4$ and logarithmic base $a = e$, we set $T = (2,3,4,5,6,7)$ considering the zero weight under $t_1 = 1$ .Then the time weight vector can be computed according to Equ.4 as shown in Table 3.

Using the AHP to determine the weight of the multiple criteria in manufacturing capability for a job shop is shown in Table 4.

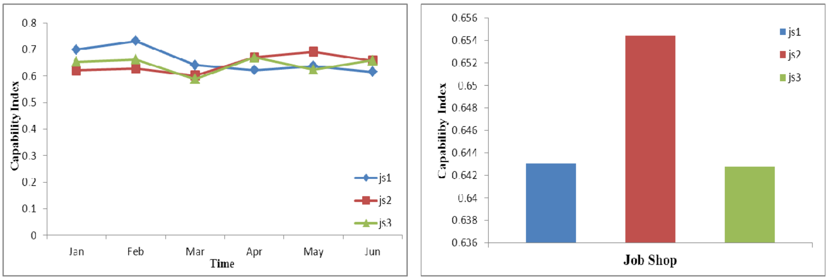
Table 3. Time weight vector in the first half year of 2012

Month	Jan	Feb.	Mar	Apr.	May	June
Weight	0.0571	0. 1065	0. 1508	0. 1914	0. 2293	0. 2648

Table 4. Relative weight of the multi-criteria in manufacturing capability

Factor	Quality	Cost	Delivery	Flexibility	Environment
Weight	0.4317	0.1940	0.1874	0.1130	0.0739

The evaluation value for each criterion in Table 2 can be classified into efficiency type index (quality, delivery, flexibility and environmental protection) and cost type index. In the normalization for the criteria we set the cost index negative and efficiency index positive. Using the tradition statistic evaluation method, the manufacturing capability of the three job shops in six month can be shown in Fig.1.a. From which, the best job shop can't be clearly figured out. In June, job shop 2 and 3 almost get the same manufacturing capability, we should not judge only by the most recent manufacturing operation data, without considering the history manufacturing operation data.



(a) (b)

Fig. 1. Comprehensive manufacturing capability for three job shops

In our proposed dynamic evaluation method, from Table 3, Table 4 and Equ.2, we can get the comprehensive evaluation value of the three job shops $D = \{0.6431, 0.6544, 0.6428\}$. And considering all the manufacturing operation data in the first half year of 2012, job shop 2 is outstanding than the other job shops as shown in Fig.1.b. We can conclude that the proposed dynamic comprehensive method is effective in evaluating manufacturing capability.

6 Conclusion

In this paper, the multiple criteria of manufacturing capability for a job shop are analyzed in terms of the review of literatures. In order to quantify the capability, a dynamic comprehensive evaluation method for measuring manufacturing capability is proposed, and we demonstrate the dynamic process with three assembling job shops to verify its application and effectiveness. Considering the historical manufacturing operation data with different time weight inspired from the entropy, we can fully utilize historical evaluation to ensure the preciseness of the evaluation, which would contribute to the task schedule and selection of manufacturing service providers. Besides, in weighting time relative importance, the scale factor can perform good flexibility for the evaluation approach.

In future, further researches can be conducted to identify the weight variation of evaluation criteria within a big data environment as well as the long time development of manufacturing process.

Acknowledgments. This work is supported by National Natural Science Foundation of China under Grant No. 51175389, Key Project of Chinese Ministry of Education under Grant No. 313042, the Ph.D. Programs Foundation of Ministry of Education of China under Grant No. 20120143110017, and the Chenguang Youth Science and Technology Development Program of Wuhan under Grant No. 201271031371.

References

1. Fumihiko, K.: Product and process modeling as a kernel for virtual manufacturing environment. Annual of the CIRP 42, 85–93 (1993)
2. Qiu, R.G.: Manufacturing grid: A next generation manufacturing. In: 2004 IEEE International Conference on Systems, Man and Cybemetics, pp. 4667–4672. IEEE Press, The Hague (2004)
3. Xun, X.: From cloud computing to cloud manufacturing. Robot. Cim-Int. Manuf. 28, 75–86 (2012)
4. Thomas, L.S.: A scaling method for priorities in hierarchical structures. J. Math. Psychol. 15, 234–281 (1977)
5. Wickham, S.: Manufacturing-missing link in corporate strategy. Harvard Business Review 6, 136–145 (1969)
6. Robert, H.H., Steven, C.W.: Restoring our competitive edge: Competing through manufacturing. Wiley, New York (1984)
7. Gary, C., Roger, G.S., John, C.A.: A theory of production comopetence. Decision Sciences 20, 655–668 (1989)
8. Lucía, A., Daniel, V.B.: The multidimensional nature of production competence and additional evidence of its impact on business performance. Int. J. of Opera. 30, 548–583 (2010)
9. Mousavi, A., Bahmanyar, M.R., Sarhadi, M., Rashidinejad, M.R.: A technique for advanced manufacturing systems capability evaluation and comparison (ACEC). Int. J. Adv. Manuf. Tech. 31, 1044–1048 (2007)
10. Farell, M.J.: The measurement of productive efficiency. J. R. Stat. Soc. A. Stat. 120, 253–281 (1957)
11. Dakua, W., Xianzhong, Z.: Rough set model and precision reduction in incomplete and fuzzy decision information system. In: 2005 IEEE International Conference on Granular Computing, pp. 278–283. IEEE Press, Beijing (2005)
12. Thomas, L.S.: Fundamentals of decision making and priority theory with the analytic hierarchy process. RWS Publishers, Pittsburgh (2000)

A Study of Aviation Swarm Convoy and Transportation Mission

Xiaolong Liang[1], Qiang Sun[2], Zhonghai Yin[2], and Yali Wang[2]

[1] Air Traffic Control and Navigation College, Air Force Engineering University, Xi'an, China
xiaolong.liang@hotmail.com
[2] Science College, Air Force Engineering University, Xi'an, China

Abstract. Aimed at the problem of the aviation swarm convoy and transportation mission, in this paper, the swarm transport path planning method and the swarm behavior control method were put forward. Taking all kinds of static constraints into account, these methods realized the path planning in the planning space, achieved the swarms' evasion to the fixed threat. The track data smooth processing was done so that it can be used in the fly. The methods of swarms' behavior control can realize the evasion to the emergent threat or obstacles and response real-timely to the coming attacker target. The simulation results show that the swarm transport path planning method and the swarm convoy behavior control method are feasible. It can meet the needs of aviation swarm convoy and transportation mission.

Keywords: swarm control, path planning, formate navigation.

1 Introduction

The future battlefield environment will be harsh, autonomous unmanned escort formation might avoid personnel's casualties. This requests escort formation to arrive the destination independently, requests to avoid the obstacle as well as to avoid the impact with own unit, and respond to the enemy's ambush. As a result of the command and control center far away from the formation, the correspondence condition could not satisfy the remote control operation, in addition, due to the operational environment indefinite result in latent threat, the formation had to overcome the influence which the individual loss brought. Therefore, the swarm cannot only depend upon the one "cerebrum" to coordinate the entire community. In order to enhance the formation the performance, the swarm must be isomerism, mixed, needs two kinds of or more than two kinds of aircraft, such as transports aircraft, escort aircraft.

As a result of the battlefield environment is constantly changing, when the aircraft according to the planed route flies, on the flying route would suddenly appear beforehand the unknown dynamic threat, this needs to consider carries on the real-time dynamic control to the flight, makes the response to the emergency situations. Based on the transportation escort mission, the article is divided into the trajectory planning, the swarm intelligence controls two levels. The path plan is the off-line plan, realization the

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 368–375, 2013.

swarm to avoid fixed threat, choice the optimal flight path; the swarm controls is the real-time distributed control, realization swarm members collision avoidance and obstacle avoidance, to make the real-time response suddenly for raids the goal.

2 Trajectory Planning

The route planning needs the overall evaluation terrain followed, the terrain avoidance and the threat evasion(TF/TA^2). In the existing planning space modeling method, uses equivalent each kind of firepower threat for the terrain threat which cannot pass through, describes the terrain and the firepower threat level sphere of action with the size different envelope circle; Use the shortest "the threat distance" between the route point and the center of the surrounding threat to evaluate the effect of threats with the route. Studies have shown that using real terrain date or firepower threat quantity increases, the killing zone is irregular, along with the enveloping circle number increase, the planning space also can appear "the dimension to explode" phenomenon.

The method of digital map visualization is presented as follows. In three-dimensional spatial planning, in order to achieve the effect of the terrain followed and the terrain avoidance, according to the following method the digital map image a grey level scope is 0~1 grayscale images: when a point height range is 0~50m, the point is transformed into a pixel which gray value is 1. The height rise 50m each time, the corresponding pixel grey level reduces 0.025, when the value surpasses 2000m, the corresponding pixel grey level is set to 0. So that we can gets a grayscale image which corresponds with the digital map.

With regard to the existence enemy firepower threat, set its cover region pixel grey level is 0. But this could brings a problem, namely, is unable to find a flight route, then according to the different threat levels, threat through degree is enhanced appropriately, reflected in the image is an appropriate increase the pixel gray value of threat corresponding to the region.

The gray image obtained by this method has 40 grayscale values, and adjacent grayscale values are equally spaced. When we carries on watershed segmentation, not only advantage to the algorithm realization, but also help to eliminate over-segmentation phenomenon.

The Planning space generation and topology identification is presented as follows. After completion of the definition of the application marked, we can carry on the segmentation to terrain image using watershed segmentation algorithm. Because when the terrain carries on image, the height value big point image into the grey level small pixel, the height value small point image into the grey level big pixel, therefore the dividing line obtained will be corresponding to the area of lower height value in the actual terrain, when the aircraft fly along the divide line, it can simultaneously realize the terrain followed and the terrain avoidance, to achieve the purpose of penetration. Planning space generation process is as shown in Figure 1.

By planning space generation can be obtained an undirected network graph of spatial planning, as a result of the terrain image edge effect influence, there have a great deal of the branching nothing to do with the planning in the obtained undirected network graph. In order to simplify the problem, first using pruning algorithm in the morphological handle irrelevant branches, it will get a new undirected network graph.

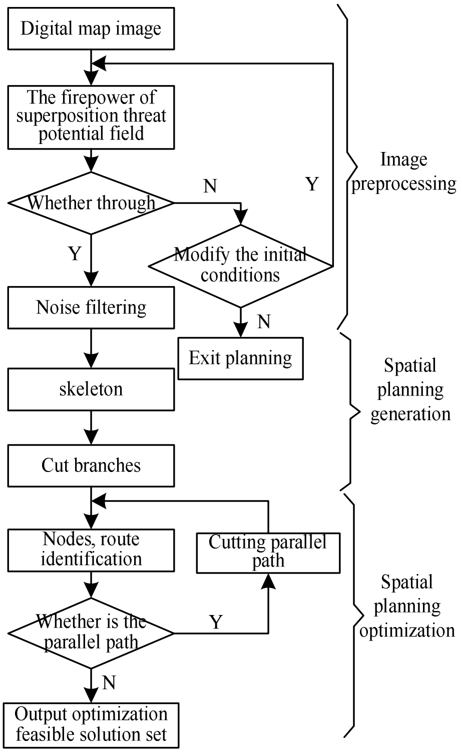

Digital map image
The firepower of superposition threat potential field
Whether through
N
Y
Modify the initial conditions
Y
Noise filtering
N
Exit planning
skeleton
Cut branches
Nodes, route identification
Cutting parallel path
Whether is the parallel path
Y
N
Output optimization feasible solution set
Image preprocessing
Spatial planning generation
Spatial planning optimization

Fig. 1. Tracks the search space optimization process

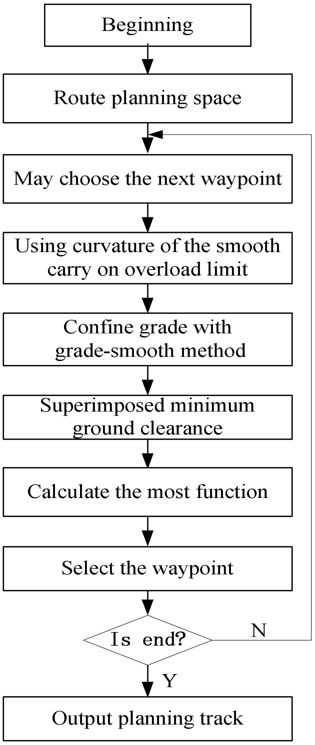

Beginning
Route planning space
May choose the next waypoint
Using curvature of the smooth carry on overload limit
Confine grade with grade-smooth method
Superimposed minimum ground clearance
Calculate the most function
Select the waypoint
Is end?
N
Y
Output planning track

Fig.2. Track planning process

On the base of pruning processing, the undirected network graph is carried on topology analysis. Take the shortest path length as the principle, cutting that may exist long parallel path between two nodes, topology picture that eventually obtained is generalized VORONOI graph. The degrees of each node are not more than 4, and the initial route follows the equivalent terrain edge change in the detail.

The Route optimization and route fitting is presented as follows. Relative to the horizontal direction, aircraft due to flight following undulating terrain induced route distance increases is much smaller, it almost negligible. Therefore when we find optimal route, from reducing the computation load point of view, route optimization only considers the distance metric in the horizontal direction, take the minimum distance in horizontal direction as the principle, the use of the A* algorithm may obtain a route. The route obtained by the A* algorithm optimization cannot carry on a route setting directly. It should consider the movement of the aircraft the maximum turning angle, the minimum step size, the maximum climb/dive angle and so on of constraints, the route which obtains is carried on the fitting.

First in the horizontal direction according to the maximum turning angle and the minimum step, as well as the surrounding terrain and the threat situation, on the route for smoothing treatment, so that the route to satisfy the aircraft maximum turning angle in the horizontal direction and the minimum step restraint.

Then, according to the aircraft maximum climb/dive angle restraint, the route which obtains in the horizontal direction fitting makes the fitting in the vertical direction. Supposing that along the route direction the height of two adjacent nodes is $h(i+1)$ and $h(i)$, respectively. The actual distance between nodes is ΔD, the maximum climb/ dive angle of aircraft is γ, according to formula (1) can be obtained the route in the vertical direction.

$$\begin{cases} h(i+1) = h(i) + \Delta D \times \tan\gamma & \dfrac{h(i+1)-h(i)}{\Delta D} > \tan\gamma \\ h(i+1) = h(i) - \Delta D \times \tan\gamma & \dfrac{h(i)-h(i+1)}{\Delta D} > \tan\gamma \end{cases} \tag{1}$$

In this way, the final route can meet the performance requirements of aircraft, in terms of the aircraft is flying.

3 Swarm Control

It is difficult to centralize control for large formation, it should be based on the different levels of the swarm system, it is need to study the method of efficient hierarchical intelligent command and control of the swarm, to realize tracking of trajectory and autonomy and intelligent of platform behavior. In the escort swarm, the individual of the swarm is decentralized control by assigning simple control commands, in the end it is integrated into the collective behavior of the whole swarm.

Let $x_i = [x_i, y_i, z_i]^T$, $v_i = [v_{xi}, v_{yi}, v_{zi}]^T$ ($i = 1, \cdots, N$)is the position and velocity of the aircraft, respectly. Then the dynamic system of aircraft are as follows:

$$[\dot{x}_i, \dot{y}_i, \dot{z}_i, \dot{v}_{xi}, \dot{v}_{yi}, \dot{v}_{zi}]^T = [v_{xi}, v_{yi}, v_{zi}, u_{xi}, u_{yi}, u_{zi}]^T \tag{2}$$

where $u_i = [u_{xi}, u_{yi}, u_{zi}]^T \in U$ is control input of the i-th aircraft, the control input consists of two basic components, namely $u_i = a_i + \alpha_i$. a_i is artificial potential of the aircraft based on the changing of the location, $\alpha_i = \frac{1}{N}\sum_{i=1}^{N} v_i$ is speed matching, They are controlled to maintain cohesion and direction of swarm.

The individual of swarm collision avoidance and keeping formation is studied firstly. Consider to the relative distance r_{ij} of the swarm between the aircraft i and the aircraft j. In order to ensure the safe of swarm flight, any aircraft in the swarm must have minimum flight safety distance r_0 . The size of r_0 depends on the type of the aircraft and motor performance. In the swarm formation flight process, in order to achieve certain tactical purposes, there requires in the formation between the aircrafts to maintain some distance. The distance between each other may be basically stable, can also be changing according to the specific circumstances, so we define the distance to the desired relative distance $r_{\exp}$. If the relative distance of aircraft within the swarm over a distance of R, it does not produce any interaction between them. Definition swarm collision avoidance and the formation maintenance behavior potential field function is as follows

$$\phi_{ij} = \begin{cases} -k_1\left\|r_{ij}\right\| + \dfrac{k_2}{\left\|r_{ij}\right\|^2} + C_1 & \left\|r_{ij}\right\| < r_0 \\ -k_3\left\|r_{ij}\right\| + C_2 & r_0 \le \left\|r_{ij}\right\| < r_{\exp} \\ k_4\left\|r_{ij}\right\| + C_3 & r_{\exp} \le \left\|r_{ij}\right\| < R \\ 0 & \left\|r_{ij}\right\| \ge R \end{cases} \tag{3}$$

This function can be used to calculate the potential field of transport unit and escort unit. Because the parameters of the dynamics performance, the sensor range and the design task performance in different types are different, then r_0 , $r_{\exp}$ and R is different respectively. Function expression means when the distance between the aircraft approaches zero, the value of the potential field will generate a near infinite repulsion, so as to ensure the flight safety of the aircraft. If the relative distance of aircraft is greater than the value of $r_{\exp}$, and not more than R, potential field will produce gravity, thus realizing swarm formation keeping.

The Obstacle avoidance problem is studied as follows.

Obstacle avoidance is concerned essentially with obstacles which are not taken into account the unexpected obstacles or missing barriers in planning to escape behavior. The function of obstacle avoidance is similar to the function of collision avoidance, shown as (4). Where the distance between aircraft and the edge of obstacle is D , R_{obs} is the range that likely counteract movement of aircraft, k_{obs} is a proportional position gain coefficient. The formula indicates when D approaches 0, the repulsion force approaching infinity.

$$\phi_{obs}(X) = \begin{cases} 0.5k_{obs}\left(\dfrac{1}{D} - \dfrac{1}{R_{rep}}\right)^2 & D \le R_{obs} \\ 0 & D > R_{obs} \end{cases} \tag{4}$$

The first class members in swarm are transport unit. From the military point of view, the task of the transport unit is only the arrival of supplies from point A to point B. But from the control point of view, the transport unit is the backbone of the swarm. The second kinds of members are escort unit, they are not trying to find the end point, but maintained around the transport unit to implement escort mission. For independent incoming attack unit, in reality, the attack unit may be artificially controlled, may also be autonomous.

Now, The transport unit describing is presented as follows. For transport unit, there are five main types of behavior to need to control, that is: along a predetermined track to the target; the collision with other transport unit; obstacle avoidance; matching the rate of the neighbor unit; escaping enemy attack unit. We define a virtual leader to achieve the track to reach the target for transport unit. The collision avoidance, obstacle avoidance and matching the rate of the neighbor unit is respectively implemented by the formula (2)~(4). If the enemy attack unit seen as a great range of obstacles, using (4) can be achieved to avoid enemy attack unit.

In order to realize transport unit flying to the target along the prescribed path, virtual leader is defined to guide the transport unit. Virtual leader, the role of the pilot, but not the actual aircraft, is a point by the command guidance center to identify. There have many methods to determine virtual leader, it can be a point in topological structure of the formation or an assumed note in formation, and it can also be the geometric center of the formation. If the location is determined with respect to the reference point in a formation, then the position of whole formation is relatively determined. Define the geometric center of transport unit formation as virtual leader, and then planning track is the flight path of virtual leader.

Take the geometric center of transport unit formation as virtual leader P_{VL} , P_{VL} is the position of the virtual leader of aircraft of the swarm, then

$$p_{VL} = [x_{VL} \quad y_{VL} \quad z_{VL}]^T = \frac{1}{n}[\sum_{i=1}^{n} x_i \quad \sum_{i=1}^{n} y_i \quad \sum_{i=1}^{n} z_i]^T \tag{5}$$

The position of transport unit P_i relative to the virtual leader P_{VL} is

$$p_i^{VL} = p_i - p_{VL} \tag{6}$$

The speed of the virtual leader is

$$\dot{p}_{VL} = [\dot{x}_{VL} \quad \dot{y}_{VL} \quad \dot{z}_{VL}]^T = \frac{1}{n}\left[\sum_{i=1}^{n} \dot{x}_i \quad \sum_{i=1}^{n} \dot{y}_i \quad \sum_{i=1}^{n} \dot{z}_i\right]^T \tag{7}$$

The virtual leader is a guide of movement of transport swarm, but also is a reflection of the transport swarm as a whole movement. Command control center guide virtual leader, according to the relative position, the transport unit can be controlled autonomously to realize following. Transport unit control law is

$$U_i^S = \underbrace{\phi_{goal}(x_i, y_i)\Big|_{(a,b)}}_{goal \quad potenial} + \underbrace{\sum_{j=2}^{N_s} \nabla \phi_{i,j}}_{\substack{collision \\ avoidance: \\ suppply \quad units \\ (excluding \quad self)}} + \underbrace{\sum_{k=1}^{N_a} \nabla \phi_{i,k}}_{\substack{attacker \\ avoidance}} + \underbrace{\sum_{m=1}^{N_o} \nabla \phi_{obst_{i,m}}}_{\substack{obstacle \\ avoidance}} + \underbrace{\left(\frac{1}{N_s}\right) \sum_{n=1}^{N_s} v_n}_{\substack{supply \quad units \\ velocity \quad matching}} \tag{8}$$

Where (a,b) is target location, N_s is the number of the transport unit, N_d is the number of the escort unit, N_a is the number of the attack unit, N_o is the number of the obstacles.

The escort unit describing is presented as follows. For escort unit, there are five main types of behavior to need to control, that is: the goal of the continuous flight changes; the collision avoidance with other transport unit and escort unit; obstacle avoidance; matching the rate of the neighbor unit; intercepting enemy attack unit. Regarding the escort unit flying position should be in periphery of the entire swarm, the incoming enemy units is local target of escort unit, location of the escort unit should in the transport unit and the enemy attack unit. The collision avoidance, obstacle avoidance and matching the rate of the neighbor unit is respectively implemented by the formula (2)~(4). Escort unit control law is as follows:

$$U_i^D = \underbrace{\phi_{goal}(x_i, y_i)\Big|_{(a,b)}}_{goal \quad potenial} + \underbrace{\sum_{j=1}^{N_s} \nabla \phi_{i,j}}_{\substack{collision \\ avoidance: \\ suppply \quad units}} + \underbrace{\sum_{k=2}^{N_d} \nabla \phi_{i,k}}_{\substack{collision \\ avoidance: \\ defender \quad units \\ (excluding \quad self)}} + \underbrace{\sum_{m=1}^{N_o} \nabla \phi_{obst_{i,m}}}_{\substack{obstacle \\ avoidance}} - \underbrace{\sum_{n=1}^{N_a} \nabla \phi_{i,n}}_{\substack{attacker \\ intercept}} + \underbrace{\left(\frac{1}{N_s}\right) \sum_{p=1}^{N_s} v_p}_{\substack{supply \quad units \\ velocity \quad matching}} \tag{9}$$

where $(a,b)=\left\|\left(\frac{1}{N_s}\right)\sum_{j=1}^{N_s}(x_j,y_j)-\left(\frac{1}{N_a}\right)\sum_{m=1}^{N_a}(x_m,y_m)\right\|$.

The attack unit describing is presented as follows. For attack unit, using a simple control law guide attack unit to the swarm center, attack interception the transport unit, there are five main types of behavior needing to control, that is: flying to the target (the central of the transport unit), collision avoidance, matching the speed of the transport unit in the sensor range. Control law of attack unit is as follows:

$$U_i^A = \underbrace{\phi_{goal}(x_i,y_i)\Big|_{(a,b)}}_{goal\ \ potenial} + \underbrace{\sum_{j=1}^{N_o}\nabla\phi_{obst_{i,j}}}_{\substack{obstacle\\ avoidance}} + \underbrace{\left(\frac{1}{N_s}\right)\sum_{k=1}^{N_s} v_k}_{\substack{supply\ \ units\\ velocity\ \ matching}} \tag{10}$$

4 Simulation

Simulation scene: the maps of combat area is N33E096, N33E097, N33E098, N34E096, N34E097, N34E098, N35E096, N35E097, N35E098 and so on, all of these are nine digital map, the "N" indicates north latitude, and "E" represents the longitude. The actual distance is represented by unit grid is $500m\times500m$ .At N34.4E97.5 in this area is enemy ground surveillance radar. The height of the radar is $100m$, the detection radius of the radar is $100km$. Set the starting position is A (4,254), the target position is B (275,110), the climb/dive angle α of the aircraft is 30°, the curvature is limited to $\rho_v\in[-2.4\times10^{-4},3.4\times10^{-4}]$,the minimum longitudinal vertical distance from the ground is 50m. Set the number of transport units $N_s=3$, the number of escort unit $N_d=5$, the number of unpredictable attack unit $N_a=3$, the unpredictable obstacle quantity $N_o=3$. Figure 3 is a projection of track on the ground; we can be seen the programming track can be good for threat avoidance from the graph. Figure 4 is 3d track, the track is good for the terrain avoidance and satisfy the constraint conditions of the three dimensional track. Figure 5 is sketch map of the swarm to avoid unexpected obstacles and we can be seen that the escort fleet can be better to avoid sudden threat. Figure 6 is the schematic diagram for the escort unit guarding behavior.

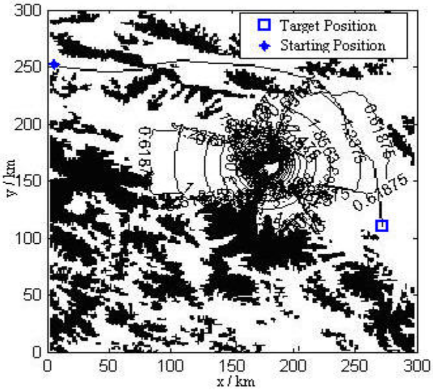


Fig. 3. The horizontal track diagram

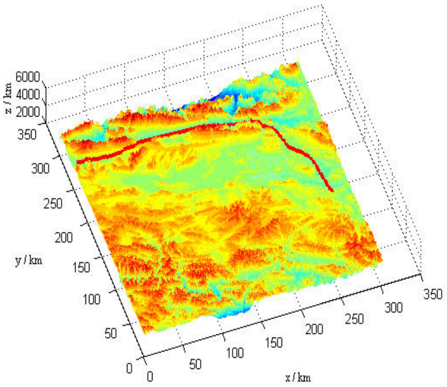


Fig.4. 3-D track diagram

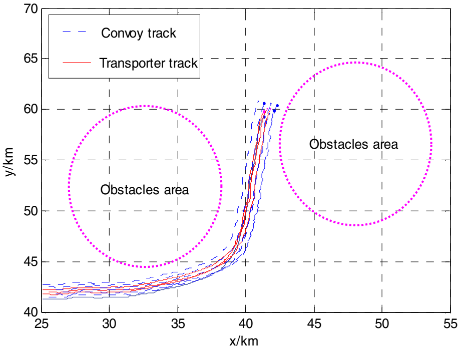


Fig. 5. Swarm escort formation avoid threat

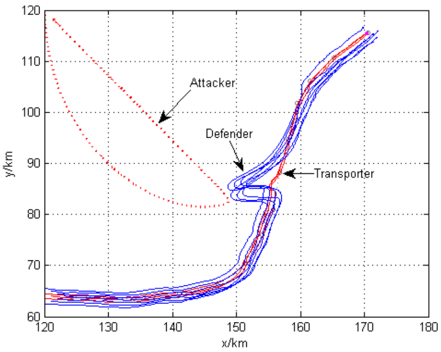


Fig.6. The swarm formations escort

5 Conclusion

Based on the swarm transport protection problems, we put forward the route optimization method for static threat avoidance, using the method of swarm control to avoid real-time obstacle, keep the swarm formation rank, and in view of the incoming target, escort unit be able to convoy to transport units. The simulation results show that we can achieve the result of complex system control expected by control simple swarm behavior of the individual.

References

1. Rafael, C.G., Richard, E.W.: Digital Image Processing, 3rd edn. Prentice Hall (2008)
2. Wu, X., Luo, X.: The Algorithm for Creating Weighted Voronoi Diagrams based on Cellular Automata. In: Proceedings of the 6th World Congress on Intelligent Control and Automation, Dalian, China, June 21-23, pp. 4630–4632 (2006)
3. Barnes, L., Alvis, W., Fields, M., Valavanis, K., Moreno, W.: Heterogeneous Swarm Formation Control Using Bivariate Normal Functions to Generate Potential Fields. In: Proceedings of the 2006 IEEE Workshop on Collective Intelligence and its Applications, pp. 85–94 (June 2006)
4. Yu, W., Chen, G., Cao, M.: Distributed leader-follower flocking control for multi-agent dynamical systems with time-varying velocities. Systems & Control Letters 59(9), 543–552 (2010)
5. Tanner, H.G., Christodoulakis, D.K.: Decentralized cooperative control of heterogeneous vehicle groups. Robot. Auton. Syst. 55, 811–823 (2007)
6. Saber, R.O.: Flocking for multi-agent dynamic systems: Algorithms and theory. IEEE Transactions on Automatic Control 51(3), 401–420 (2006)
7. Zhang, T., Zhu, Y., Song, J.: Real-time motion planning for mobile robots by means of artificial potential field method in unknown environment. Industrial Robot: An International Journal 37(4), 384–400 (2010)

A Multiple Interfaces and Multiple Services Residential Gateway Scheme

Wenyao Yan[1], Zhixiao Wang[2,3], Kewang Zhang[2], Junhuai Li[3], and Deyun Zhang[2]

[1] Xi'an Innovation College, Yan'an University, 710010 Xi'an, China
[2] Xi'an Jiaotong University, 710049 Xi'an, China
[3] Xi'an University of Technology, 710048 Xi'an, China
yanwenyao05@yahoo.com.cn, maplewzx@163.com, kewwang@xjtu.edu.cn, lijunhuai@xaut.edu.cn, dyzhang@xanet.edu.cn

Abstract. Residential gateways that interconnect home networks, pubic networks, and smart household devices play a critical role in intelligent home systems. However, the existing gateways could hardly be adapted to emerging multiple access methods and the multiple services' requirements for future home intelligent environments. This paper introduces the work in progress in constructing a stable and efficient Broadband Multiple Modes Residential Gateway (BMMRG) which supports multiple access interfaces, multiple services, IPv6, security, QoS and remotely web management. It is mainly based on an IXP425 network processor and a Linux kernel. We first present the hardware and software architectures of BMMRG, and then we introduce their in-detailed implementations. In the meantime, an intelligent home system is proposed based on BMMRG and household appliances equipped with wireless and ZigBee adapters. Finally, the effectiveness and feasibility of BMMRG is verified through testing.

Keywords: Residential Gateway, Multiple Modes, Multiple Interfaces, Multiple Services.

1 Introduction

With the advancements in such areas as sensing techniques, embedded computing techniques, distributed information processing techniques and wireless communication techniques, intelligent environments [1-3] have gained unprecedented consideration from wider research communities, and are also coming gradually to reality. One of the important and practical fields in intelligent environments is smart home networks where all sorts of intelligent devices and sensors are continuously working to make inhabitants' daily lives more convenient and comfortable. Some of the latest technologies, services and applications in home networks are discussed in [4-6]. In the meantime, in order to realize effective information exchange not only among household devices but also between home networks and outer networks such as the Internet, the residential gateway (also called home gateway) which is one of the significant components of networks, is being explored and attempted to be used actively.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 376–384, 2013.

Earlier, the gateway could be generally described through different devices, such as cable modem, xDSL modem, wireless router, network switch, voice over internet protocol (VoIP) analog telephony adapter, wireless access point, or any combination of the above. However, these gateways could not be adapted to emerging requirements for future intelligent home environments. The trend of next generation gateway will be provided through more powerful functions, more abundant and user-friendly interfaces. What's more, it will be a manageable terminal device that supports automatic or through the web configurations, remote control, multiple services and provides Quality of Service (QoS) to simultaneously support different types of services [7].

This paper proposes an efficient and stable Broadband Multiple Modes Residential Gateway (BMMRG) for a smart home automation system which offers multiple access interfaces, multiple services, security, QoS and remote web management.

The remainder of this paper is organized as follows. In Section 2, a brief review on previous works is shown. In Section 3, we report on the hardware and software architectures of BMMRG. In Section 4, we discuss the implementation of BMMRG which both adopts division and rule methods. And then the performance of BMMRG is evaluated in Section 5. Finally, Section 6 concludes the whole work.

2 Related Works

Currently, more focus is being given to prevalent intelligent residential gateways in home intelligent networks. In [8], the real-time software framework for a gateway set-top-box that serves as intermediary connecting home networks to Internet is proposed. Similarly, for the purpose of interconnecting all kinds of heterogeneous networks, an embedded gateway software framework is also designed in [9]. In [10], an enhanced OSGi (Open Service Gateway Initiative) service gateway architecture is demonstrated, which integrates many existing wired and wireless protocols and standards in home networks. In [11], a reasoning-based Domotic OSGi Gateway is proposed which not only supports heterogeneous devices interconnection but provides different domestic networks cooperation. In [12], in order to deal with stream media services a distributed real-time OSGi gateway framework is studied. In [13], the partial RGs are overviewed. At the same time, a novel OSGi Multi-Residential Gateway capable of managing several residential units (e.g. homes, flats) is explored. Herein the modular firewall and QoS functions in MRG are particularly emphasized. In [14], a programmable mobile gateway that offers Zigbee and Bluetooth technologies and supports remote access is developed and implemented.

Moreover, the partial smart home automation systems relies on various smart gateways have been also quested. In [15], a peer-to-peer service-oriented architecture for smart-home environments is suggested, to deal with the problems caused by traditional architectures, to solve the dynamic environment, and to provide appropriate services, which relies on multiple OSGi and mobile-agent technology. In [16], an intelligent home control system with smart alarm clock is probed and performed, which is composed of an internet access point and information acquisition module, in-house networking services with blue-tooth wireless connectivity, an information fusion based controller using fuzzy logic and fuzzy neural network, and embedded computational units. In [17], another one home automation system which integrates a common gateway, Zigbee and Wi-Fi is also studied.

In addition, for the purpose of saving energy, a low-cost energy information gateway and an intelligent home electricity system are also respectively proposed in [18] and [19].

However, most of the aforementioned gateways provide simple function, single or limited interconnection interfaces, and hardly support multi-service and multi-interface types. Therefore, a novel multi-mode intelligent access terminal should be probed. Actually, some devices manufactures or institutes are focusing on the next generation intelligent gateways or home intelligent networks. For example, MUSE [20] and Figaro [21] projects are sponsored by EU (European Union).

In this paper, we also concentrate on designing and implementing an efficient and stable Broadband Multiple Modes Residential Gateway (BMMRG) for a smart home automation system. The proposed gateway and home intelligent network system offer multi-service and modish wired or wireless interface types, and can also manage intelligent household appliances through home web portal.

3 Architecture of the Broadband Multiple Modes Residential Gateway

3.1 Hardware Architecture

Some of the main goals of an intelligent gateway are to make inhabitants access in-house and outer networks more convenient, to use and manage different home electronic devices simpler and easier, and to interconnect and exchange various information with WAN and LAN, especially Internet and future network household appliances. Therefore, BMMRG should support and utilize many kinds of access methods and deal with different types of network services.

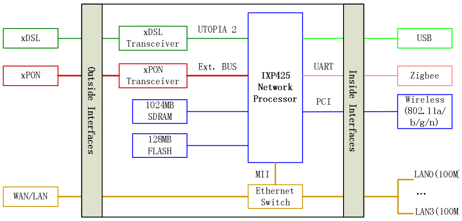


Fig. 1. Hardware overall framework

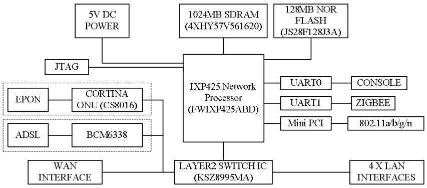


Fig. 2. Hardware detailed architecture

The hardware overall framework of BMMRG is illustrated in Fig. 1, which consists of a core control chip (Intel IXP425 network processor), SDRAM & Flash, Ethernet interface, xPON interface, xDSL interface, USB interface and wireless interface, etc. Furthermore, the hardware detailed architecture and the relevant IC chips selected are given in Fig. 2. And in Table 1, we show the different inside and outside interfaces.

Table 1. Interfaces of BMMRG

	LAN Interfaces	WAN Interfaces
Outside Interfaces		• XDSL • Wired Ethernet • EPON
Inside Interfaces	• Wireless (802.11a/b/g/n) • Wired Ethernet • USB • ZigBee	

3.2 Software Architecture

The software architecture of BMMRG is shown in Fig. 3, which is made up of a Boot Loader layer, the embedded RTOS layer, middleware layer and application layer. Redboot is used as init bootstrap in the Boot Loader layer. The uClinux 2.4.8 and devices drivers are included in embedded layer, since Linux is a free, simple, stable, robust, secure and pruned Open Source system. JVM is laid in the layer 3. In the highest layer, OSGi based Linux platform is an important component, in which a huge amount of services would be developed and distributed as bundles, as well as several typical services and applications, embedded browser and other end-users applications.

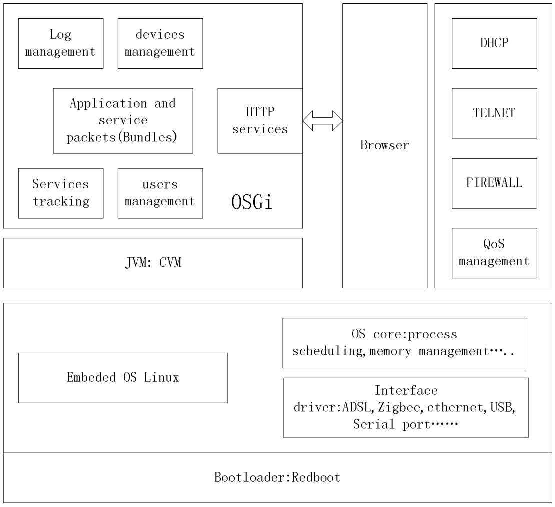


Fig. 3. Software architecture

4 Implementation of the Broadband Multiple Modes Residential Gateway

In this section, we show the implementation of BMMRG based on the Intel IXP425 Network Processor (NP) and uClinux. Hereinto, the Intel IXP425 network processor combines integration with support for multiple WAN and LAN technologies in a common architecture to meet requirements for high-end gateways, Voice over IP (VoIP) applications, wireless access points, SME routers, switches, security devices, (DSLAMs), ADSL line cards, industrial control and networked imaging applications.

The uClinux is configured on IXP425 NP platform to manage various hardware devices, which not only offers a large number of Open Source Code programs, libraries and tools, but also supports strongly the TCP/IP protocol stack.

4.1 Implementation of Hardware of BMMRG

On the one hand, taking into account the high-speed PCB design, the quality of analog signals is more easily suffered interference from power, the parameters of external components and PCB traces. The BMMRG design should minimize PCB system noise and crosstalk. On the other hand, for the purpose of a more convenient implementation and maintenance, we adopt the Divide and Rule scheme for the BMMRG hardware implementation. The core module based on the key component——IXP425 that is shown in Fig. 2, EPON module shown in Fig. 4, ADSL module shown in Fig. 5, and wireless and Zigbee adapter module shown in Fig. 6 which will be equipped in each household appliance are fulfilled respectively.

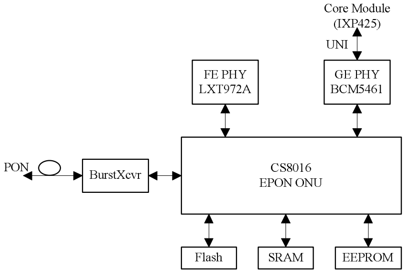


Fig. 4. EPON module

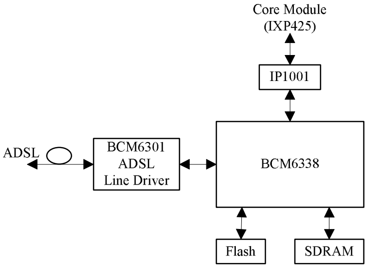


Fig. 5. ADSL module

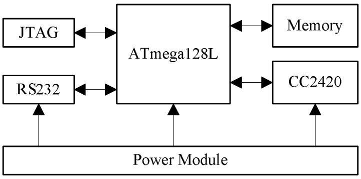


Fig. 6. Wireless and Zigbee Module

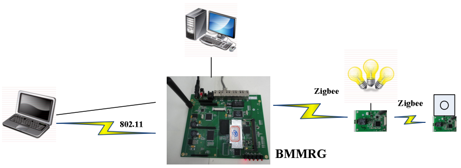


Fig. 7. Evaluation environment

4.2 Implementation of Software of BMMRG

In this subsection, Linux kernel is freely pruned for supporting the TCP/IP protocol, devices drivers and other services and applications through using library functions derived from the Access Library. In addition, JVM (Java Virtual Machine) and OSGi framework further offer platform-independent, heterogeneous services and applications.

Linux/Bootloader. SnapGear uCLinux2.4 and other tools, such as binutils-2.16, gcc-3.4.4 and glibc-2.3.3, are implemented in BMMRG. On the one hand, lots of function operations such as firewall security devices, etc. could be configured. On the other hand, Linux kernel could be built in BMMRG by means of Intel Access Library Snapgear-modules, Ethernet driver and tool chains to cross compile. Other utilized softwares include: Redboot 2.0 that functions as bootloader, Minicom that functions as terminal program to configure Redboot, DHCP to assign an IP address to BMMRG, and TFTP as a transport tool to load zImage and ramdisk.gz to BMMRG. During boot period, IxNpeDl function is called to load the corresponding code image into the instruction cache of each NPE. Next, two functions, IxQmgr and IxNpeMh, are invited to initialize the queue management; in the meantime the message handler is in charge of the communications between NPEs and Xscale.

Devices Drivers. For the purpose of good communication between the embedded Linux and different devices, a huge number of devices drivers such as NPEA/B/C, MMU, UART0/1, Intel FLASH, Intel PRO 2200BG wireless network card, etc. are developed. This effort could be aided and implemented by the Access Library. Besides devices drivers, two OS dependent modules, namely OSSL and IxOSServices, in Access Library should be suitably modified in order to guarantee appropriate OS-related function operations.

Middleware and Multiple Services. In order to communicate and interoperate between BMMRG and home devices, some middleware could be employed such as UPnP, JVM, etc. technologies. For instance, Voice over IP is implemented by way of Java embedded technology. Multimedia services are also achieved via UPnP AV architecture [22] and OSGi platform.

OSGi Platform. OSGi is one of the key components of the BMMRG software because by way of OSGi the remote service providers can dynamically and handily install, activate, deactivate, update, and remove services bundles via WAN such as the Internet. It also offers the interconnection between the smart home systems and WAN.

Security/QoS. Considering various kinds of information, such as voice, video, pictures, text, control messages etc., different users need different access levels of security and different priorities. In BMMRG, system security and QoS management are relied on differentiating different traffic services. Meanwhile, UPnP-QoS [23] framework is used to guarantee the reliability between BMMRG and home digital devices.

In addition, MAC-related operations and the IPSec process are fulfilled through library functions such as IxEthAcc, IxEthDB, and IxCryptoAcc APIs.

Finally, a light-weighed web server is implemented and configured in BMMRG for conveniently controlling and managing. As a result, remote consumers can control BMMRG and manage smart home system easily.

5 Evaluation

The main purpose of this section is to verify that the BMMRG in an intelligent home system is feasible and effective. The evaluation environment consists of the proposed BMMRG, a laptop with wired and wireless network adapters, a light and a switch with ZigBee module controllers, which is shown in Fig. 7. The system was performed to a cycle of strenuous operations to simulate a high level of daily usage. The light was changed 30 times using the ZigBee module controller directly and 30 times using remote web control indirectly through wired and wireless LAN. The experiments showed the correct functionality and performance of the system 100% of the time. Table 2 displays the average access delay of the requested change using direct and indirect control.

Table 2. Average Access Delay Comparison

Direct Access	Indirect Access: Wired	Indirect Access: Wireless
758ms	1.4s	2.1s

As Table 2 presents, the average delay was longer for indirect web access than for the direct ZigBee controller. However, the ZigBee controller also had an average control delay of 758ms. This implies that the majority of the control delay exists in the switch and electromagnetic relay actuation and subsequent bulb state change. Moreover, the main reason of the greater indirect control delay not only lies in the aforesaid factors but also includes the request and response time of the web server.

Secondly, the capability of packet forwarding is determined by measuring the achievable throughput by sending packets from one network to the other network via the BMMRG. In our experiment, the packet length was also varied from 64, 128, 256, 512, 1024, 1280 to 1518 bytes. The measurement of throughput was taken for 120 minutes.

Fig. 8 shows the throughput results with respect to different size packets when there was one connection. We observed that the throughput of BMMRG is maximal when the packet size is 1024B, rather than other larger sizes. This is because the longer processing time of larger packets offsets the benefit from their reduced header processing overhead. Fig. 9 represents the throughput of BMMRG when there was one connection and the packet size was fixed to 1024B. Although the fluctuation exists because of lightly congestion, the overall is almost steady and the average throughput is above 92.4 Mbps. This implies that the performance of BMMRG is competitive during the run.

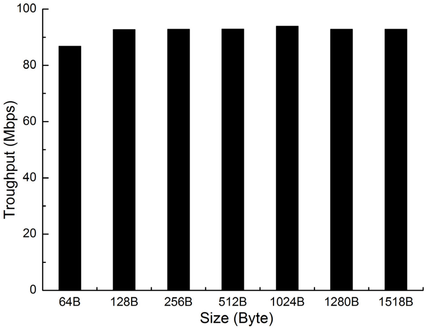


Fig. 8. Throughput of one connection

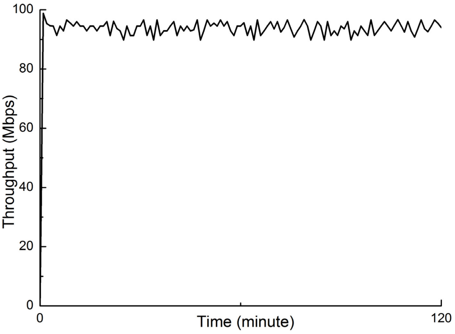


Fig. 9. Throughput with packet of 1024B

6 Conclusion

In this paper, a Broadband Multiple Modes Residential Gateway, namely BMMRG, has been designed and implemented based on IXP425 network processor and Linux kernel. The outstanding performance of the IXP425 platform and the open source and pruned characteristic of Linux makes our solution an attractive solution to be produced, tried out and commercialized will be done in the future.

We firstly have represented the architectures of hardware and software in BMMRG, and then respectively introduced their in-detailed implementations. Finally, a smart home system has been presented based on BMMRG, and wireless and Zigbee adapters implemented in the household appliances. The BMMRG has shown excellent features, such as competitive processing and access performance to deal with multiple access and multiple services, home PC independence, abundant wired and wireless interfaces, flexible and open platform architecture, IPv4/IPv6, security and QoS, as well as remotely managing functionality using the web. Meanwhile, an intelligent home network that has been built through using the proposed BMMRG has formed a prototype of IoT in a home, which will play a certain positive role in the development and research of IoT.

Acknowledgments. This work is partially supported by China CNGI Grant No. CNGI-09-02-07, NSFC Grant No. 61172018, Education Department of Shaanxi Grant No. 2013JK1120 and Beilin District 2012 High-tech Plan, Xi'an, China.

References

1. Augusto, J.C., Nakashima, H., Aghajan, H.: Ambient Intelligence and Smart Environments: A State of the Art. Springer (2010)
2. Bose, R.: Sensor Networks Motes, Smart Spaces, and Beyond. IEEE Pervasive Computing 8, 84–90 (2009)
3. Cook, D., Das, S.: Smart Environments Technology, Protocols and Applications. John Wiley & Sons, Inc. (2004)
4. Tsutsui, A.: Latest Trends in Home Networking Technologies. IEICE Transactions on Communications E91-B, 2470–2476 (2008)

5. Bakht, H., Merabti, M., Askwith, B.: Home Networking. In: Sloane, A. (ed.) Home-Oriented Informatics and Telematics. IFIP, vol. 178, pp. 311–322. Springer, Boston (2005)
6. Stefanov, D.H., Zeungnam, B., Won-Chul, B.: The smart house for older persons and persons with physical disabilities: Structure, technology arrangements, and perspectives. IEEE Transactions on Neural Systems and Rehabilitation Engineering 12, 228–250 (2004)
7. Residential Gateway, `http://en.wikipedia.org/wiki/Residential_gateway`
8. Jae-Chul, M., Hyo-Sang, L., Soon-Ju, K.: Real-time event kernel architecture for home-network gateway set-top-box (HNGS). IEEE Transactions on Consumer Electronics 45, 488–495 (1999)
9. Xie, X., Deng, D., Deng, X.: Design of embedded gateway software framework for heterogeneous networks interconnection. In: Processing of International Conference on Electronics and Optoelectronics (ICEOE), pp. 306–309. IEEE Press, Dalian (2011)
10. Valtchev, D., Frankov, I.: Service gateway architecture for a smart home. IEEE Communications Magazine 40, 126–132 (2002)
11. Bonino, D., Castellina, E., Corno, F.: The DOG gateway: Enabling ontology-based intelligent domotic environments. IEEE Transactions on Consumer Electronics 54, 1656–1664 (2008)
12. Basanta-Val, P., Garcia-Valls, M., Estevez-Ayres, I.: Real-time distribution support for residential gateways based on OSGi. In: Processings of IEEE International Conference on Consumer Electronics (ICCE), pp. 747–748. IEEE Press, Chiang Mai (2011)
13. Arrizabalaga, S., Cabezas, P., Legarda, J., Salterain, A.: Multi-residential gateway: An innovative concept and a practical approach. IEEE Transactions on Consumer Electronics 54, 444–452 (2008)
14. Angove, P., O'Grady, M., Hayes, J., O'Flynn, B., O'Hare, G., Diamond, D.: A Mobile Gateway for Remote Interaction with Wireless Sensor Networks. IEEE Sensors Journal 11, 3309–3310 (2011)
15. Chao-Lin, W., Chun-Feng, L., Li-Chen, F.: Service-Oriented Smart-Home Architecture Based on OSGi and Mobile-Agent Technology. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 37, 193–205 (2007)
16. Lan, Z., Henry, L., Chan, K.: Information fusion based smart home control system and its application. IEEE Transactions on Consumer Electronics 54, 1157–1165 (2008)
17. Gill, K., Yang, S., Yao, F., Lu, X.: A zigbee-based home automation system. IEEE Transactions on Consumer Electronics 55, 422–430 (2009)
18. Pal, A., Bhaumik, C., Shukla, J., Kolay, S.: Energy Information Gateway for Home. In: Processing of International Conference on Intelligent Systems, Modelling and Simulation (ISMS), pp. 235–240. IEEE Press, Phnom Penh (2011)
19. Shaowei, L., Weimin, L., Xianmin, P., Wei, Z.: The communication model based on SIP and ZigBee for intelligent home electricity system. In: Processing of International Conference on Multimedia Technology (ICMT), pp. 5594–5597. IEEE Press, Hangzhou (2011)
20. Garcia, J., Valera, F., Vidal, I., Azcorra, A.: A broadcasting enabled Residential Gateway for Next Generation Networks. In: Processing of IEEE/IFIP International Workshop on Broadband Convergence Networks, pp. 1–12. IEEE Press, Munich (2007)
21. Pankakoski, V.: An Experimental Design for a Next Generation Residential Gateway. Aalto University, Helsinki (2010)
22. UPnP AV Architecture v2, `http://www.upnp.org/specs/av/UPnP-av-AVArchitecture-v2-20101231.pdf`
23. UPnP QoS Architecture v3, `http://www.upnp.org/specs/qos/UPnP-qos-Architecture-v3.pdf`

Particle Swarm Optimization Combined with Tabu Search in a Multi-agent Model for Flexible Job Shop Problem

Abir Henchiri and Meriem Ennigrou

ISG, Institut Supérieur de Gestion, ur. SOIE, Stratégies d'Optimisation et Informatique Intelligente, 2000 Le Bardo, Tunisia
abirhenchiri89@gmail.com, meriem.ennigrou@enit.rnu.tn

Abstract. Flexible job shop scheduling problem (FJSP) is an important extension of the classical job shop scheduling problem, where the same operation could be processed on more than one machine and has a processing time depending on the machine used. The objective is to minimize the makespan, i.e., the total duration of the schedule. In this article, we propose a multi-agent model based on the hybridization of the tabu search (TS) method and particle swarm optimization (PSO) in order to solve FJSP. Different techniques of diversification have also been explored in order to improve the performance of our model. Our approach has been tested on a set of benchmarks existing in the literature. The results obtained show that the hybridization of TS and PSO led to promising results.

Keywords: Flexible Job Shop, Multi-agent system, Tabu Search, Particle swarm optimization, Diversification techniques.

1 Introduction

Scheduling is one of the most important issues in the planning and operation of manufacturing systems. The flexible job shop scheduling problem is a generalization of the classical job shop problem. Each operation can be processed on a given machine chosen among a finite subset of candidate machines. The objective is to assign and to sequence the operations on the resources so that they are processed in the smallest time. For its strongly NP-hard nature [1], finding an optimal solution for such problems in a reasonable time seems to be very hard therefore many efficient heuristics and meta-heuristics methods have been developed to get nearly optimal solutions. The heuristic approaches for solving FJSP are generally classified as hierarchical and integrated approaches. In hierarchical approaches, the assignment of operations to machines and the sequencing of operations on the machines are treated separately. In integrated approaches, assignment and sequencing are considered together. In recent years, many approximate methods have been developed to solve FJSP such as tabu search (TS), simulated annealing (SA), genetic algorithm (GA), Ant Colony Optimization system (ACS), etc.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 385–394, 2013.

For the hierarchical approaches, [2] was the first to use decomposition based on TS to solve the FJSP with the objective of minimizing the makespan time. [3, 4] presented a GA for mono objective and multi-objective FJSP. [5] proposed hybridizing PSO and TS algorithms for the multi-objective FJSP.

For the integrated approaches, [6] defined a new model and used a TS algorithm. [7] improved the TS techniques of [6]. [8] proposed a hybrid metaheuristic namely Variable Neighbourhood Particle Swarm Optimization (VNPSO), which is a combination of Variable Neighborhood Search (VNS) and PSO to solve multi objective FJSP.

Distributed approaches have also been proposed to solve FJSP problems. A multi-agent model was proposed by [9]. It consists of three classes of agents: Job agents and Resource agents responsible for satisfying disjunctive, temporal and precedence constraints, and an Interface agent containing the nucleus of the tabu search method. Another work of [10] which is an improvement of the model proposed in [9], in which the optimization is not only in the Interface agent but is distributed among the Resource agents. Despite of the promising results obtained in this model, some limitations have been detected.

We found on the one hand that the Job agents are useless in the proposed model because they are performed only in the phase of construction of the initial solution and are absents in throughout the optimization phase or the initial solution may be generated by the Interface agent.

On the other hand the global diversification technique used in this model is based on tabu search. Although tabu search is a very efficient method in the exploitation of the search space, it remains less efficient in the exploration of the search space [11].

For these reasons, we propose in this paper a new multi-agent model inspired from the model proposed by [10] ,namely Flexible Job Shop Multi-Agent Tabu Search Local Optimization (FJS MATSLO+) ,in which we combine the tabu search with the particle swarm optimization, to obtain a compromise between exploitation and exploration in order to solve the flexible job shop scheduling problem. Our objective is to minimize the makespan.

The remainder of this paper is organized as follows: The details of the FJSP are outlined in Section 2. The TS and PSO methods are described in section 3 and 4. Then, we describe the model proposed namely Flexible Job Shop Multi-Agent Tabu Search Particle Swarm Optimization (FJS MATSPSO) and its global dynamics. Finally, we give some experimental results.

2 Problem Formulation

The FJSP problem may be formulated as follows. $Let\ J = \{J_1 \dots J_n\}$ be a set of n jobs, which are performed on m machines $M = \{M_1 \dots M_m\}$. Each job $J_i, i = 1 \dots n$, is formed by a sequence of n_i operations that must be processed on a set of machines according to a predefined order. The operation O_{ij} denotes the j^{th} operation of the i^{th} job. Each operation O_{ij} can be performed by one or more resources and has its processing time denoted by t_{ijk} and dependent on the resource M_k used. The FJSP is thus to both determine an assignment and a sequence of the operations on the machines in order to minimize the maximal completion time of all the operations, which is denoted by makespan.

The several constraints on the jobs and machines are summarized as follows:

- each machine can process only one operation at a time;
- there are no precedence constraints among operations of different jobs;
- each job has a release time r_i,indicating the earliest possible time that the job can be processed and a due date d_i;
- preemption is prohibited, i.e. an operation running cannot be stopped until it ends;
- setup times for the operations are sequence-independent and included in the processing times;
- machines are available at any time.

3 Tabu Search

Tabu search method proposed by [12] is based on the principle of local search. It consists in visiting the neighboring states from a current one in order to find the optimal solution.

To avoid the trap in local optima, tabu search uses a temporary memorization structure in which it memorizes the last moves performed to prohibit the return to solutions recently visited. These solutions will be tabu solutions and are then forbidden in the next iterations. They are stored in a list denoted $tabu\ list$.

Tabu search has many parameters that have to be adjusted:

- initial solution;
- neighborhood function;
- evaluation technique;
- tabu list size;
- Diversification techniques.

In what follows, we will describe the adaptation of different parameters of TS to flexible job shop problem inspired from [10].

3.1 Neighbourhood Function

In order to introduce our neighborhood function, we have to define before some fundamental notions:

- A critical path is the longest path which is composed of related operations linked either by a precedence constraint or by a disjunctive constraint.
  It is possible that multiple critical paths exist in the same graph, all of the same length.
- Critical operations are the operations belonging to the critical path.
- A critical block is a sequence of adjacent critical operations performed on the same resource.

The neighbourhood of a solution S is obtained by two types of movements:

- Swap of two adjacent critical operations assigned to this resource.
- Reassignment of a critical operation performed by this resource to another potential resource.

3.2 Neighbourhood Evaluation

A complete evaluation, i.e. calculation of all start dates of all operations of each neighbor would take considerable time. This fact has led us to calculate only the start dates of a subset of operations that are effectively concerned by the movement.

In the following, we define this subset of operations in both cases: swapping two critical operations and replacement of a critical operation on another potential resource. We note $JS(O_{ij})$ the successor of operation O_{ij} in the job J_i and $MS(O_{ij})$ the next operation executed after O_{ij} on the resource executing O_{ij}.

3.2.1 Swap of Two Operations

Let O_{ij} and O_{gh} two critical operations performed by the resource R_k. The only operations that will be concerned by swapping two operations O_{ij} and O_{gh} are:

- $JS(O_{ij})$; $JS(JS(O_{ij}))$,etc..,$JS(O_{gh})$,$JS(JS(O_{gh}))$; etc ;
- $MS(O_{ij})$;$MS(\mathrm{MS}(O_{ij}))$,etc..,$MS(O_{gh})$; $MS(MS(O_{gh}))$; etc ;
- $MS(JS(O_{ij}))$;$MS(MS(JS(O_{ij})))$,etc.., $MS(JS(O_{gh}))$; $MS(MS(JS(O_{gh})))$; etc ;
- $JS(MS(O_{ij}))$; $JS(JS(MS(O_{ij})))$,etc.., $JS(MS(O_{gh}))$; $JS(JS(MS(O_{gh})))$; etc.

3.2.2 Reassignment of an Operation

Let O_{ij} be a critical operation assigned to a resource R_k and to replace on resource R_l at a date d. Let O_{xy} be the operation executed by R_l at instant d. Operations that can be changed are:

- $JS(O_{ij})$; $JS(JS(O_{ij}))$,etc..,$MS(O_{ij})$; $MS(\mathrm{MS}(O_{ij}))$; etc ;
- $MS(JS(O_{ij}))$;$MS(MS(JS(O_{ij})))$,etc…., $JS(MS(O_{ij}))$; $JS(JS(MS(O_{ij})))$; etc ;
- O_{xy};$JS(O_{xy})$;$JS(JS(O_{xy}))$;etc.., $MS(JS(O_{xy}))$; $MS(JS(MS(O_{xy})))$; etc.

3.3 Diversification Techniques

To guide the search into new areas of search space in hope to find the global optimum, diversification techniques have been integrated into the TS method of Resource agents. Each Resource agent executes a number of diversification phases during its process optimization. Each of these phases is performed when the number of iterations from the last improvement or from the last phase of diversification exceeds the predefined diversification criterion $nb\_iter\_diversif\_max_k$.

When a diversification phase should take place, Resource agent chooses between three possible diversification techniques:

- Choose a solution randomly (according to a certain probability $pelite$) from the list of elite solutions $list\_elites_k$(a list which contains the best solutions met during its optimization process or during the optimization processes of the other Resource agents) and which not belong to the diversification *tabu list* $LT\_diversif_k$ of the Resource agent;
- Choose a solution randomly (according to a certain probability $pelite\_second$) from the list of secondary elite solutions $list\_elites\_sd_k$ (This is a list containing solutions of rather good quality that the Resource agent R_k met during its optimization process or other Resource agents met during their optimization processes and they sent to it) and which not belong to the diversification *tabu list* $LT\_diversif_k$. A solution is considered of rather good quality if the difference in cost between the current solution S_k and the best solution found S_k^* not exceed a certain constant $\mu > 0$;
- Create a new solution by re-sequencing the operations of a job randomly chosen.

The solution obtained after the phase of diversification will be temporarily stored in the diversification tabu list $LT_diversif_k$to not be selected in the following diversification phases.

4 Particle Swarm Optimization

Particle swarm optimization was introduced by [13]. In this approach, individuals are called *particles* and the population is called *swarm*. Each particle in the swarm has its position, its velocity, and the value of the objective function for its current position, its neighbor's best position and its best previous position. These particles are placed randomly in the search space of the objective function. At a given moment, the position of each particle is updated taking into account its current position, its best previous position $pbest$ and the best position in its neighbor $gbest$. This move can be formulated analytically by the following relationships:

$$\boldsymbol{V_k(t+1) = w \times V_k(t) + c_1 \times rand_1 \times \left(pbest_k(t) - X_i(t)\right) + c_2 \times rand_2 \times ((gbest(t) - X_k(t))} \ . \quad (1)$$

$$\boldsymbol{X_k(t+1) = X_k(t) + V_k(t+1)} \ . \quad (2)$$

Where $V_k(t)$,$X_k(t)$, $pbest_k(t)$and $gbest(t)$ indicate respectively the current velocity, the current position, the best previous position and the neighbor's best position of the particle k at iteration t. w is the inertia weight represented to balance between the exploration and exploitation. $rand_1$ and $rand_2$ are two random numbers within the range of $[0,1]$. $c1$ and $c2$ are acceleration coefficients which control the influence of $pbest$ and $gbest$ on the search process.

We present in this section the adaptation of the parameters of particle swarm optimization for the flexible job shop problem: the first step of this adaptation is to create a structure of particles and then propose an algorithm showing the way to follow for better solving of FJSP.

4.1 Presentation of the Particle Structure

Each particle k is represented by a position vector, which is a sequence of elements. These elements represent the affectation of machines to the set of operations of all jobs. The order of appearance of the machines in the position vector represents the execution order of the operations on these machines in the solution.

The total number of elements composing the position vector of each particle is equal to the total number of operations of all the jobs. The 1st operation of the first job is represented at the 1st position; the 2nd operation of the first job is represented at the 2nd position and so on.

4.2 Modified Particle Velocity

The velocity of operation O_{ij} of a particle k is denoted V_{ij}^{k}, $V_{ij}^{k} \in \{0,1\}$ where O_{ij} is the j^{th} operation of job i. For $t = 0$, the velocity $V_{ij}^{k}(0)$ will be generated randomly and it will be updated for the rest of iterations. When V_{ij}^{k} equals to 0, it means that operation O_{ij} has just been moved to the current location and we should not move it in this iteration. On the contrary, if $V_{ij}^{k} = 1$ then the operation O_{ij} can be moved to a new location in this iteration and if the replacement is performed we set V_{ij}^{k} to 0 and O_{ij} will not be moved in the next few iterations.

To prevent the optimization process to be trapped into local optima, the velocity is controlled by inertia weight w in *(Eq. (1))*. We randomly update velocities at the beginning of each iteration. For each operation O_{ij} of particle k, if $V_{ij}^{k}(t)$ equals to 0, $V_{ij}^{k}(t+1)$ will be set to 1 with probability w.

4.3 Modified Particle Movement

The position of each particle is updated with respect to its previous best tion $pbest$ and the global best position $gbest$.The modified particle movement is based on replacement of an operation on another resource. If $V_{ij}^{k} = 1$, the operation O_{ij} of the current position of the particle k will be moved to the selected machine for the execution of this operation in the best known position of the particle $pbest^{k}$ with probability c_1 and will be moved to the corresponding machine of the best position of the neighborhood $gbest$ with probability c_2, where c_1 and c_2 are constant between 0 and 1, and $c_1 + c_2 \leq 1$. After replacement we set $V_{ij}^{k} = 0$.

The new particle's position found in the current iteration will replace its previous best position $pbest^{k}$ if its solution is better than the previous best. The best tion $pbest^{k}$ will be updated as the global best solution if its solution is better than the previously stored global best solution $gbest$. The procedure is repeated until the termination criterion is satisfied.

5 Architecture of the Multi Agent Model

Our model consists of two classes of agents: Resource agents and an Interface agent. Each agent in our model has its own static and dynamic knowledge and a mailbox where it stores its messages received from its acquaintances, i.e. the agents that it knows and with which it can communicate.

5.1 Resource Agents

They are responsible of the satisfaction of disjunctive constraints. Several local optimization processes based on TS method are placed on each Resource agent. These different processes communicate with each other by sending solutions. Diversification techniques have been introduced in each Resource agent to better explore the search space.

5.2 Interface Agent

This class contains a single agent which is responsible of the satisfaction of temporal and precedence constraints and plays the role of interface between the set of agents and the user:

- create a collection of agents necessary for solving the flexible job shop problem;
- recognize that the problem was solved by this collection; and
- inform the user about the result found.

A global optimization process based on particle swarm optimization method is located in the Interface agent.

6 Global Dynamic Multi Agent

The first step in the global dynamic of the model consists in generating an initial solution S_0; this generation is done randomly by the Interface agent which assigns each operation on a potential resource chosen randomly. The initial solution S_0 is then sent to each Resource agent who executes its own local optimization process based on TS in order to better exploit the good areas. A global optimization process is integrated at the Interface agent. It is based on PSO in order to diversify the search towards unexplored areas. This global optimization is executed in parallel to all other local optimization processes.

6.1 Local Optimization Process

When the Resource agent R_k receives the initial solution sent by the Interface agent which became its current solution, it determines the neighborhood of this solution and evaluates it according to the evaluation technique presented previously. The movement will be performed locally, it becomes tabu for some iterations and the new solution will be obtained after satisfaction of all constraints. Two cases are possible:

- If the new solution improves the cost of the best solution found until this iteration then the Resource agent sends it to all other Resource agents R_k, $k = 1 \dots m$, to save it in their list of elite solutions and it saves it in its own list of elite solutions. This new solution is also sent to the Interface agent to be saved in its own list of solutions.
- If the new solution does not improve the cost of the best solution found and if the difference in cost between these two solutions is less than a constant μ then the Resource agent R_k sends it to all Resource agents to save it in their list of secondary elite solutions. This solution will be sent to the Interface agent after a number n of secondary elite solutions found to be saved in its own list of solutions. When the number of iterations from the last improvement or the last phase of diversification exceeds $nb\_iter\_diversif\_max$, the Resource agent R_k executes one of the diversification techniques (section 3.3) according to a certain probability p_{elite}.

This iteration of the TS will be repeated $nb\_iter\_max_k$ times.

6.2 Global Optimization Process

The global optimization process based on PSO will be started when the number of solutions sent to the Interface agent by the Resource agents will be equal to the number of particles in the swarm. This number will be fixed experimentally. The initial swarm generation is then formed by the list of solutions received from the Resource agents i.e. each solution represents a particle. The particle is represented by a position vector which contains the assignment of machines to operations. Each time, a particle is selected from the swarm and a neighborhood containing the particle designated is chosen. At each iteration, the Interface agent compares the assignments of machines of the selected particle with those of the best solution in its neighborhood $gbest$ and with those of its best known position $pbest$ and it makes the modifications after satisfaction of all the constraints (section 4.3). The new particle's position obtained will replace its previous best position if it improves the cost and will also replace the global best particle if it is better in cost than the previously stored global best solution.

This procedure will be repeated until the number of iterations reaches $nb\_iter\_PSO\_max$. At each iteration, if the Interface agent finds a global best solution $gbest$ it sends it to all the Resource agents to be saved in their list of elite solutions.

7 Computational Results

The FJS MATSPSO was implemented in Eclipse with the language Java. For the development of multi-agent system we chose JADE (Java Agent Development framework) which is a free and an open source agent development platform. The implementation has been performed on a machine based on Intel processor "Core2Duo" clocked at 2.10 GHz and 4Go ram. To illustrate the performance of our proposed model, tests on problem instances from [2], [6] and [14] were performed. This proposed model is run 5 times for each problem and the best solution obtained has been taken for comparison.

The parameters of the TS and PSO were fixed experimentally as follows:

- Size_TL$_k$=12;
- nb_iter_max$_k$=1000;
- nb_iter_diversif_max=15;
- Size_list_elites=5;
- Size_TL_diversif=3;
- pelite=0.7;
- pelite_second=0.3;
- Size_swarm =50;
- nb_iter_PSO_max =1000;
- c_1=0.6;
- c_2=0.4;
- w=0.6;
- Size_neighborhood=5.

To illustrate the efficiency of our model the results were compared to the two bounds (LB, UB) existing in the literature and the results obtained by the Multi-agent model proposed by [10]. Our experiments were concentrated on the makespan and the CPU time. However, in the multi-agent model proposed by [10] the experiments were limited on the makespan and CPU time was ignored. For that, our model was outperformed in terms of solution quality (the results refer to the makespan obtained by both approaches for each benchmark considered).

In this paper we present only the results obtained on some benchmarks from [2]. We mark with (*) the instance having a cost belonging to the interval [LB, UB].

The Table 1 shows that our proposed model found the optimal solution for 90% of instances and the results obtained are clearly better than those obtained by FJS MATSLO$^+$ model in 100 % of instances. The superior results indicate the successful hybridization of PSO and TS. This is due to the fact that TS has facilitated a better exploitation of the search space whereas PSO has facilitated a better exploration of it. Further more, the results obtained by FJS MATSLO$^+$ encouraged us to re-implement our model with a parallel architecture in hope of finding more better results.

Table 1. Benchmarks of BRdata [2]

	n×m	LB	UB	FJS MATSPSO	CPU time	FJS MATSLO+
Mk01	10×6	36	42	39*	36s	40
Mk02	10×6	24	32	27*	37.2s	32
Mk03	15×8	204	211	207*	50.1s	207
Mk04	15×8	48	81	65*	41.8s	67
Mk05	15×4	168	16	174*	48.9s	188
Mk06	10×15	33	86	72*	50.1s	85
Mk07	20×5	133	157	154*	54.1s	154
Mk08	20×10	523	523	523*	55.1s	523
Mk09	20×10	299	369	340*	56.7s	437
Mk10	20×15	165	296	299	63.1s	380

8 Conclusion

In this article, a hybridization between PSO and TS via a multi-agent model is proposed for solving the flexible job shop problem. The proposed multi-agent model is composed of Resource agents and an Interface agent. On each Resource agent is placed a local optimization process which is based on tabu search. Local diversification techniques are executed in each Resource agent. A global optimization process based on particle swarm optimization has been integrated at the Interface agent. The experiments prove that the proposed approach is better than the FJS MATSLO$^+$ in terms of makespan and this is due to the hybridization of TS and PSO methods.

References

1. Garey, E.L., Johnson, D.S., Sethi, R.: The Complexity of Flow-shop and Job-shop scheduling. Mathematics of Operations Research 1, 117–129 (1976)
2. Brandimarte, P.: Routing and scheduling in a flexible job shop by tabu search. Annals of Operations Research 4(1-4), 157–183 (1993)
3. Kacem, I., Hammadi, S., Borne, P.: Approach by Localization and Multi-Objective Evolutionary Optimization for Flexible Job Shop Scheduling Problems. IEEE Trans. Systems, Man and Cybernetics 32(1), 1–13 (2002)
4. Kacem, I., Hammadi, S., Borne, P.: Pareto-Optimality Approach for Flexible Job Shop Problems: Hybridization of Evolutionary Algorithms and Fuzzy Logic. Mathematics and Computers in Simulation 60, 245–276 (2002)
5. Zhang, G., Shao, X., Li, P., Gao, L.: An effective hybrid particle swarm optimization algorithm for multi objective flexible job-shop scheduling problem. Computers and Industrial Engineering 56(4), 1309–1318 (2009); Moslehi, G., Mahnam, M.: Int. J. Production Economics 129 (2009)
6. Dauzère-Pérès, S., Paulli, J.: An Integrated Approach for Modeling and Solving the General Multiprocessor Job Shop Scheduling Problem using Tabu Search. Annals of Operations Research 70(3), 281–306 (1997)
7. Mastrolilli, M., Gambardella, M.L.: Effective Neighborhood Functions for the Flexible Job Shop Problem. J. Scheduling 3(1), 3–20 (2000)
8. Liu, H., Abraham, A., Choi, O., Moon, S.H.: Variable Neighborhood Particle Swarm Optimization for Multi-objective Flexible Job-Shop Scheduling Problems. In: Wang, T.-D., Li, X., Chen, S.-H., Wang, X., Abbass, H.A., Iba, H., Chen, G.-L., Yao, X. (eds.) SEAL 2006. LNCS, vol. 4247, pp. 197–204. Springer, Heidelberg (2006)
9. Ennigrou, M., Ghédira, K.: Approche Multi-Agents basée sur la Recherche Tabou pour le Job Shop flexible. In: RFIA, 14ème Congrès Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle, Toulouse, France (2004)
10. Ennigrou, M., Ghédira, K.: New local diversification techniques for the Flexible Job Shop problem with a Multi-agent approach. JAAMAS, Journal of Autonomous Agents and Multi-Agent Systems 17(2), 270–287 (2008)
11. Duvivier, D.: Etude de l'hybridation des méta-heuristiques, application a un problème d'ordonnancement de type job shop. Université du Littorial côté d'opale (2000)
12. Glover, F.: Future paths for integer programming and links to artificial intelligence. Computers and Operations Research 5, 533–549 (1986)
13. Kennedy, J., Eberhar, R.C.: Particle swarm optimization. In: IEEE International Conference on Neural Networks, Piscataway, pp. 1942–1948 (1995)
14. Hurink, E., Jurisch, B., Thole, M.: Tabu search for the job shop scheduling problem with multi-purpose machine. Operations Research Spektrum 15, 205–215 (1994)

A Novel Preprocessing Method for Illumination-Variant Color Face Image

Wei Li[1], Qinghua Yang[2], and Wei Pan[1]

[1] Computer School, China West Normal University, Nanchong, Sichuan, 637002 China
[2] Medical Imaging Department, North Sichuan Medical College, Nanchong, 637000 China
nos036@163.com

Abstract. This paper proposes a novel preprocessing method for the illumination-variant color face image. The proposed method aims to balance the luminosity and the color variation by color adjustment, bilateral filtering, and luminosity adjustment of sub image, which were required form input image by row-column transform and form which the output image was required by row-column inverse transform. The experimental results show that this preprocessing method can help to improve the segmentation precision and has good speed and robustness.

Keywords: face recognition, luminosity variance, preprocessing method.

1 Introduction

Face recognition has various applications in many fields such as information security, personal identification, and commercial transaction, but it is a challenging problem because of some factors, such as pose, aging, facial expression, and background variation. The variable illumination is one of the most important and familiar factor, which influences, even changes the visibility of appearance of the human face, as it maybe induces the color cast, color and luminosity unbalance, underexposure and overexposure. For solving the problem of illumination-variant in face recognition, people presented some method, such as illumination compensation [1][2], illumination normalization [2][3], shadow compensation [4], and so on. This paper proposed a novel and adaptive method to remove the color cast, balance the color and luminosity, depress the shadow and high light caused by underexposure and overexposure, and then using CbCr ellipse skin model, analyzed comparatively the result of segmentation against the original input image and preprocessing image.

2 Overview of the Preprocessing Method

As the important part of face recognition, the preprocessing of color face image in this paper mainly contains row-column transform and inverse transform, color adjustment, bilateral filtering, as well as luminosity adjustment. The flow of the preprocessing was depicted in Fig. 1, which was described as follows:

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 395–403, 2013.

1. Color Face Image Input: to get the input image by input device.
2. Row-Column Transform: to transform form the input image to the sub-images.
3. Color Adjustment of sub-images: to decrease the color cast and color unbalance of sub-images in the RGB color space.
4. Bilateral Filtering of sub-images: to smooth the face and background region, remove noise and reserve the facial features.
5. Luminosity Adjustment of sub-images: to compensate the insufficient luminosity, and depress the excess luminosity for balancing the luminosity of sub-images.
6. Row-Column Inverse Transform: to transform form the color balanced and illumination compensated sub-images to the output image.
7. Color Face Image Output: to output the preprocessed image.

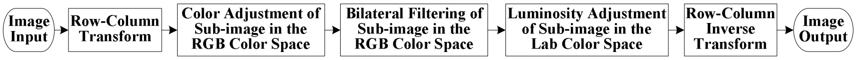


Image Input
Row-Column Transform
Color Adjustment of Sub-image in the RGB Color Space
Bilateral Filtering of Sub-image in the RGB Color Space
Luminosity Adjustment of Sub-image in the Lab Color Space
Row-Column Inverse Transform
Image Output

Fig. 1. Flowchart of the preprocessing

3 The Preprocessing Method

3.1 Row-Column Transform and Row-Column Inverse Transform

For using the local characteristics of subspace and preserving the global facial features of the face image, the process of the color adjustment, the bilateral filter and the luminosity adjustment implemented separately in every subspace of the original image. The function of row-column transform was to transform the whole input image to the sub-images in their subspaces. After filtering, color and luminosity adjustment of sub-images, the sub-images were transformed to the output image inversely again. The row-column transform was described as follows:

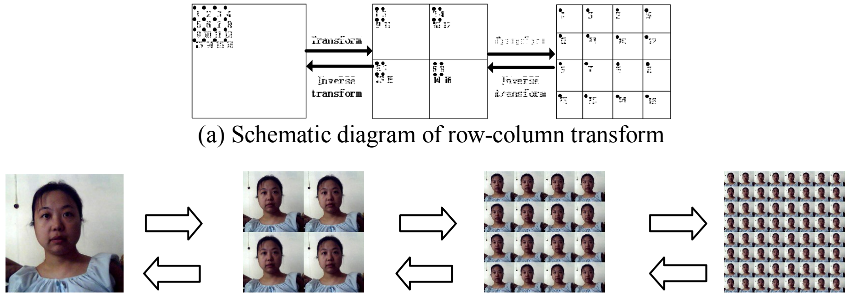
$$\begin{cases} k=1,2,\cdots,N, \quad N=3, \quad kj=1,2,\cdots,2^{k-1}, \quad ki=1,2,\cdots,2^{k-1} \\ nw=w/2^{k}, \; nh=h/2^{k}, \; pw=w/2^{k-1}, \; ph=h/2^{k-1} \\ i=(ki-1)\times pw,\cdots,ki\times pw, \quad j=(k2-1)\times ph,\cdots,kj\times ph \\ d_i=(i\% \, pw)/2+(ki-1)\times pw, \quad d_j=(j\% \, ph)/2+(kj-1)\times ph \end{cases} \tag{1}$$

$$\begin{cases} i'=d_i, j'=d_j \quad if\ (i\%2==0\ \&\ \&\ j\%2==0) \\ i'=d_i+nw, j'=d_j \quad if\ (i\%2!=0\ \&\ \&\ j\%2==0) \\ i'=d_i, j'=d_j+nh \quad if\ (i\%2==0\ \&\ \&\ j\%2!=0) \\ i'=d_i+nw, j'=d_j+nh \quad if\ (i\%2!=0\ \&\ \&\ j\%2!=0) \end{cases} \tag{2}$$

$$R^{'}_{i',j'}=R_{i,j}, \quad G^{'}_{i',j'}=G_{i,j}, \quad B^{'}_{i',j'}=B_{i,j} \tag{3}$$

where i, j are a point's abscissa and ordinate of the image before transformation, d_i and d_j are the corresponding point's abscissa and ordinate of the row-column transformed image, N is the transform layer number, ki and kj are the transform levels, pw and ph are the width and length of the image before transformation, the nw and nh are the width and length of the row-column transformed sub-image.

The process of row-column transform is shown in Fig. 2 from left to right. Note that the changes of these points' position. The process of row-column inverse transform is shown in the opposite direction in Fig. 2(a). The way to the row-column inverse transform was described in Eq. (4).



(a) Schematic diagram of row-column transform

(b)Input image (c)Transformed image1 (ki=kj=1) (d)Transformed image2 (ki=kj=2) (e)Transformed image3 (ki=kj=3)

Fig. 2. Example of the row-column transform and row-column inverse transform

$$R_{i,j}=R'''_{i',j'},\quad G_{i,j}=G'''_{i',j'},\quad B_{i,j}=B'''_{i',j'} \tag{4}$$

where $R'''_{i',j'}$, $G'''_{i',j'}$, $B'''_{i',j'}$ are the R, G, B components of the image, which is processed by color adjustment, bilateral filtering, and luminosity adjustment.

Fig. 2(b)-2(e) showed an single-face example of the row-column transform and row-column inverse transform. It is obvious that the transformed sub-images preserved the global facial features separately, and at the same time they each could provide the information of subspace to be processed later. After the color adjustment, bilateral filtering, and the luminosity adjustment, the processed sub-images were transformed inversely to the single-face image, which is different from the original input image.

3.2 Color Adjustment of Sub-image

Aiming at the row-column transformed sub-images, the color adjustment, bilateral filtering, and the luminosity adjustment are the vital parts of the preprocessing. The color adjustment is to decrease the color cast and the color unbalance, which may be caused by angle and instability of light source. In the RGB color space, by means of the average values of R, G, B components of every transformed sub-image (see Eq. (5)-(7)), the color components of sub-image can be adjusted (see Eq. (8)-(9)).

$$Y'_{i',j'}=0.299\times R'_{i',j'}+0.587\times B'_{i',j'}+0.114\times G'_{i',j'}+128 \tag{5}$$

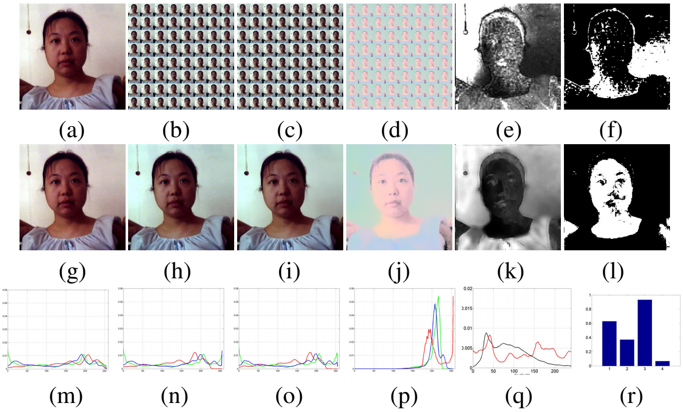
$$kr=ki\times pw,\quad krr=(ki-1)\times pw,\quad kt=kj\times ph,\quad ktt=(kj-1)\times ph \tag{6}$$

$$sR = \frac{\sum_{i=krr}^{kr}\sum_{j=ktt}^{kt} R'_{i',j'}}{(pw \times ph)}, \quad sB = \frac{\sum_{i=krr}^{kr}\sum_{j=ktt}^{kt} B'_{i',j'}}{(pw \times ph)}, \quad sG = \frac{\sum_{i=krr}^{kr}\sum_{j=ktt}^{kt} G'_{i',j'}}{(pw \times ph)}, \quad sY = \frac{\sum_{i=krr}^{kr}\sum_{j=ktt}^{kt} Y'_{i',j'}}{(pw \times ph)} \tag{7}$$

$$R''_{i',j'} = R'_{i',j'} \times \frac{\text{sRGB}}{sR}, \quad B''_{i',j'} = B'_{i',j'} \times \frac{\text{sRGB}}{sB}, \quad G''_{i',j'} = G'_{i',j'} \times \frac{\text{sRGB}}{sG}, \quad \text{sRGB} = \frac{(sR + sB + sG)}{3} \tag{8}$$

$$R''_{i',j'} = 255 \; if \, (R''_{i',j'} > 255), B''_{i',j'} = 255 \; if \, (B''_{i',j'} > 255), G''_{i',j'} = 255 \; if \, (G''_{i',j'} > 255) \tag{9}$$

where sR, sG, sB, sY are the average values of R, G, B, Y components of sub-images. $R'_{i',j'}$, $G'_{i',j'}$, $B'_{i',j'}$ are the R, G, B components of a pixel in a sub-image, $R''_{i',j'}$, $G''_{i',j'}$, $B''_{i',j'}$ are the adjusted R, G, B components of the corresponding pixel.

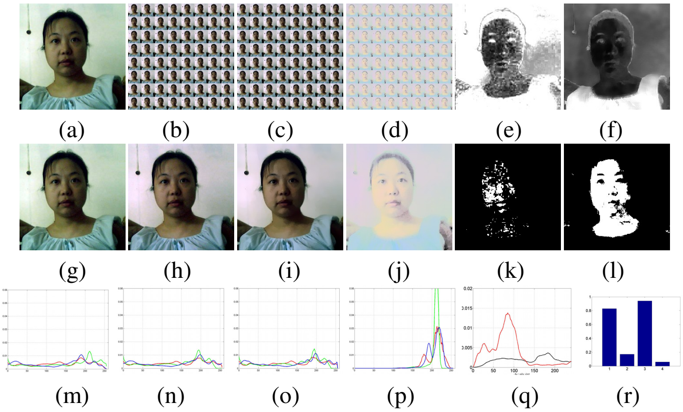


(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

(m) (n) (o) (p) (q) (r)

Fig. 3.1. The preprocessing and segmentation of red color cast input image



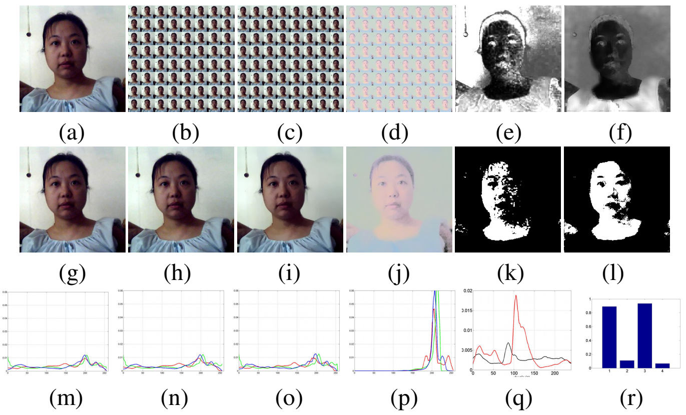
(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

(m) (n) (o) (p) (q) (r)

Fig. 3.2. The preprocessing and segmentation of green color cast input image

Fig. 3 shows three examples of the preprocessing and segmentation, which were based on the red color cast, green color cast, and blue color cast input image separately. The Fig. 3.1(a), 3.2(a), 3.3(a), Fig. 3.1(g), 3.2(g), 3.3(g) are the input images, the Fig. 3.1(b), 3.2(b), 3.3(b) are the color adjusted sub-images, the Fig. 3.1(h), 3.2(h), 3.3(h) are the color adjusted images. Comparing the Fig. 3.1(a) and 3.1(h) , the Fig. 3.2(a) and 3.2(h), the Fig. 3.3(a) and 3.3(h), the color cast (red, green, and blue) and the color unbalance have been decreased. Comparing the Fig. 3.1(m) and 3.1(n) (or Fig. 3.2(m) and 3.2(n), or Fig. 3.3(m) and 3.3(n)), the curve peaks of color cast shifted to the left, and that means the color cast have decreased.


(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

(m) (n) (o) (p) (q) (r)

Fig. 3.3 The preprocessing and segmentation of blue color cast input image

Fig. 3. Examples of the preprocessing and segmentation

(a) Original input images, (b) Color adjusted sub-images, (c) Bilateral filtered sub-images, (d) Luminosity adjusted sub-images, (e) CbCr ellipse skin likelihood images based on (a), (f) CbCr ellipse skin likelihood images based on (j), (g) Original input images, (h) Color adjusted images, (i) Bilateral filtered images, (j) Luminosity adjusted images, (k) Segmented images based on (a), (l) Segmented images based on (j), (m) The probability density curve of R, G, B distribution based on (a), (n) The probability density curve of R, G, B distribution based on (h), (o) The probability density curve of R, G, B distribution based on (i), (p) The probability density curve of R, G, B distribution based on (j), (q) The probability density distribution of gray levels based on (e) and (f) , (r) The correct and error rates based on (k) and (l).

3.3 Bilateral Filtering of Sub-image

The bilateral filtering is to remove some noises, which were caused by input device, the imaging condition, but preserve the edge information of facial features detail, such as eyes, mouth, face contour, and so on. The bilateral filtering of sub-images in the RGB color space was realized by Eq. (10)-(12).

$$\begin{cases} R''_{i',j'} = \sum\limits_{i'=0}^{w}\sum\limits_{j'=0}^{h}\sum\limits_{k=i'-1}^{i'+1}\sum\limits_{l=j'-1}^{j'+1} R''_{k,l}w_{i',j',k,l} \Big/ \sum\limits_{i'=0}^{w}\sum\limits_{j'=0}^{h}\sum\limits_{k=i'-1}^{i'+1}\sum\limits_{l=j'-1}^{j'+1} w_{i',j',k,l} \\ B''_{i',j'} = \sum\limits_{i'=0}^{w}\sum\limits_{j'=0}^{h}\sum\limits_{k=i'-1}^{i'+1}\sum\limits_{l=j'-1}^{j'+1} B''_{k,l}w_{i',j',k,l} \Big/ \sum\limits_{i'=0}^{w}\sum\limits_{j'=0}^{h}\sum\limits_{k=i'-1}^{i'+1}\sum\limits_{l=j'-1}^{j'+1} w_{i',j',k,l} \\ G''_{i',j'} = \sum\limits_{i'=0}^{w}\sum\limits_{j'=0}^{h}\sum\limits_{k=i'-1}^{i'+1}\sum\limits_{l=j'-1}^{j'+1} G''_{k,l}w_{i',j',k,l} \Big/ \sum\limits_{i'=0}^{w}\sum\limits_{j'=0}^{h}\sum\limits_{k=i'-1}^{i'+1}\sum\limits_{l=j'-1}^{j'+1} w_{i',j',k,l} \end{cases} \tag{10}$$

$$R''_{i',j'}=255\ if(R''_{i',j'}>255), B''_{i',j'}=255\ if(B''_{i',j'}>255), G''_{i',j'}=255\ if(G''_{i',j'}>255) \tag{11}$$

$$d_{i',j',k,l}=\exp\left(\frac{-[(i'-k)^2+(j'-l)^2]}{2\sigma_d^{\ 2}}\right),\ r_{i',j',k,l}=\exp\left(\frac{-[(R''_{i',j'}-R''_{k,l})+(B''_{i',j'}-B''_{k,l})+(G''_{i',j'}-G''_{k,l})]^2}{2\sigma_d^{\ 2}}\right) \tag{12}$$

where $R''_{i',j'}$, $G''_{i',j'}$, $B''_{i',j'}$ depend on the weight value $w_{i',j',k,l}=d_{i',j',k,l}\bullet r_{i',j',k,l}$, containing definition domain nuclear $d_{i',j',k,l}$ and range nuclear $r_{i',j',k,l}$. The σ_d=100, i', j' are the coordinates of the present pixel, and k, l are the coordinates of the adjacent pixel. Fig. 3.1(c) (or 3.2(c), 3.3(c)) and Fig. 3.1(h) (or 3.2(h), 3.3(h)) are the bilateral filtered sub-images and bilateral filtered images. By comparison, the bilateral filtered images removed the noise and some pixels which have a great difference from their adjacent pixels, but the edge information contained the facial features have not been damaged.

3.4 Luminosity Adjustment of Sub-image

The luminosity adjustment is to balance the insufficient and excess luminosity of the sub-images by adjusting the L component of the sub-images in the Lab color space. The conversion mode between the RGB and the Lab color space are described in Eq. (16) and Eq. (18).The proposed luminosity adjustment function (see Eq. (17)) was realized by using the adjustment coefficient $C_{i',j'}$ (see Eq.(17)). The adjustment coefficient depends on sY(see Eq. (5)), $Y''_{i',j'}$, $T_{i',j'}$, $S_{i',j'}$ (see Eq. (13)-(15)) components. The process of luminosity adjustment of sub-images is shown in Eq. (13)-(19).

$$Y''_{i',j'}=0.299\times R''_{i',j'}+0.587\times B''_{i',j'}+0.114\times G''_{i',j'}+128 \tag{13}$$

$$S_{i',j'}=\sqrt[3]{R''^{\ 2}_{i',j'}+B''^{\ 2}_{i',j'}+G''^{\ 2}_{i',j'}},TS_{i',j'}=\sqrt{\frac{1}{2}[(T_{i',j'}^{\ 2}+S_{i',j'}^{\ 2})\big/(T_{i',j'}+S_{i',j'})^2]} \tag{14}$$

$$T_{i',j'}=\frac{\mathrm{atan}(\frac{R''_{i',j'}}{B''_{i',j'}})+\mathrm{atan}(\frac{R''_{i',j'}}{G''_{i',j'}})+\mathrm{atan}(\frac{B''_{i',j'}}{R''_{i',j'}})+\mathrm{atan}(\frac{B''_{i',j'}}{G''_{i',j'}})+\mathrm{atan}(\frac{G''_{i',j'}}{B''_{i',j'}})+\mathrm{atan}(\frac{G''_{i',j'}}{R''_{i',j'}})}{2\pi} \tag{15}$$

$$\begin{bmatrix} L'_{i',j'} \\ a'_{i',j'} \\ b'_{i',j'} \end{bmatrix} = \begin{bmatrix} 0.2126 & 0.7152 & 0.0722 \\ 0.32639537 & -0.4999911 & 0.17359573 \\ 0.12171505 & 0.37825905 & -0.4999747 \end{bmatrix} \begin{bmatrix} R''_{i',j'} \\ G''_{i',j'} \\ B''_{i',j'} \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} \tag{16}$$

$$\begin{aligned} C_{i',j'} &= \left(T_{i',j'} \times TS_{i',j'}\right)^{\left(Y''_{i',j'}/sY\right)} \\ L'_{i',j'} &= \max\ Y \times \left[\left(L'_{i',j'} \times C_{i',j'}\right)/Y''_{i',j'}\right]^{\left(C_{i',j'}\right)} + C_{i',j'} \end{aligned} \tag{17}$$

$$\begin{bmatrix} R''_{i',j'} \\ G''_{i',j'} \\ B''_{i',j'} \end{bmatrix} = \begin{bmatrix} 1 & 2.09336015 & 0.86950063 \\ 1 & 0.62592315 & 0.07238507 \\ 1 & 0.03609180 & 1.84354662 \end{bmatrix} \begin{bmatrix} L'_{i',j'} \\ a'_{i',j'} - 128 \\ b'_{i',j'} - 128 \end{bmatrix} \tag{18}$$

$$R'''_{i',j'} = 255\ if\,(R'''_{i',j'} > 255), B'''_{i',j'} = 255\ if\,(B'''_{i',j'} > 255), G'''_{i',j'} = 255\ if\,(G'''_{i',j'} > 255) \tag{19}$$

Fig. 3.1(d) (or 3.2(d), 3.3(d)) and Fig. 3.1(j) (or 3.2(j), 3.3(j)) are the luminosity adjusted sub-images and the luminosity adjusted images. The Fig. 3.1(p) (or 3.2(p), 3.3(p)) are the probability density curve of R, G, B distribution based on the luminosity adjusted images (Fig. 3.1(j) (or 3.2(j), 3.3(j))). By comparison, it is find that the shadow and high light were depressed in the luminosity adjusted images, the luminosity were balanced, and the differences between the skin region and the background region became more obvious. Fig 3.1(e) (or 3.2(e), 3.3(e))and Fig. 3.1(f) (or 3.2(f), 3.3(f))are the CbCr skin likelihood images based on the original input images and the preprocessed images (luminosity adjusted images), Fig. 3.1(q) (or 3.2(q), 3.3(q)) are the probability density distribution of gray levels based on above two likelihood images. By comparison, it is obvious that the skin region and non-skin region are both more smooth and equal, but it's more easy to distinguish the skin and non-skin region of Fig 3.1(f) (or 3.2(f), 3.3(f)), based on which the probability density distribution of gray levels have the more obvious and sharp double peaks. Fig. 3.1(k) (or 3.2(k), 3.3(k))and Fig. 3.1(l) (or 3.2(l), 3.3(l))are the segmented images based on the CbCr skin likelihood images of original input images and of the processed images, Fig. 3.1(r) (or 3.2(r), 3.3(r)) show the correct rate and error rate based on above two segmented images. After this preprocessing method, the segmented images can get more real skin region, and get less non-skin region, the correct rates of segmentation are higher and the error rates of segmentation are lower.

4 Experiment Results

The self-built color face database contained some images which came from network and were autodyne images (about 100 faces), and the other images that came from the AR database (about 50 faces). The conditions of experiment are: P4, 2.1 GHz CPU, 2G memory, WinXP OS, and VC6.0. The results of preprocessing and segmentation were shown in Fig. 4, which were based on the single faces in several different conditions of the illumination and color. Comparing the Fig. 4(a) and 4(c) (or Fig. 4(f) and 4(h)) in Fig. 4,it is easy to find that the preprocessing can depress the influence of non-ideal conditions of illumination and color, improve the qualities of the face images. Comparing the Fig. 4(d) and 4(e) (or Fig. 4(i) and 4(j)), it is suggested that the segmentation based on preprocessing got better effect of skin segmentation.

The table 1 shows the average correct rate and the error rate of segmentation using the CbCr ellipse skin models, which are built based on the original input images and the preprocessed images. It indicated that the preprocessing method can help to improve the average correct rate and decrease the average error rate of segmentation. The table2 lists the average time in the stages of preprocessing method. From table2, the average time of total shows that the preprocessing method had a good speed.

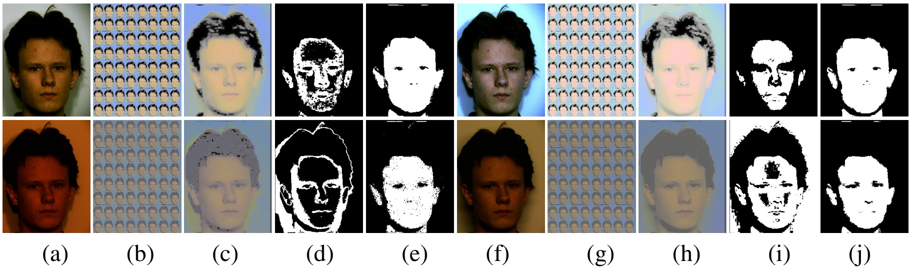


Fig. 4. The results of preprocessing and comparison of segmentation
(a)(f) Input images, (b)(g) Preprocessed sub-images, (c)(h) Preprocessed images, (d)(i) Segmented images based CbCr ellipse skin likelihood of (a), (e)(j) Segmented images based CbCr ellipse skin likelihood of (c).

Table 1. The Average Correct Rate and Error Rate of Segmentation(%)

Method of segmentation	Correct rate	Error rate
Segmentation using CbCr ellipse skin model of input images[5][6]	77.6 %	21.4%
Segmentation using CbCr ellipse skin model of preprocessed images	93.6%	5.4%

Table 2. The Average Time in Every Stage of Proposed Method

Stage pixels	Row-column transform and inverse transform	Color adjustment	Bilateral filtering	Luminosity adjustment	Total
50*50	8ms	8ms	30ms	24ms	70ms
80*80	8ms	23ms	109ms	36ms	176ms
100*100	8ms	23ms	187ms	55ms	273ms

5 Conclusion

A novel preprocessing method for illumination-invariant color face image has been presented in this paper, in which the sub-images were required by row-column transform, and were restored to a whole face image by row-column inverse transform. Based on the sub-images, the color adjustment, the bilateral filtering, and the luminosity adjustment were proposed for decreasing the color cast, color unbalance, noise, insufficient and excess luminosity. The experiments show that the proposed method of preprocessing can help to improve the accuracy of image segmentation, and have a favorable real-time performance and robustness. So it has a great potential for the robust face recognition when the lighting and color conditions change severely.

Acknowledgment. This work was supported by Sichuan Province Education Department General Project: NO. 13ZB0012.

References

1. Liau, H.F., Isa, D.: New illumination compensation method for face recognition. International Journal of Computer and Network Security 3, 5–12 (2010)
2. Javier, R.S., Julio, Q.: Illumination compensation and normalization in eigenspace-based face recognition: A comparative study of different pre-processing approaches. Pattern Recognition Letters 29, 1966–1979 (2008)
3. Villegas, M., Paredes, R.: Comparison of illumination normalization methods for face recognition. In: Third COST 275 Workshop - Biometrics on the Internet, pp. 27–28 (2005)
4. Choi, S.I., Jeong, G.M.: Shadow compensation using fourier analysis with application to face recognition. IEEE Signal Process Lett. 18, 23–26 (2011)
5. Lee, J.Y., Yoo, S.I.: An elliptical boundary model for skin color detection. In: International Conference on Imaging Science, Systems, and Technology, pp. 579–584 (2002)
6. Hsu, R.L., Mottaleb, M.A., Jain, A.K.: Face detection in color images. IEEE Trans. Patter. Anal. Mach. Intell. 5, 696–706 (2002)

Design of Face Detection System Based on FPGA

Yujie Zhang, Meihua Xu, and Huaming Shen

School of Mechatronics Engineering and Automation,
Shanghai University, Shanghai, China
zhangjay0710@gmail.com

Abstract. To solve the real-time problem of face detection, considering the realization bottleneck of AdaBoost pure software algorithm, FPGA-based hardware acceleration platform strategy is proposed. The paper analyzes the algorithm and partition the module for accelerating. The ZYNq-7000 platform FPGA from XILINX is adopted in the experiment, in which the hardware and software co-design is used. The final results show that it can detect face sat a 17fps speed with high hit rate and low false detection rate.

Keywords: FPGA, Face Detection, Adaboost Algorithm.

1 Introduction

Face detection is a computerized technology that can achieve the locations and sizes of human faces in arbitrary (digital) images. It detects the facial features only and ignores the else, such as buildings, trees and bodies. Face detection technology can be used in face reorganization, video conference, image and video retrieval, intelligent human–computer interaction and other fields. With the development of embedded technology and smart device, face detection technology will play an important role in meeting the requirements of mobilization and outdoor work.

There're two key indicators in measuring the performance of face detection: Accuracy and Speed. Proposed by Freund and Scrapire, AdaBoost target detection algorithm in accuracy and speed has reached a higher level. AdaBoost face detection algorithm is applied to achieve a true real-time detection, which makes real-time embedded platform and face detection possible. There is already some research and practice in this area currently, but most of the research work is in software. AdaBoost algorithm has a heavy load and large amount of data, so the pure software implementation of the access point's chart will come across the bottleneck of the algorithm. Therefore, the detection algorithm embedded platform with the ability of the processor alone cannot achieve real-time requirements, so we need to find a hardware acceleration AdaBoost algorithm approach [3]. In this paper, due to the bottlenecks of its design improved algorithms for software implementation hardware-accelerated method is applied and it achieves the square and the integral image and integration of real-time calculation and classification features of value. The ZYNq-7000 platform FPGA from XILINX is used in this paper.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 404–410, 2013.

2 The Software Algorithm in Face Detection

2.1 Adaboost Face Detection Algorithm Principle

AdaBoost, short for Adaptive Boosting, is a machine learning algorithm, formulated by Yoav Freund and Robert Schapire [4]. It is a meta-algorithm, which can be used in conjunction with many other learning algorithms to improve their performance. AdaBoost is self-adaptive, in this sense; subsequent classifiers built are tweaked in favor of those instances misclassified by previous classifiers. AdaBoost is sensitive to noisy data and outliers. In some problems, however, it can be less susceptible to the over fitting problem than most learning algorithms. The usage of the classifiers can be weak (i.e., display a substantial error rate), but as long as their performance is not random (resulting in an error rate of 0.5 for binary classification), they will improve the final model. Even classifiers have an error rate higher than they're expected from a random classifier; it will be useful, since they will have negative coefficients in the final linear combination of classifiers they will behave like their inverses. In this paper, AdaBoost features face detection algorithm based on Haar-like is used.

2.2 Harr-Like Characteristics and Points of Calculation

Haar-like features are digital image features used in object recognition. They owe their name to their intuitive similarity with Haar wavelets and the Haar-like features were used in the first real-time face detector. Viola and Jones adopted the idea of Haar wavelets and developed the so-called Haar-like features [5]. A Haar-like feature considers adjacent rectangular regions at a specific location in a detection window, summing up the pixel intensities in each region and calculates the difference between these sums. This difference is then used to categorize subsections of an image.

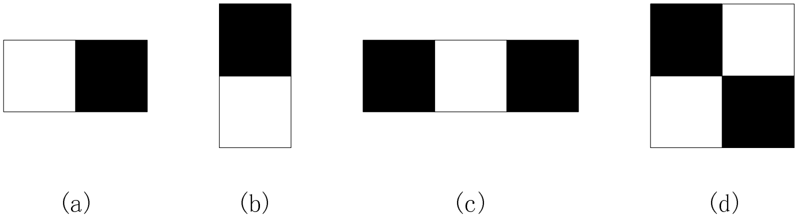
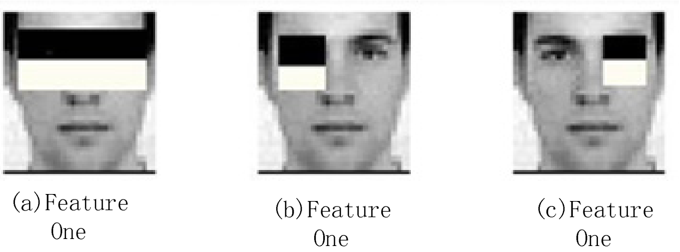


Fig. 1. Basic Harr-like features

Now many researches have been established based on the positive characteristics of Harr-like face detection system, in Fig.1 shows the four basic Haar-like rectangular features, the white areas and black areas, and the difference between the pixel gray characteristics for the Harr-like the characteristics of value [7]. Figure.2 shows typical facial features of Haar, application of these typical characteristics of the face can effectively separate the non-human face region.

Integral map is a transformation of the original image, which is defined as the following

$$\mathrm{ii(x,y)} = \sum_{x'<x.y'<y} i(x',y') \tag{1}$$



(a)Feature One (b)Feature One (c)Feature One

Fig. 2. The typical features of face in Harr-like

Visualization of the calculation process is shown in Fig. 3. Figure 3 points out the black dots in the figure [8], its value is equal to the original image from the upper left corner to the point (not including the black spots in the cross of the rows and columns) of all pixels and, in the gray area of figure. Figure and calculations indicate points in Fig. 3.

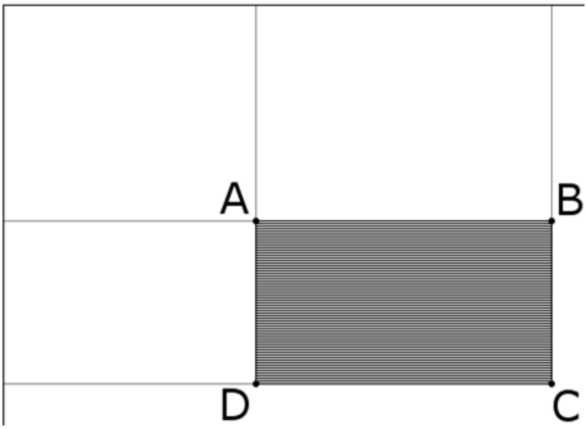


Fig. 3. Diagrams and calculation of plot points indicate

Using the points chart, any rectangle feature regional and use of the four reference points can be calculated. Figure 3 in the rectangular area ABCD pixel gray value uses A, B, C, D four reference points

$$\mathrm{sum} = \mathrm{I(C)} + \mathrm{I(A)} - \mathrm{I(B)} - \mathrm{I(D)} \tag{2}$$

Which are the gray pixels of the area of ABCD.

3 Adaboost Algorithm for Bottleneck Analysis

According to the algorithm theory, integral graphs and data classification algorithm is the critical data of the implementation. The face detection involves four layers from the inside to the outside loop: traverse the classification characteristics of each Harr and calculate the characteristic values of the cascade classifier traverse each level. One of the central part of the innermost of the two cycles (the execution time is mainly determined by the classifier Harr number of features in the detection window and progression through the classification), computing capacity and memory access frequency, which are difficult to achieve real-time requirements, therefore, Harr characteristics and the integral method of calculation is the bottleneck in a software implementation. Through testing, the calculation of integral figure accounted for about 40% of the detection time [9], and through algorithm analysis, the calculation of the relative integral map is more independent and can be used as a separate module to design the hardware acceleration.

4 The Hardware Implementation of Face Detection

4.1 System Structure

The ZYNq-7000 platform FPGA from XILINX is used in this paper. System structure is shown in figure.4. Xilinx Automotive-Grade (XA) Zynq-7000 All Programmable SoCs ideally address the technical and business challenges for one of the fastest growing automotive applications: Advanced Driver Assistance Systems (ADAS). The automotive-grade devices deliver unprecedented design flexibility as a single chip that combines a dual-core ARM Cortex-A9 processor, high-speed programmable I/O, and flexible programmable logic including DSP blocks for hardware acceleration of critical design components.

The increased system performance and highly integrated architecture of the Xilinx All Programmable SoCs reduce overall power and bill of materials (BOM) cost. The XA Zynq-7000 devices also enable end product differentiation with complete control of IP, and help system designers keep up with constantly changing feature requirements. Both engineers and business teams benefit from the Xilinx ecosystem that lowers time to market and trims life cycle costs to contribute to profitability.

4.2 The Hardware Design Face Detection

From the above analysis, a simple pure software solution can't achieve real-time requirements of Adaboost face detection on embedded platform. So this paper continues to try a new method in which hardware run the critical time-consuming part.

In the image RAM, the image is stored to be detected; the vertical integration logic is used for calculating the column integral and the column square integral; FIFO, is used to cache the value calculated by the column integral and the column square integral; According to the classifier of hierarchical cascade, Haar-like feature in RAM flows into the pipeline in turn; The small squares in the middle of hardware framework represent the data storage and the transfer unit, which flow from FIFO into the column integral and the column square integral one by one, and it is delivered to the right direction; some of the squares in the lower portion of frame stand for the total number of rectangular grayscale of weak classifier and the calculation unit of the image quadratic sum.

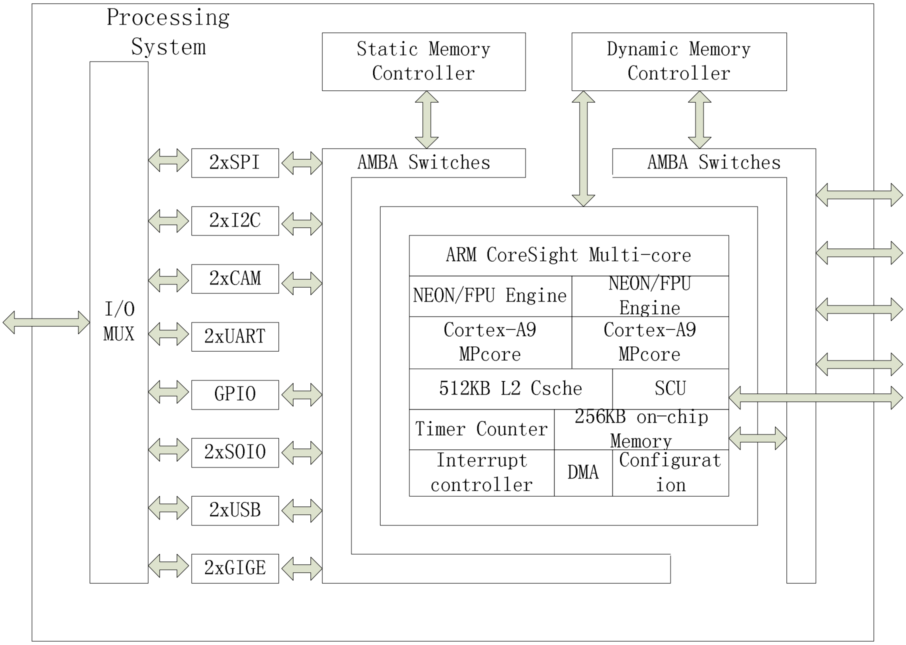


Fig. 4. System structure of ZYNq-7000 platform

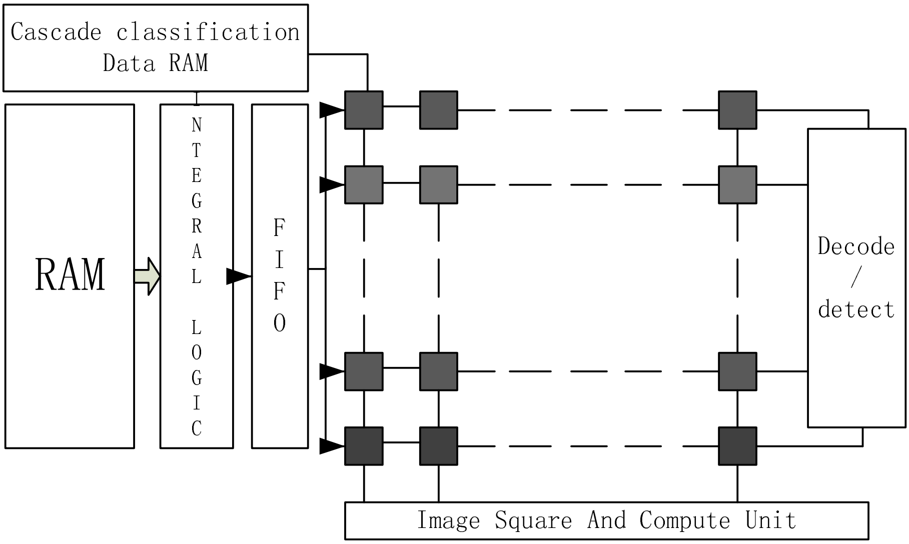

Cascade classification Data RAM
RAM
INTEGRAL LOGIC
FIFO
Decode / detect
Image Square And Compute Unit

Fig. 5. The framework of hardware design

5 Experiment Results

The system uses ARM Cortex-A9 processor ZYNq-7000 platform FPGA implementation of face detection from XILINX , with five data channels pipeline, hardware multiply and divide unit, 32-bit general registers, memory management unit (MMU) and a dedicated on-chip memory interface.

In this paper, the AdaBoost fixed-point processing algorithms in OpenCV are ported to FPGA platforms. The classifier using the OpenCV library provides the training data, a total of 25, containing 2913 Haar features. Detection result is shown in figure 6:

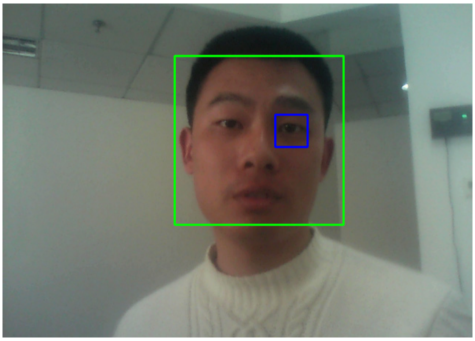


Fig. 6. Detection results

Through the time-consuming analysis of algorithms, and in accordance with the characteristics of FPGA resources, parallel architecture is most conducive to be realized, and the speed can be multiplied for many times. Partial function is used to help the experiment of hardware acceleration, and has a real-time 17fbs detection effect. The final results show that it can detect face sat a 17fps speed with high hit rate and low false detection rate.

6 Conclusion

This paper aims to analyze the bottlenecks of AdaBoost face detection algorithm software. In the FPGA platform, it takes the advantages of parallel processing hardware to reduce hardware size; to grayscale in the form of flowing water to improve the detection rate; to fully account access memory access bandwidth and storage optimization to enhance the system's processing performance, etc., and finally ultimately realizes high precision real-time face detection.

Acknowledgments. The authors would like to acknowledge the financial support by Shanghai Annual Plan for Absorption and Innovation of Technology Imported under Grant No. 11XI-15.

References

1. Podlubny, I.: Fractional differential equations. Academic, New York (1999)
2. Hartley, T.T., Lorenzo, C.F., Qammer, H.K.: Chaos in a fractional order Chua's system. IEEE Trans. CAS-I 42, 485–490 (1995)
3. Petrá, I.: Method for simulation of the fractional order chaotic systems. Acta Montanistica Slovaca 11(4), 273–277 (2006)
4. Wikipedia, http://en.wikipedia.org/wiki/Adaboost
5. Wikipedia, http://en.wikipedia.org/wiki/Haar-like_features
6. Chen, G., Lvjin, H.: Lorenz system family dynamics analysis, control and synchronization of. Science Press, Beijing (2003)
7. Chen, G., Yu, X.: Chaos control: theory and applications. LNCIS, vol. 292. Springer, Heidelberg (2003)
8. Tavazoei, M.S., Haeri, M.: Unreliability of frequency-domain approximation in recognising chaos in fractional-order systems. IET Sig. Process 1(4), 171–181 (2007)

Remote Sensing Image Segmentation Based on Rough Entropy

Hui-jie Sun[1,2], Ting-quan Deng[1], and Ying-ying Jiao[1]

[1] College of Computer Science and Technology, Harbin Engineering University,
Harbin Heilongjiang 150001, China
[2] College of Computer Science and Information Engineering, Harbin Normal University,
Harbin Heilongjiang 150025, China
{Hui-jieSun,Ting-quanDeng,ying-yingJiao,sunh858}@163.com

Abstract. Remote sensing image segmentation algorithms are proposed for different thresholds with rough sets theory and fuzzy sets theory in this paper. The target and background fuzzy sets are gotten with the gray image as a fuzzy sets ; The target and background fuzzy sets are approximated by two rough fuzzy sets, the optimal image segmentation threshold is chosen by the optimal standard, Experimental results show that the proposed algorithms are more effective and flexible.

Keywords: Rough Entropy, Threshold Segmentation, Remote Sensing Image.

1 Introduction

Rough set theory [1] is professor Pawlak proposed a research incomplete, uncertain knowledge and data expression, learning and induction theory and method. Is widely used in granular computing and data mining field [1-4]. Its main idea is based on the indiscernibility relations, according to a given problem has the knowledge of the problem domain, determine the division of a concept of degree of support. Due to the uncertainty of the image, the image in solving problem has certain advantages, especially in image enhancement, image filtering and image target feature extraction (image segmentation) has significant effect. 2005 Pal etc [5] combined with particle size calculation proposed rough entropy, and applied to the image of the target detection, improve the image segmentation effect. This paper will be based on the improved accordingly, this paper puts forward a rough entropy based on the remote sensing image segmentation algorithm, the simulation results verify the effectiveness of the proposed algorithm.

2 Image Threshold Segmentation Algorithm on Rough Entropy

2.1 The Image of the Rough Set

Pal combined with particle size calculation theory and rough entropy, this paper presents a threshold segmentation algorithm. The image pixels set as theory field U, U

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 411–419, 2013.

will be divided into the size of the overlap each other for little window set, a total of a small image, each division of small window as a grain of Gi, also called G = {Gi} for U of a particle size classification. For any pixel p∈U, there is only one grain of Gp meet p∈Gp, which can be used to define the target area of the upper and lower approximation for:

$$\underline{O}_T = \bigcup_{i=1}^{K} \{G_i \mid H(p_j) \geq T, \forall j = 1,2,\ldots,mn, \; p_j is \text{ a pixel of } G_i\} \tag{1}$$

$$\overline{O}_T = \bigcup_{i=1}^{K} \{G_i \mid \exists j \in \{1,2,\ldots,mn\}, \; s.t. H(p_j) \geq T, \; p_j is \text{ a pixel of } \} \tag{2}$$

2.2 The Image of the Rough Entropy Measurement and Threshold Segmentation Algorithm

Def 1: for gray image H, give the domain U on the particle size classification $\{G_p \mid p \in U\}$ and threshold T, may be defined target for roughness

$$\rho_{O_T} = 1 - \frac{|\underline{O}_T|}{|\overline{O}_T|} = \frac{|\overline{O}_T| - |\underline{O}_T|}{|\overline{O}_T|} \tag{3}$$

Similarly, may be defined as the background roughness

$$\rho_{B_T} = 1 - \frac{|\underline{B}_T|}{|\overline{B}_T|} = \frac{|\overline{B}_T| - |\underline{B}_T|}{|\overline{B}_T|} \tag{4}$$

In the same image, the boundary is a common goal and background, namely

$$\overline{O}_T - \underline{O}_T = \overline{B}_T - \underline{B}_T = Q_T$$

Q_T the edge boundaries, so we can write the roughness of target and background for Def1:

$$\rho_{O_T} = 1 - \frac{|\underline{O}_T|}{|\overline{O}_T|} = \frac{|\overline{O}_T| - |\underline{O}_T|}{|\overline{O}_T|} = \frac{|Q_T|}{|\overline{O}_T|} \tag{5}$$

$$\rho_{B_T} = 1 - \frac{|\underline{B}_T|}{|\overline{B}_T|} = \frac{|\overline{B}_T| - |\underline{B}_T|}{|\overline{B}_T|} = \frac{|Q_T|}{|\overline{B}_T|} \tag{6}$$

Further, changing of threshold T for the roughness of target and background :

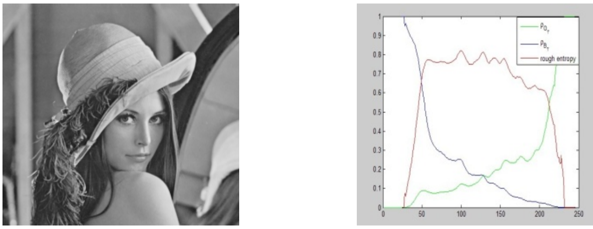
Theorem 1[5]. $|\underline{O}_T|$、$|\overline{O}_T|$ smaller ; $|\underline{B}_T|$、$|\overline{B}_T|$ bigger as T increasing, and

(1) ρ_{O_T} is bigger, ρ_{B_T} is smaller if the boundary $|Q_T|$ is not changed ;

(2) ρ_{O_T} is bigger if the boundary $|Q_T|$ is bigger;

(3) ρ_{B_T} is smaller if boundary $|Q_T|$ is smaller;

But to pay attention : if T increases, the $|Q_T|$ is smaller, and cannot be completely out of ρ_{O_T} is bigger, the change of ρ_{O_T} depends on the target interior point changes and border changes.



(a) The sample (b) Roughness of the diagram

Fig. 1. Target and background roughness and rough entropy with threshold change curve diagram

So, the available ρ_{O_T} in general with the threshold value increases, ρ_{B_T} have opposite change trend of the conclusion. Figure 1 (a) as an example, calculate the threshold value in each in the target and background roughness value, the result see figure 1 (b), figure 1 (b) shows its intuitive change trend, the green line is target roughness with threshold change change curve, blue line is background roughness change curve. It can be intuitive validation roughness and the change of threshold value relationship.

Def 2[6]: The rough entropy of the grey image H is defined:

$$E(H) = -\frac{e}{2}[\rho_{O_T} \log_e(\rho_{O_T}) + \rho_{B_T} \log_e(\rho_{B_T})] \tag{7}$$

The algorithm of the rough entropy image threshold segmentation [5] is given by Pal that take the maximum entropy as the corresponding threshold is needed for optimal thresholds, think the goal roughness and background roughness minimum. Pal image as a set of pixels, that is, from set domain (spatial domain) point of view image of rough set said method, when the roughness of the target and the background to the hours, also is the minimum edge boundaries, and edge detection in image segmentation or hope to get in the boundary of the thinner the better, had better be single pixel set. So the target and background roughness minimum can get optimal thresholds. But Pal about rough entropy and roughness of the relationship between interpretation by the theorem 1 analysis knowledge is not correct. Since the target and background between roughness with threshold value bigger and have opposite change tendency, is balanced binary image boundary effect, combined with the maximum entropy principle knowledge, take the maximum entropy can get optimal solution, to get the optimal image segmentation threshold. Figure 1 (b) red's sample image in the Def 2 and the rough entropy with threshold value change curve.

The algorithm is improved [7] by Pal in 2009, from image contains ambiquity perspective, the grey level as theory field, consider domain with characteristics in a fuzzy set, the use of rough entropy measure image gray ambiquity; Image pixels as theory field, considering the gray value has some characteristics of the fuzzy set, and gives the fuzzy space of the measurement method, and its application to image processing.

The improvement of Pal is more reasonable for the application of rough set theory in image processing, but how to choose the good characteristics and its representation of a form of membership function with certain subjective artificial. Next day the whole image as a fuzzy set, this paper discusses the image fuzzy rough set representation model.

2.3 The Expression for the Fuzzy Rough of Image

This section will image as a fuzzy set, from gamma curve perspective view image, considering the target image from the extracted. For a image H of $J\times K$, the image grey level is [0,255], the domain into a interval [0,1], use the same granulating method for image spatial domain for granulating, get the target and the background image of the upper and lower approximate as follows.

Def 3: given the grey image *H* and domain *U* ,the division of granularity is $\{G_p \mid p\in U\}$, $T\in[0,1]$, for any $p\in U$, we define the lower approximation of *H* for the image target:

$$\underline{O}_T(H)(p)=\begin{cases}\min_{q\in G_p} H(q), & \min_{q\in G_p} H(q)\geq T\\ H(p), & otherwise\end{cases} \tag{8}$$

The upper approximation of H for the image target:

$$\overline{O}_T(H)(p)=\begin{cases}\max_{q\in G_p} H(q), & \max_{q\in G_p} H(q)\geq T\\ H(p), & otherwise\end{cases} \tag{9}$$

the lower approximation of H for the image background:

$$\underline{B}_T(H)(p)=\begin{cases}\min_{q\in G_p} H(q), & \max_{q\in G_p} H(q)< T\\ H(p), & otherwise\end{cases} \tag{10}$$

The upper approximation of H for the image background:

$$\overline{B}_T(H)(p)=\begin{cases}\max_{q\in G_p} H(q), & \min_{q\in G_p} H(q)< T\\ H(p), & otherwise\end{cases} \tag{11}$$

2.4 The Rough Entropy of the Image and the Algorithm of Threshold Segmentation

Roughness of rough set is rough degree measure an effective tool, the following is target and background of rough fuzzy set the roughness of the definition.

Def 4: given the division of granularity $\{G_p \mid p \in U\}$ on domain U for the grey image H, $T \in [0,1]$, for any $p \in U$, we define the lower approximation of H for the image target and Target rough fuzzy set of roughness, recorded as target roughness, defined as:

$$\rho_{O_T}(H) = 1 - \frac{M(\underline{O}_T(H))}{M(\overline{O}_T(H))} \tag{12}$$

Similarly, the background of rough fuzzy set, remember the roughness of the background of the roughness, defined as:

$$\rho_{B_T}(H) = 1 - \frac{M(\underline{B}_T(H))}{M(\overline{B}_T(H))} \tag{13}$$

And $M(\cdot)$ is a measure of fuzzy set.

Obviously, $\rho_{O_T}(H)$ 、 $\rho_{B_T}(H) \in [0,1]$ .To $(\underline{O}_T(H), \overline{O}_T(H))$ as a example, the value of membership value of $\{\min_{q \in G_p} H(q) \mid p \in U\} >> \{H(p) \mid p \in U\}$, when $\rho_{O_T}(H)$ closes to 1, almost non-existent 0 element in $Q_O(H)$, similarly there is a the same conclusion in $\rho_{B_T}(H)$. At this time, there is a bigger difference for $\underline{O}_T(H)$ and $\overline{O}_T(H)$.

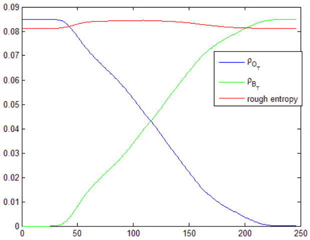


Fig. 2. The curve diagram of image 1 *(a)* in the fuzzy rough set model target roughness and background roughness and rough entropy with threshold changed

Still to Figure 1 (a) as an example, calculates the fuzzy rough image model, each threshold value in the target and background roughness, figure 2 shows its intuitive change trend, the blue line is target roughness with threshold change change curve, and the green line is background roughness change curve.

Rough set is used to target approximation, the approximation degree usually use upper and lower approximation to show. Here want to use fuzzy rough set image model good description image, but it should be noted at the target area (background area can also be seen as a target) with image are relatively flat, and the difference between the lower approximation is small; And at the edge of the area, the difference between the lower approximation is more big, in the image segmentation, hope to find a threshold value causes the upper and lower approximate difference between on the edge as far as possible little. But target roughness and background roughness appears to reverse the trend, you need a metric to balance $\rho_O(H)(T)$ and $\rho_B(H)(T)$, to achieve a satisfactory segmentation effect. Entropy is a good measurement method, here the rough entropy to balance the target and background roughness.

For the domain U in the vague definition set X, according to the references [8] given fuzzy entropy, gives a class definition rough entropy as follows:

$$E(X) = \frac{1}{2}[\upsilon(X) + \upsilon(X^c)] \tag{14}$$

$\upsilon(D) = \rho_R(D)(2-\rho_R(D))$ for any $D \subseteq U$, X^c is a Complement of X, we can get the diffrent rough entropy for $\rho_R(X)$ and $\rho_R(X^c)$ through defined the different upper and lower approximation.

We can rewrite the type for this form in order to better understand the rough entropy:

$$E(A,B) = \frac{1}{2}[A(2-A) + B(2-B)] \tag{15}$$

$A \in [0,1]$ and $B \in [0,1]$ said respectively $\rho_R(X)$ and $\rho_R(X^c)$ .Figure 3 is the three dimensional diagram for entrop. This kind of entropy is to define two mentioned entropy was revised, from the chart that the entropy more accord with human cognitive intuition, but consistent with uncertainty.

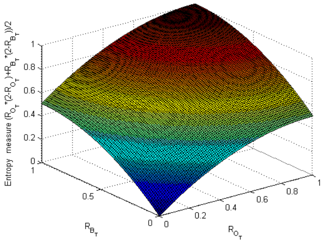


Fig. 3. The three dimensional diagram for entropy

Def 5: given the division of granularity $\{G_p \mid p \in U\}$ on domain U and threshold T for the grey image H ,the rough entropy is defined as:

$$E(T) = \frac{1}{2}[\rho_O(H)(T)(2-\rho_O(H)(T)) \\ +\rho_B(H)(T)(2-\rho_B(H)(T))] \tag{16}$$

$E(T)$ is a function for T and we known, Target roughness $\rho_O(H)(T)$ is about t decreasing function, a background roughness is about $\rho_B(H)(T)$ increasing function; And the threshold is lesser, target roughness is lesser, background roughness larger. Definition 4 and definition of rough entropy is target roughness and background of the roughness of a comprehensive. In order to better analysis of the rough entropy and image segmentation threshold value of the relationship, it will be rewritten into the following form:

$$E(T) = \frac{1}{2}[A(T)(2-A(T)) + B(T)(2-B(T))] \tag{17}$$

$A(T)$ is about T decreasing function and $B(T)$ is about T increasing function..

3 The Analysis of Experimental Results

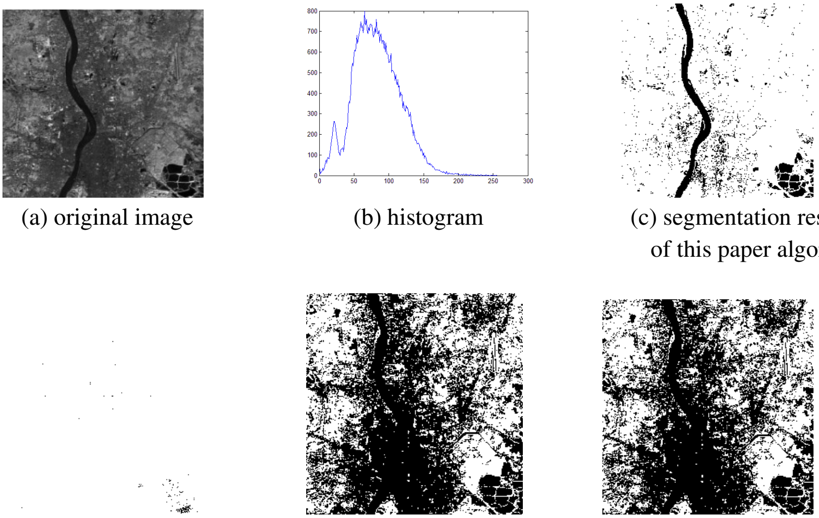
In the MATLAB of this algorithm simulation. In order to see this chapter proposed algorithm experimental effect and generality, in experiments, the selection of remote sensing images. Using references [9] the most kinds of variance between law and references [10] fuzzy entropy method and the references [5] Pal method to carry out the contrast.

The most kinds of variance between method is a classical threshold segmentation method, its principle is to consider the background image is divided into two parts and prospects, and separately calculated them in a certain threshold value of the points, quality moment and average gray level, the best threshold value selection should make the biggest difference between foreground and background, choose the most kinds of variance between here as a standard. [10] fuzzy entropy method is the iconicity into a interval-valued fuzzy sets, compared with the traditional fuzzy set, avoid to determine the membership function, this method according to the image histogram to determine fuzzification factor, the choice of appropriate main membership function of the image of the interval-valued fuzzy sets said, through to the interval-valued fuzzy sets minimum fuzzy entropy image segmentation for the optimal threshold value, in the experiments in order to writing is convenient, its simple notes for the new fuzzy entropy method.

In this paper the algorithm and the particle size Pal method are elected, this paper algorithm roughness measurement choose area to carry out the measure of value, the choice of measurement and similar area. The experimental results (see figure 4).

The images will because external light, the photography equipment external reasons or boundary, texture and so on some concepts or knowledge of the uncertainty is defined and ambiquity; Its itself but also because of gray gradient can produce fuzzy edge, neighboring pixels or neighboring gray level tend to have the rough similarity, these to image processing and object extraction increased the difficulty, but also from other aspects that the fuzzy set and rough set is introduced into the image of the necessity and effectiveness.

Remote sensing image is basically in the form of digital image obtained by the influence of the resolution, usually get image contains some kind of noise structure. Figure 4 (a) is a remote sensing image, using the above four segmentation algorithm on the processing, the hope can get the image of the black part of the path. Experimental get segmentation threshold for: 43 (algorithm), 5 (Pal method), 85 (Otsu method), 85 (the new fuzzy entropy method). From 4 (c) - 4 (f) can intuitive see, this algorithm can obtain better segmentation result, less influenced by environmental noise influence of the structure of the class. But need to pay attention to is the result of the influence of particle size by, take larger size, also will be not satisfactory segmentation results.



(a) original image

(b) histogram

(c) segmentation results of this paper algorithm

(d) segmentation results of Pal

(e) segmentation results of Otsu

(f) segmentation results of new Rough etropy

Fig. 4. Different remote sensing image threshold segmentation algorithm comparison chart

4 Conclusion

In this paper fuzzy set is introduced into the image of the rough set representation model, the image as a fuzzy set, get target fuzzy set and fuzzy set background, respectively, using two rough fuzzy set to approximate description target and background, get image fuzzy rough representation model. Using rough entropy to balance the target and background, lower approximation in the border difference, according to the principle of maximum entropy image segmentation to determine the optimal threshold value. Through the text images, medical image and remote sensing image contrast experiments show that the algorithm can be very good to adapt to the image itself contains ambiquity, segmentation effect is better than that of Pal method and other threshold segmentation algorithm.

References

1. Pawlak, Z.: Rough Sets. International Journal of Parallel Programming 11, 341–356 (1982)
2. Pawlak, Z.: Rough Sets and Fuzzy Sets. Fuzzy Sets and Systems 17(1), 99–102 (1985)
3. Pawlak, Z.: Rough Sets. In: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
4. Zhang, W., Wu, W., Liang, J.: Rough Sets Theory and Method. Science Publishers, Beijing (2001)
5. Sheng, C.D.: Intelligent Image Segmentation Methods Based on Variable Precision Rough Entropy, A Dissertation for the Degree of M.Sci of Harbin Engineering University (2010)

6. Pal, S.K., Shankar, B.U., Mitra, P.: Granular Computing Rough Entropy and Object Extraction. Pattern Recognition 26(16), 2509–2517 (2006)
7. Sen, D., Pal, S.K.: Generalized Rough Sets, Entropy and Image Ambiguity Measures. IEEE Transactions on System, Man, and Cybernetics-Part B: Cybernetics 39, 117–128 (2009)
8. Liu, X.C.: Entropy, Distance Measure and Similarity Measure of Fuzzy Sets and Their Relations. Fuzzy Sets and Systems. 52, 305–318 (1992)
9. Otsu, N.: A Threshold Selection Method from Gray-level Histograms. Systems, Man and Cybernetics, 62–66 (1979)
10. Deng, T.Q., Wang, P.P., Mei, Y.L.: Thresholding Approaches with Interval-valued Fuzzy Sets to Image Segmentation. In: Proceedings of the 3rd International Conference on Intelligent System and Knowledge Engineering, pp. 1059–1064 (2008)

A Real-Time Noise Image Edge Detector Based on FPGA

Meihua Xu[1], Chenjun Xia[1], and Shuping Huang[2]

[1] Microelectronic Research and Development Center, Shanghai University, Shanghai, China
[2] School of Mechatronics Engineering and Automation, Shanghai University, Shanghai, China
{Mhxu,iversn}@shu.edu.cn, xiachenjun0403@126.com

Abstract. This paper describes a real-time noisy image edge detector to remove the noise which will bring negative effects on extraction and detection of image features. The average filtering algorithm is used to eliminate the noise of the original image and the Sobel edge detection operator is used to obtain image data. Both of the operation completes the functions including image acquisition and processing. Feasible verification of the edge detector is implemented in Altera EP3C55 using Verilog HDL language. Experimental results show that the edge detector is adaptive to the environment and can extract the noisy image edge effectively and promptly.

Keywords: average filtering, edge detecting, noisy image, Sobel, FPGA, dual threshold.

1 Introduction

At present, the image processing technology is a hotspot of the algorithmic research, and the face recognition technology is an important part of image processing. The public security departments can use this technology to identify criminals hid in the crowd. However, the process that gathering and transferring image by monitor system will inevitable produce image noise signal. Noise will have a serious impact on complex image recognition operation. In order to solve this problem, we design a real-time image de-noising and edge detector system based on FPGA. It can filtering image noise and detects image edge effectively. This system will reach a basis for the subsequent face detection system.

2 Algorithm Design

2.1 Average Filtering Algorithm

There are many kinds of image noises, and the commonly used image de-noising algorithms include median filtering, average filtering, adaptive wiener filtering, morphological filtering and wavelet filtering. The median filtering and average filtering are the most commonly used filtering algorithms, median filter can drastically remove impulse noise, and it does not reduce the feature of image edge. However, if images appear a large area of noise, such as white noise distributed by Gauss, the smoothing

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 420–427, 2013.

noise ability of average filtering is better than the median filtering based on the mean square error rule. In addition, the median filtering has another defect. When the system need to running at a high-speed, median filtering will consume more clock cycle. This is because the median filtering algorithm needs to sort the pixels, and then select the pixel in the middle of the sequence to replace the center pixel, This ranking process will take many operations, reducing the working speed. However, the average filtering algorithm will avoid this situation, so we chose it as the filtering algorithm.

This design uses the arithmetic average filtering algorithm. We regard Z [I, J] as the coordinates which center is the point (I, J) and the range of coordinate is m×n image window. In the area defined by Z [I, J], arithmetic average filtering calculates the contaminated image and figures out the average value of p [I, J]. At last, we can use the new value to replace the old center pixel, the formula is as below:

$$p' = \frac{1}{mn} \sum_{(s,t)\in Z[i,j]} p(s,t) \tag{1}$$

The m and n in this design are three, average filtering will make the image become blurred, but noise will be dropped substantially.

2.2 Edge Detection Algorithm

When an image was grayed, the main information of the original image exists in the edge of the new image. We grayed an image and extract the gradient of it to achieve the effective dada. Eventually, we can detect the image edge by the threshold value judgment. Roberts, Prewitt, Sobel, and Canny operators are often used to realize this algorithm.

This design uses Sobel operator because it is a high-performance way. The detection window is as below:

$$\begin{matrix} S_{00} & S_{01} & S_{02} \\ S_{10} & S_{11} & S_{12} \\ S_{20} & S_{21} & S_{22} \end{matrix} \tag{2}$$

The transverse and vertical axis direction gradients of the center pixels are:

$$\begin{gathered} G_x = (S_{02} + 2\,S_{12} + S_{22}) - (S_{00} + 2\,S_{10} + S_{20}) \\ G_y = (S_{00} + 2\,S_{01} + S_{02}) - (S_{20} + 2\,S_{21} + S_{22}) \end{gathered} \tag{3}$$

Total gradient expression is:

$$G = \sqrt{{G_x}^2 + {G_y}^2} \tag{4}$$

We can compare the total gradient and the threshold value to judge whether the point is the edge point.

3 System Realization

3.1 Hardware Platform

This design is a part of the intelligent vehicle identification project, so we use the same hardware platform. The platforms comprised of Camera_MT9M111camera, two pieces of Samsung K4S561633F DRAM, cycloneIII FPGA chip, Chrontel CH7301-TF video decoding chip and related configuration circuit, detailed parameters are shown in table 1:

Table 1. Parts of hardware platform

Chips	Type	Parameter
FPGA	CycloneⅢ EP3C55F48417N	55856Logic elements 2396160 bit RAM
DVI transmitter	Chrontel CH7301C	165M pixel/second Up to 1600×1200 pixels
SDRAM	SUMSUNG K4S561633F	4M×16bit×4Banks 105MHz Max Frequency
Image	MICRON	Up to 1280H×1024V
Sensor	MT9M111	27Mps/54MHz

3.2 System Architecture

The core part of the system is a piece of Altera FPGA chip. its main parameters have been listed in table 1. The system adopts the modularized design. Every module only realizes single function, so we can move them to other hardware platform conveniently. As shown in figure 1, the large rectangular frame in picture is FPGA part, and the rest of them are peripherals.

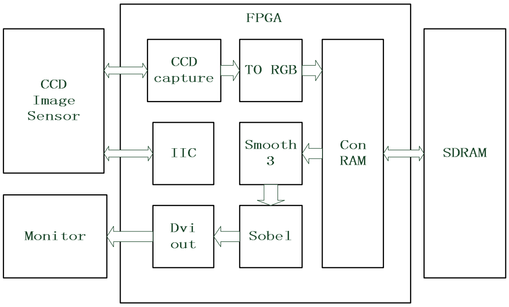


Fig. 1. System architecture

Among them, the CCD_CAPTURE is responsible for the data transmission between FPGA and the camera. The IIC module is used to match their speed. When the CCD_CAPTURE is collecting data, we need to transform image format to RGB mode and divided data into (red, green) and (blue, green) parts. Every part is 12bits data stored in SDRAM. This process needs CON_ RAM module to coordinate. The display sends the READY signal to DVI_OUT module. The SMOOTH3, average filtering module, began to work. It reads green part of the data from the SDRAM. When filtering completed, the SMOOTH3 wake up the SOBEL module by the handshake signal. Then the SOBEL module calculates the edge of the image. At last, it passes data to DVI_OUT module to display. The flow chart of the system operation is shown in figure 2.

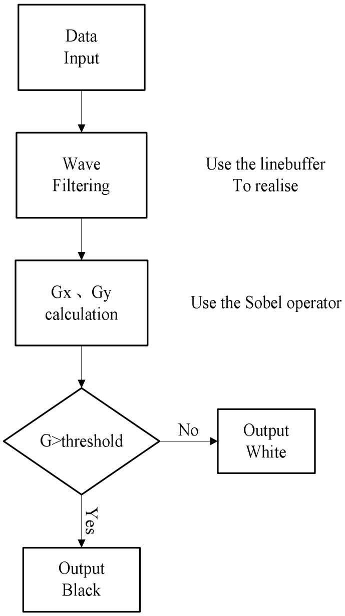


Fig. 2. Flow chart of the system operation

3.3 Achievement of Average Filtering Algorithm

We have briefly described the average filtering algorithm and then we will show how to realize it in FPGA chip. Because the concept of physical address does not exist in the Verilog HDL, we put forward a measure by using the Linebuffer module to solve this problem in image processing. For example: We can regard it as a buffer to store the data which will be used in 640th pixel. In the design, we set the number of lines to 3and put the number of columns to 640. This is convenient for the image operation with matrix. We remove the central pixel of a 3 × 3matrix, sum up the other pixels and divide eight. We use the result to replace the central of the matrix.

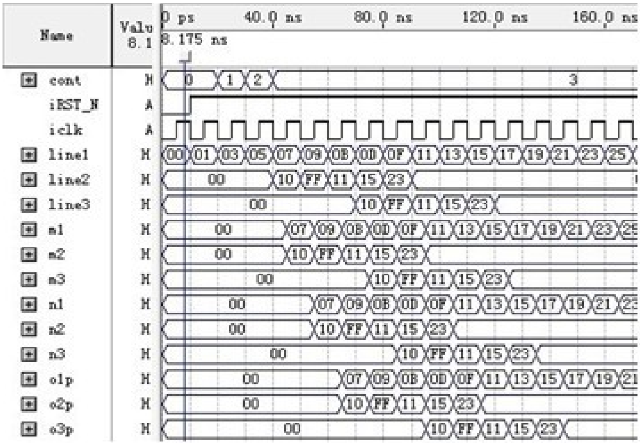


Fig. 3. Waveform simulation diagram of three Line shift register

The three Line shift register is the core of the module. If we set the number of columns to 5, then its waveform simulation shown in figure 3. Schematic diagram for the average filtering module shown in figure 4:

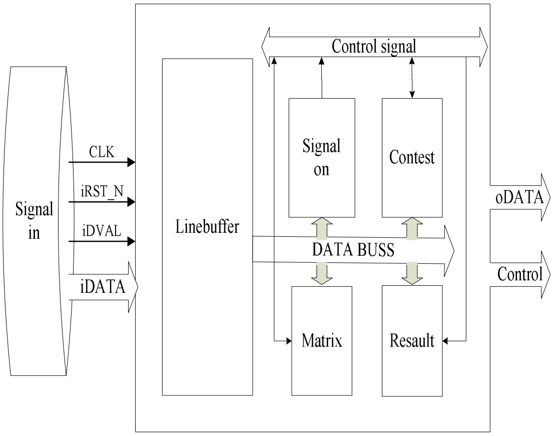


Fig. 4. The architecture of smooth3

3.4 Achievement of Sobel Edge Detector

When average filtering completed, the noise of the data greatly reduced. The Sobel edge detection operation is running in this part. But the three Line shift register which was used in SMOOTH3 will be used again. In this module, the most important task is figure out the Gx and Gy, the solution of Gx was described as below:

When the camera is gathering the real-time data, the matrix in the upper left corner just like a small river, data deposit on the matrix is always changing. The matrix move from the left to the right of the whole image, from line n to line n+1. The name of matrix on the upper right corner is always the same, but it is always collecting data from the left matrix. At last calculates data according to the method shown in the figure 5.

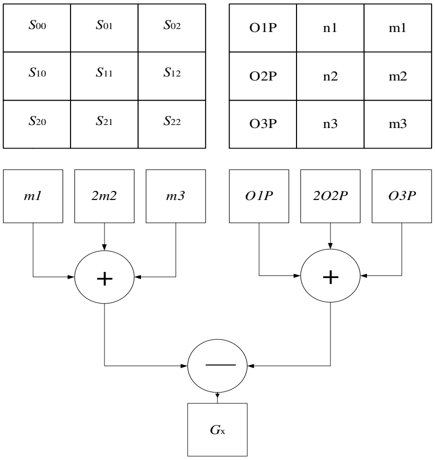


Fig. 5. The solution of Gx

This design uses the double threshold method, when the value of G is greater than or less than the two extreme values, we don't think that is the edge point. When debugging the system, we found some edge points are not the real one. They are just some high gradient noise points. Generally, the gradient of real edge point is not very high. So this kind of point can be ignored. The Single threshold edge detection waveform simulation results shown in figure 6:

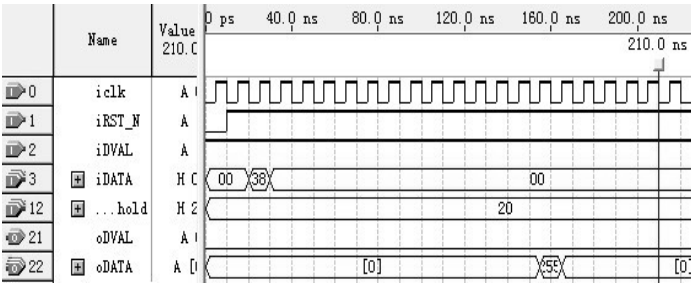


Fig. 6. Waveform simulation diagram of Sobel

4 Experiment Result

We put the three pictures for the experiment result, the effect is obviously.

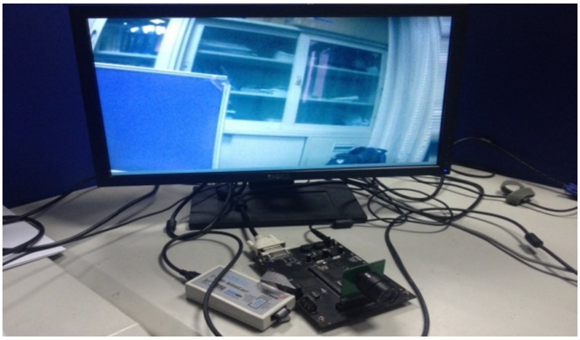


Fig. 7. Original image

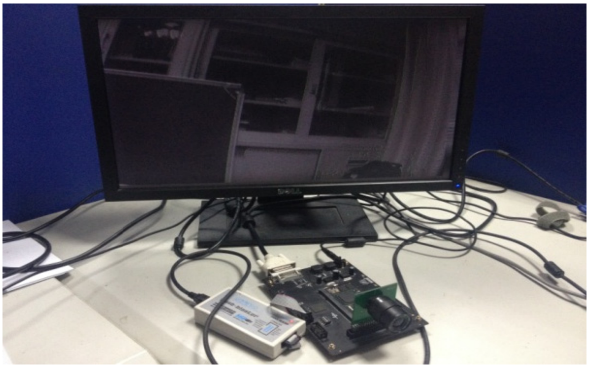


Fig. 8. Image after filtering

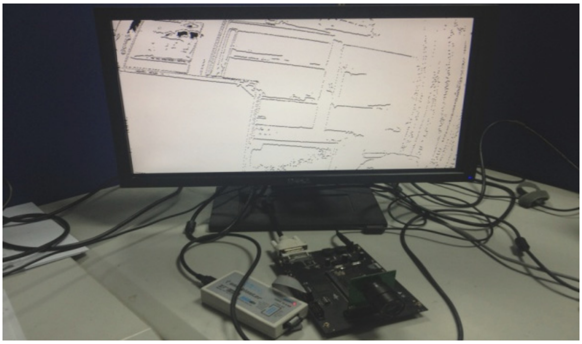


Fig. 9. Final image

By comparing the quality of the two different pictures, we can see in the original image of figure 7and it has some noise points. The figure 9 is the image after filtering and edge detecting. The result is good.

We can compare the system with the pc. We use a computer to realize the same function by the matlab and observe the FPGA board speed by the oscilloscope. The performances of them are as table 2:

Table 2. The speed of pc and FPGA

System platform	Read data time (ms)	average filtering time (ms)	Sobel time (ms)	Total time (ms)
PC (matlab)	60.9	47.6	405.7	514.2
FPGA (Altera)	9.2	6.4	17.2	32.8

Because of the parallel processing characteristic of FPGA, we found that it runs faster than PCs which relying on code execution.

Acknowledgments. The authors would like to acknowledge the financial support by Shanghai Annual Plan for Absorption and Innovation of Technology Imported under Grant No. 11XI-15.

References

[1] Palnitkar, S.: Verilog HDL: A Guide to Digital Design and Synthesis. Publish House of Electronics Industry, Beijing (2009)

[2] Wang, K., Xin, Y.: Efficient filtering algorithm. Application Research of Computers. SiChuan, (2010)

[3] Yue, L.: Research and Hardware Design of Low Illumination Image Denoising Algorithm. Taiyuan University of Technology, ShanXi (2011)

[4] Hu, J.: Implementation of Video Capture Output System based on FPGA. Maritime Affairs University of Dalian, DaLian (2011)

[5] Wolf, W.: Modern VLSI Design: IP-Based Design. Publish House of Electronics Industry, Beijing (2011)

[6] Wang, X.: A Dissertation Submitted to Shanghai Jiao Tong University for. Shanghai Jiao Tong University, Shanghai (2009)

[7] Jun, Q.: A New ImProved Filtering Algorithm Based on Median Filter. Beijing University of Posts and Telecommunications, Beijing (2010)

[8] Zhu, Z.: APP lieation Research on Median Filtering Teehniquein. Northeastern University, Shenyang (2008)

[9] Rafael, C.G.: Digital Image Processing. Publish House of Electronics Industry, Beijing (2011)

[10] Karin, S.: Tapering of Multit ransmit. IEEE Trans. on Antennas 2, 830–833 (2008)

Optimization Algorithm and Implementation of Pedestrian Detection

Meihua Xu[1], Huaimeng Zheng[1], and Tao Wang[2]

[1] Microelectronic Research and Development Center, Shanghai University, Shanghai, China
mhxu@shu.edu.cn，huaimeng.happy@163.com
[2] School of Mechatronics Engineering and Automation, Shanghai University, Shanghai, China
alada@shu.edu.cn

Abstract. Pedestrian detection is widely used in automotive assisting driving system. The algorithm based on Histograms of Oriented Gradient (HOG for short) feature is the main one in the current pedestrian detection. This paper uses tri-linear interpolation method to extract the image HOG features, and gives the optimization algorithm based on look-up table to reduce the amount of calculation in extracting HOG feature. And then classifies them by RBF and linear SVM to explore its speed and accuracy. At the end of the paper, an effective method is given to merge windows that contain detected pedestrians. Experiments on INRIA and MIT databases show that the detecting accuracy and speed of this method is relatively high.

Keywords: pedestrian detection, Histograms of Oriented Gradient, SVM, window's merging, tri-linear interpolation.

1 Introduction

Pedestrian detection in automotive assisting driving system [1] has important significance and practical value and aims at building an intelligent assisting driving system to ensure the safety of driving and the security of pedestrians' life and property. However, it is rather difficult to make an effective pedestrian detection due to the influences caused by light, changeable postures of the pedestrian, visual angle, and the diversification of the clothing and background. How to detect pedestrians effectively, rapidly and accurately from videos or images is still a hot research topic.

Currently the pedestrian detection method [2] mainly includes the conventional methods and the methods that are based on image features. The former detects the pedestrian according to the physically visual information such as the shape, size, color, the distance of the detection target and priori knowledge, but all of these methods are sensitive to the noise, the changes of the background, and the pedestrians with varied postures. Among the conventional methods, the main algorithm includes the method based on shape matching, the method based on optical flow and the model-based method. The latter based on the application of the principle of statistics can transform the pedestrian detection into feature classification and eventually put

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 428–435, 2013.

the feature into classifiers for further machine learning and training. As a research focus in recent years, this method can effectively overcome the defect and deficiency in traditional methods, and is robust for more complicated background, the diversity of the gestures of human and the noise. Among these methods, the most representative one is the method proposed by Navneet Dalal which is based on Histograms of Oriented Gradient [3] features and Support Vector Machines [4] (SVM). In this paper, the author selects the sliding window with the size of 128×64, block with the size of 16×16, cell with the size of 8×8 and bin with nine directions. Each time the block or sliding window moves 8 pixels. After each step of sliding window, a feature vector with 3780 dimension can be gained and it can be sent into the Support Vector Machines for classification or identification. At the end of the detection, the windows that contain pedestrians shall be merged to get the final detecting results. In order to reduce the calculation amount in extracting HOG features, optimization algorithm based on look-up table is adopted to calculate the feature vector of the sliding window.

2 Extraction of HOG Feature

Extracting the HOG feature of an image [5] includes the following steps: (1) calculate the gradient in the directions of X and Y, the gradient magnitude and direction of each pixel with the operators of [-1, 0, 1] and $[-1, 0, 1]^T$; (2) get the 36-dimensional feature vectors of each block by using the method of tri-linear interpolation; (3) normalize the block feature vector to eliminate the influences caused by the light, a small bias etc.; (4) merge all the block features in each sliding window into one 3780-dimensional vector , i.e. , the feature vector of the sliding window .

2.1 Pixel Gradient of Gray-Scale Image

If H represents a gray-scale image, $\mathrm{Hx,y}$is the gray value at the position $\mathrm{x,y}$in the image, then the gradient at the pixel is:

$$\mathrm{G_x(x,y) = H(x+1,y) - H(x-1,y).} \tag{1}$$

$$\mathrm{G_y(x,y) = H(x,y+1) - H(x,y-1).} \tag{2}$$

In the formula $\mathrm{G_x(x,y)}$ represents the gradient in the horizontal direction, and $\mathrm{G_y(x,y)}$, the vertical direction. The gradient magnitude and gradient direction at the pixel are:

$$\mathrm{G(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2}.} \tag{3}$$

$$\mathrm{\alpha(x,y) = tan^{-1}\left(\frac{G_y(x,y)}{G_x(x,y)}\right).} \tag{4}$$

2.2 The Feature Vector of Block

Divide the sample into several blocks with the size of 16×16, and every block contains 2×2 cells of 8×8. Divide $0^{\circ} - 180^{\circ}$ into 9 bins, every bin with the scope of 20°. In accordance with the gradient direction and the gradient magnitude of each pixel in the block, calculate the projection value on the two bins in each cell of the block, which are close to this gradient direction with the method of tri-linear interpolation [6].

The formula 5 describes the tri-linear interpolation and the figure 1 shows the parameters used in the expression:

$$
\begin{aligned}
&h(x_1,y_1,\alpha_1) \leftarrow h(x_1,y_1,\alpha_1) + G(x,y)(1-\tfrac{x-x_1}{d_x})(1-\tfrac{y-y_1}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_1}{d_\alpha})\\
&h(x_1,y_1,\alpha_2) \leftarrow h(x_1,y_1,\alpha_2) + G(x,y)(1-\tfrac{x-x_1}{d_x})(1-\tfrac{y-y_1}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_2}{d_\alpha})\\
&h(x_1,y_2,\alpha_1) \leftarrow h(x_1,y_2,\alpha_1) + G(x,y)(1-\tfrac{x-x_1}{d_x})(1-\tfrac{y-y_2}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_1}{d_\alpha})\\
&h(x_1,y_2,\alpha_2) \leftarrow h(x_1,y_2,\alpha_2) + G(x,y)(1-\tfrac{x-x_1}{d_x})(1-\tfrac{y-y_2}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_2}{d_\alpha})\\
&h(x_2,y_1,\alpha_1) \leftarrow h(x_2,y_1,\alpha_1) + G(x,y)(1-\tfrac{x-x_2}{d_x})(1-\tfrac{y-y_1}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_1}{d_\alpha})\\
&h(x_2,y_1,\alpha_2) \leftarrow h(x_2,y_1,\alpha_2) + G(x,y)(1-\tfrac{x-x_2}{d_x})(1-\tfrac{y-y_1}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_2}{d_\alpha})\\
&h(x_2,y_2,\alpha_1) \leftarrow h(x_2,y_2,\alpha_1) + G(x,y)(1-\tfrac{x-x_2}{d_x})(1-\tfrac{y-y_2}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_1}{d_\alpha})\\
&h(x_2,y_2,\alpha_1) \leftarrow h(x_2,y_2,\alpha_1) + G(x,y)(1-\tfrac{x-x_2}{d_x})(1-\tfrac{y-y_2}{d_y})(1-\tfrac{\alpha(x,y)-\alpha_1}{d_\alpha}) \qquad (5)
\end{aligned}
$$

Tri-linear interpolation refers to calculating the weights in X-direction, Y-direction and angle space with the gradient magnitude. When calculating the final voting weights, firstly, the weights in X and Y direction shall be calculated according to the distance between the pixel point (x,y) (the black point in Figure 2) and the center of the four cells. Secondly, calculate the voting weights by the distance between the gradient direction and the bin centers beside the gradient direction. And then the projective value in the two bins can be gained according to the gradient magnitude and the weights obtained in the step one and step two. The feature of the block can be gained by combining the HOG features of the four cells.

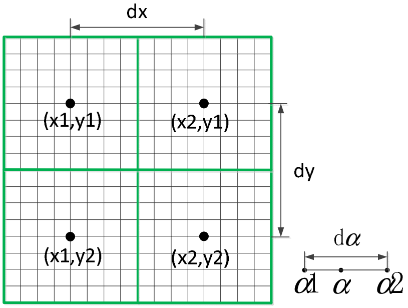


Fig. 1. Formula parameter diagram [7]

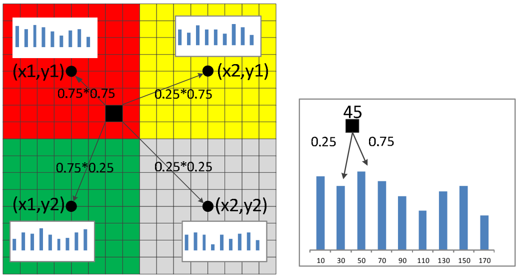


Fig. 2. Tri-linear interpolation diagram

LUT(look-up table) can be used to avoid calculating the weights in the same station of different blocks.

2.3 Normalization of the Feature Vector

The function L2-norm can be adopted to normalize the feature vector of each block, which can eliminate the influence caused by the light and a small bias.

$\mathrm{L2-norm}, \mathrm{v} \leftarrow v/\sqrt{\|v\|^2+\varepsilon^2}$, where ε is a small constant to avoid the denominator is 0, and v is the feature vector before normalization.

2.4 Acquisition of the HOG in Each Sliding Window

Merge all the feature vectors of a sliding window into a 3780-dimensional vector, i.e. the HOG features of the sliding window.

The feature vector of the sliding window obtained can be fed to the classifier for machine training and testing.

3 Rapid Extraction of HOG Features in the Sliding Window

In Dalal's article, he makes use of a sliding window to scan images in different zooming scale. But in the adjacent sliding windows, there are a lot of overlaping blocks. If calculation is made at every new window, it would lead to a lot of double counting. It is a heavy burden to the system. Therefore, look-up table [8] (LUT) is applied to accelerate the calculation. When starting to detect an image, we traversal it by the window at the size of a block, calculate the eigenvectors of the location of each block, and store it in a matrix. When using the sliding window to detect it, we merely need to look up all the blocks contained in the window, take out the corresponding eigenvectors, and combine them to get the HOG features of the detection window.

4 Training and Classification of SVM

Support Vector Machine (SVM) is established on the foundation of the VC dimensional theory of statistical learning theory and structural risk minimization principle which was first presented by Corinna Cortes and Vapnik in 1995. It seeks the best compromise between the complex of the models and the learning ability with the use of limited samples so as to gain the best generalization ability. It has many special advantages in dealing with the small samples, nonlinear and high dimensional pattern recognition and can also be extensively applied to other machine learning such as function fitting.

The key of SVM lies in the kernel function. As the features vector obtained in the pedestrian detection belongs to the nonlinear partition case, thus it needs to be mapped into the upper space by kernel function and turns it into a linearly separable case. There are four types of commonly used kernel function. That is, Linear Kernel Function, Polynomial Kernel Function, Radial Basis Function (RBF for short) and Sigmoid Kernel Function. Here, considering the hardware implementation in the subsequent pedestrian detection, we select Linear Kernel Function and RBF Kernel Function to make detection and contrast.

In Training and testing, we train the classifier by applying the libsvm package [9] provided by Dr. Lin Zhiren, PHD of Taiwan University, on the matlab2008 platform of a PC with Pentium Dual-Core T4300 2.1GHz 2GB. All the training and testing samples are selected from INRIA and MIT databases. Portion of positive and negative samples used in the training can be seen in Figure 3:

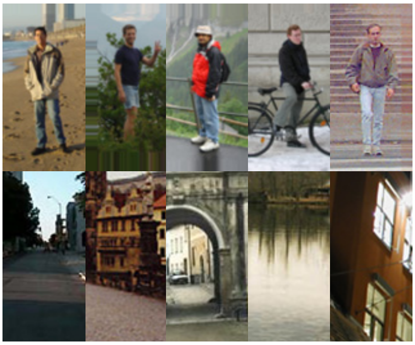


Fig. 3. Positive and negative samples

In test training, the best parameter combination, $\mathrm{C} = 8$ and $\mathrm{g} = 0.0078125$, is selected from a sample set of 200×2 through cross validation by use of the grid.p tool [9] provided by Dr. Lin Zhiren when RBF Kernel Function is applied. If Linear Kernel Function is used, the parameter $\mathrm{C} = 8$ is selected. Select 3,000 positive samples from INRIA and MIT databases as well as 3,000 negative samples from INRIA, then cut them into samples with the size of 128×64. Select 400×2 test samples set from INRIA database to test the detection rates of the two methods. The detection rates of different Kernel Functions are listed in table 1.

Table 1. Detection rates comparison.

Kernel Function type	Detection rates
Linear Kernel Function	98.125%
RBF Kernel Function	98.825%

We can get from Table 1: the testing rate of the RBF Kernel Function is slightly higher than that of the Linear Kernel Function. By observing the predicted labels, we can find that the predicted samples that are error predicted are different from each other.

In detection training, select 3,900 positive samples from INRIA and MIT databases and 3,900 negative samples from INRIA database, then cut them into samples with the size of 128×64. Table 2 shows the number of support vector gained from the training:

Table 2. Support vector comparison.

Kernel Functions type	Support Vector Number
Linear Kernel Function	639
RBF Kernel Function	994

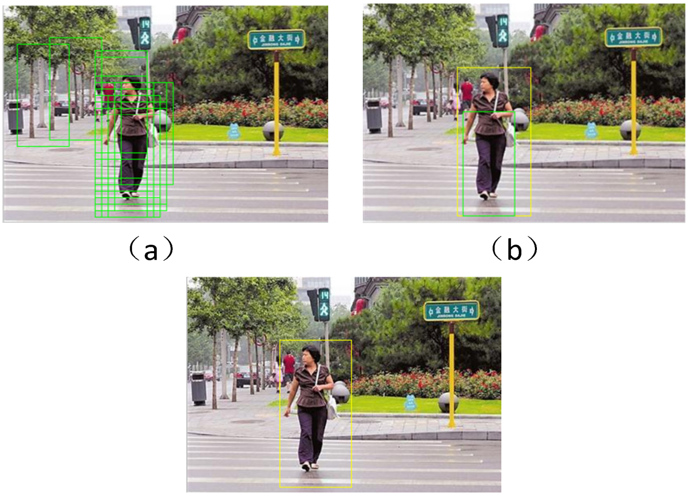
We can get from Table 2:the number of support vector of Linear Kernel Function is far less than that of the RBF Kernel Function. It leads to a higher detection speed by using a Linear Kernel Function than that of RBF Kernel Function.

5 Image Detection and Postprocessing

The original images needs to be scaled to make the pedestrians in the image match the training samples in size because of the different distances from the pedestrians to the camera, the changeable sizes of the pedestrians in the images and the variable images in size. The number and factor of scaling can be chosen according to the size of the image and required detection speed.

Since the detection window scans the images at a certain step in different scales, thus at last the windows that judged containing the same pedestrian by the classifier are not sole, i.e., the pedestrian in the image to be detected is not marked by only one window so these windows contain the same pedestrian need to be merged. At the same time the detection marks the targets of non-pedestrian. We can see from figure 4-(a): the non-pedestrian windows are much less than the pedestrian ones, and merging these windows can exclude a number of non-pedestrian windows [10].

The method of the merger of the windows is as follows: for the same scaled image, first of all, extract one of the detected windows, and then judge whether the subsequent detected window overlaps with this one. If they overlap, calculate the overlap ratio of the two windows. If the overlap ratio is more than 0.6, calculate the average of two windows’ coordinates, then delete these two original coordinates and compare the new window with next window. If the overlap ratio is less than 0.6, the next window belongs to another target, and in case the number of combination is greater than or equal to 4, there must be pedestrians, otherwise not. When complete the merger at this level, these new windows obtained shall be merged with other detected windows at other levels, as shown in Figure 4-(b). Whether merge or not is determined by the overlap area. If the overlap area is 60 percent greater than that of the small window, then delete the small window, with the coordinates of the larger one unchanged; if less than 60 percent, they must be different pedestrians. Figure 4-(c) is the final merging result.

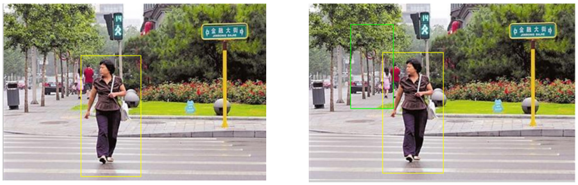


（a） （b）

（c）

Fig. 4. Detecting results diagrams

Figure 5 is the detecting result. The image with the size of 400×270 is detected by the Linear Kernel SVM and the RBF SVM at the scaling factor 1 and 0.7. Figure 5-(a) is the detecting result with the RBF, and Figure 5-(b) is the result detected by the Linear Kernel Function. According to the tests, the detection speed by the Linear Kernel Function is almost 2 times of the detecting speed by the RBF. The Linear Kernel Function has smaller support vector number, less computational complexity and faster detection speed except for the higher false alarm rate than that of the RBF.



（a） （b）

Fig. 5. Detecting results diagrams

6 Conclusion

The adoption of LUT has effectively improved the detecting speed by comparing the detection speeds of two methods in calculating the HOG features. By comparing the detecting speed and effect of Liner Kernel Function and RBF, the detecting speed by Liner Kernel Function is significantly improved except for the false alarm rate. In order to achieve the same detecting effect as RBF, a large number of training samples are necessary. Further research will transplant the training results onto the hardware platform and do depth exploration on the improvement of the detection speed.

Acknowledgments. The authors would like to acknowledge the financial support of Shanghai Economic and Information Technology Committee by the Annual Projects for Absorption and Innovation of Shanghai Imported Technology under Grant No.11XI-15.

References

1. Yanwu, X., Xianbin, C., Hong, Q.: Survey on the latest development of Pedestrian detection system and its key technologies expectation. Acta Automatica Sinica 36, 962–968 (2008)
2. Qian, H., Jiefeng, G., Wenliang, Y., et al.: Pedestrian Detection Based on Histograms of Oriented Gradients. Science Technology and Engineering 9, 3646–3651 (2009)
3. Dalal, N.: Finding people in images and videos. Institute National Polytechnique de Grenoble, France (2006)
4. Cortes, C., Vapnik, V.: Support Vector Networks. Machine Learning (1995)
5. Haiyan, X., Zhitao, X., Fang, Z.: Method of pedestrian detection ahead of vehicle based on linear SVM. Journal of Tianjin Industrial University 31, 72–76 (2012)
6. Rong-yu, Q., Qing, L., Jianming, G., et al.: HOG and Color Based Pedestrian Detection. Journal of Wuhan University of Technology 33, 134–137 (2011)
7. Yanwei, P., Yuan, Y., Xuelong, L., et al.: Efficient HOG human detection. Signal Processing (2011)
8. Dongsheng, L., Pengpeng, L., Gang, W.: A Fast Human Detection Based on Histograms of Oriented Gradients. Electronic Design Engineering 20, 190–192 (2012)
9. Zheng, Z., Yanping, W., Guixiang, X., et al.: Digital Image Processing and Machine Vision. Posts & Telecom Press, Beijing (2010)
10. Shaorui, Y.: Research on Pedestrian Detection of Vehicle Auxiliary System. Xi'an Industrial University, Xi'an (2012)

Video Image Clarity Algorithm Research of USV Visual System under the Sea Fog

Zhongli Ma, Jie Wen, and Xiumei Liang

College of Automation, Harbin Engineering University,
150001 Harbin, China
{mazhongli,wenjie,liangxiumei}@hrbeu.edu.cn

Abstract. The visual system is one of the main equipment of unmanned surface vehicle (USV) autonomous navigation. Under the sea fog, atmospheric particles scattering leads to serious image degradation of the visual system. Because there is obvious sea-sky-line and the larger sky area in the image of offshore, so firstly, the image segmentation is done to get sky area, and through anglicizing sky area characteristics, the sky brightness is estimated, and then a simplified physical model of atmospheric scattering is built up, lastly image scene recovery is finished. Thinking about using this simple image defogging method to video image, foreground and background separation is done. Comparative research with several defogging methods onshore, results show that the proposed method can enhance the video image clarity of the USV visual system under sea fog very well. This research brought a good foundation to further improve the accuracy and precision of surface target identification and tracking algorithm.

Keywords: Sea fog, USV, image segmentation, atmospheric scattering, image clarity.

1 Introduction

Nowadays, the most of studies about image defogging focus on the onshore. Clarity method of degradation image under the fog can divide into two types: one is image contrast enhancement; another one is the weather degraded image restoration method based on physical model [1].

In accordance with the specific needs, image enhancement processing, ignoring the causes of image degradation, can highlight some useful information in the image and weaken or remove some unwanted information. Classical image enhancement processing used to image defogging includes histogram equalization, homomorphic filtering, contrast stretching, wavelet transform and curvelet transform, multi-scale Retinex transform, color-preserving defogging method etc. All of these methods have both advantages and disadvantages. For example, global histogram equalization can make information entropy biggest, but it is not adapted to the depth variation of the local scene; local histogram equalization can change contrast of every small domain of image, but the block effect destroys whole defogging effect[2]; Due to the window

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 436–444, 2013.

function effect, wavelet transform can not process images lower contrast under different scene depth very well[3]; multi-scale Retinex transform has so many advantages such as sharpening, color constancy, large dynamic range and color fidelity, but it can not process uneven fog so well[4]; Xu's method can obtain natural image process effect, but it has serious cast problem when processing uneven fog[5].

Image restoration method mainly studies physical process of image degradation under the fog weather, and establishes degradation model, inverses degradation process and compensates the distortion, so that image without fog and its optimal estimate value can be obtained. Fattal's defogging method is achieved through estimating light path propagation and the reflectance of object surface, but this method can not process image under heavy fog[6]; Tan's defogging method is based on enlarging local contrast of recovery image, but the disadvantage of this method is the over-saturated color of recovery image [7]; He's dark channel prior theory based on statistics from a lot of images without fog is used successfully and defogging effect is so good, but this method is not fitful for image with large block sky domain[8]; Yu's defogging method is to simplify atmospheric model through white balance of atmospheric lighting and to estimate coarser using a fast bilateral filtering, so that this method can effectively restore the contrast and color of scenes, but this method can not process white color object located in an image[9].

Until now, the research about offshore image defogging is not very popular. Normal defogging method is based on image enhancement processing. But Hu's group gave a special method [10], thinking about platting of sea surface, they built up an equal scene depth model, where pixel dot on every equal scene depth line is separately processed using image enhancement method. This method can improve image clarity and reduce noisy. Due to model parameters limitation, this method can not get better process effect to closer distance object, and resolution is not high.

2 Classical Atmospheric Scattering Physical Models

Narasimhan [11] gave monochrome atmospheric scattering model under the condition of fog and haze weather, the image (captured by narrow-band camera) gray value $I(x)$ can be expressed as formula (1):

$$I(x) = A\rho(x)e^{-\beta d(x)} + A\left(1 - e^{-\beta d(x)}\right). \tag{1}$$

Where, x is the spatial coordinates; A is the brightness of the sky; $\rho(x)$ is the scene albedo; $d(x)$ is scene depth; β is the coefficient of atmospheric scattering.

The method based on the physical model is essentially to solve scene albedo using atmospheric scattering model.

As shown in equation(1), the atmospheric scattering model contains two parts. The first part $A\rho(x)e^{-\beta d(x)}$ represents the direct transform model. Due to the scattering effect of the atmospheric particles, in the propagation process of light reflected from an object, a portion of the surface reflection light is lost, and only the non-scattering portion can arrive at the image sensor and form the scene image. With the increasing of the propagation distance, the intensity of reflected light will decrease with exponential law. The second part $A\left(1-e^{-\beta d(x)}\right)$ represents the ambient light model. Because the atmospheric particles will also be scattered by natural light, it will let the atmosphere exhibit characteristics of light source. These characteristics will be passed to the imaging sensor, and affect scene image.

3 Simple Image Defogging Clarity Methods

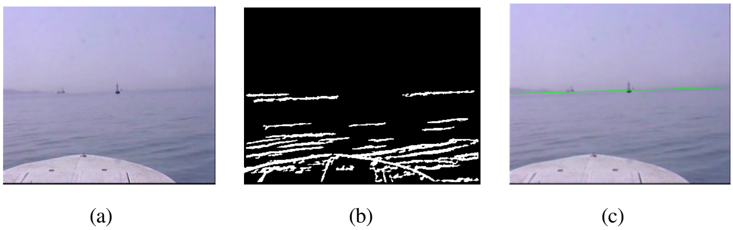
3.1 Sky Brightness Estimation

The Sea-Sky-Line of Image Is Detected. From a long distance, the background of offshore image is generally separated into three areas: sky area, sea area, sea-sky-line area. Closer to the sea-sky-line area, the gray value of image changes bigger. The brightness of the part above the boundary line is higher, while the brightness below the boundary line is lower, so the edge feature is clear.

Mohanty thought that sea-sky-line is a line connected by many points which have larger gray gradient value in image [12]. Under the state of long distance and head up, when a target appears on the surface of sea, it must locate the near area of sea-sky-line. The target search area can be narrowed by detecting sea-sky-line, which can bring the great significance to reduce the computational complexity of the subsequent target detection and identification and inhibit the noise interference from outside the boundary line. Here, on the base of straight line characteristic of the sea-sky-line, the longest curve method [13] is used to detect sea-sky-line. The steps are as follows:

- The image is pretreated using the median filter of 3 × 3 to filter out noise;
- Edge detection is done based on Canny algorithm;
- Image obtained by edge detection is extracted to a straight line segment using the Hough transform line detection;
- The longest straight line is found by calculating the length of all straight line segments.

The sea-sky-line of a USV video frame is extracted shown as figure 1.



(a) (b) (c)

Fig. 1. These show basic steps and results of the sea-sky-line detection of a USV video frame, Fig. 1 (a) is original image, Fig. 1(b) is image after edge detection, Fig. 1 (c) shows the sea-sky-line detected

The Brightness of Sky Is Estimated. As research as above, the area above the sea-sky -line is sky area and the maximum pixel value of area above the sea-sky-line on the original image is selected as the estimation value of sky brightness A.

The Atmospheric Scattering Model is Simplified. In order to simplify atmospheric scattering model, $t(x)=e^{-\beta d(x)}$ is defined, where $0<t(x)<1$, $t(x)$ represents media spread function or transmittance.

To define $J(x)=A\rho(x)$, $J(x)$ represents image gray without fog, so original physical model can be simplified as:

$$I(x)=J(x)t(x)+A(1-t(x)). \tag{2}$$

Atmospheric dissipation function, defined as $U(x)=1-e^{-\beta d(x)}$, indicates an additional image part from the ambient light. So $U(x)=1-t(x)$, $0<U(x)<1$ and formula (2) can be simplified as:

$$I(x)=J(x)t(x)+AU(x). \tag{3}$$

3.2 Atmospheric Dissipation Function Estimation

Rough Estimation Based on the Smallest Color Component. To an image J, define J^c as a color component of J, and $\Omega(x)$ is a square area that x is regarded as center. Assuming that in a local area transmittance is the same, the method of single image fog removal using dark channel prior presents [8]:

$$J^{dark}(x)=\min_{c\in\{r,g,b\}}\left(\min_{y\in\Omega(x)}\left(J^c(y)\right)\right). \tag{4}$$

To take the smallest operation:

$$\min_{y\in\Omega(x)}\left(I^c(y)\right)=\tilde{t}(x)\min_{y\in\Omega(x)}\left(J^c(y)\right)+\left(1-\tilde{t}(x)\right)A^c. \tag{5}$$

Where, A^c is a color component of sky brightness A.

Because the smallest operations of three color components are processed independently, formula (4) is equal to:

$$\min_{y\in\Omega(x)}\left(\frac{I^c(y)}{A^c}\right)=\tilde{t}(x)\min_{y\in\Omega(x)}\left(\frac{J^c(y)}{A^c}\right)+\left(1-\tilde{t}(x)\right). \tag{6}$$

When at least a kind of color component existing on every local area in an image without fog approaching to zero, by taking the smallest operations of three color components, the rough estimation of atmospheric dissipation function can be obtained:

$$\tilde{U}(x)=1-\tilde{t}(x)=\min_{c\in\{r,g,b\}}\left(\min_{y\in\Omega(x)}\left(\frac{I^c(y)}{A^c}\right)\right). \tag{7}$$

In the sky area, $\min\limits_{c\in\{r,g,b\}}\left(\min\limits_{y\in\Omega(x)}\left(\frac{I^c(y)}{A^c}\right)\right)\to 1$ and $\tilde{t}(x)\to 0$. In order to get a more natural image restoration, a constant $\omega(0<\omega\le 1)$ is introduced, and

$$\tilde{t}(x)=1-\omega\min_{c\in\{r,g,b\}}\left(\min_{y\in\Omega(x)}\left(\frac{I^c(y)}{A^c}\right)\right)=1-\omega\tilde{U}(x). \tag{8}$$

Where, the value of ω will be determined according to the fog concentration and the size of the area of the sky, here, the value of ω is equal to 0.75 generally, and the fog is the more concentrated, the value of ω is larger; while when the sky area is the greater, the value of ω is smaller. So, the atmospheric dissipation function is described as:

$$U(x)=1-\tilde{t}(x)=\omega\min_{c\in\{r,g,b\}}\left(\min_{y\in\Omega(x)}\left(\frac{I^c(y)}{A^c}\right)\right)=\omega\tilde{U}(x). \tag{9}$$

Fine Operation Based on Median Filter. Atmospheric dissipation function $U(x)=1-e^{-\beta d(x)}$ is only function of scene depth $d(x)$ and it has no relationship with the scene albedo $\rho(x)$. The rough estimation of the function $U(x)$ has been finished previously and now, a median filtering will be done. Median filtering can keep the edge details, reduce the complexity, improve the speed of operation, and at the same time suppress Halo effect effectively [15]. Here median filtering is respectively used into the three channels of the original image, and processing is shown as followed:

$$M(x)=\underset{y\in\Omega(x),c\in\{r,g,b\}}{med}\left(\frac{I^c(y)}{A^c}\right). \tag{10}$$

Where, the dimension of $\Omega(x)$ is $N\times N$. N depends on the size of the image or video. So:

$$\tilde{U}(x)=\omega\min_{c\in\{r,g,b\}}\left(M(x)\right)\cdot \tag{11}$$

3.3 Scene Recovery

According to formula (3) and formula (9), gray of original image is:

$$J(x)=\left[I(x)-AU(x)\right]/\left(1-U(x)\right)=\left[I(x)-A\omega\tilde{U}(x)\right]/\left(1-\omega\tilde{U}(x)\right). \tag{12}$$

According to formula (8) and sky brightness A, the gray of original image can be obtained through formula (12). The projected rate $t(x)$ can be gotten from formula (8), and then scene depth can be solved. Therefore,

$$\rho(x)=\left[I(x)/A-\omega\tilde{U}(x)\right]/\left(1-\omega\tilde{U}(x)\right). \tag{13}$$

4 Video Image Clarity

4.1 Background Modeling

When video image is defogged, a frame-by-frame defogging method is so time-consuming that the defogging method of the single image cannot satisfy the real-time defogging requirement. Considering the backgrounds of many pictures in the video sequence of USV are almost the same, defogging of the background image is only done several times, and then these defogged backgrounds are applied to other video image frames, this kind of method will greatly reduce the amount of calculation [16].

A frame image of a video sequence with sea fog is shown in Figure 2(a), and Figure 2(b) shows another. The static parts of the scene such as sea surface and sky are known as the background, and the moving target such as a ship is called foreground. Waves in the video image, with no concern about their variation, can be classified as a part of background. Here, the Frame Difference method [17] is used to extract the background of a video.

While, because USV moves continually, so the background of video will change with the time, so the method of real-time extracting background in the reference [18] is adopted: after the images of the current frame and the background are differenced, if the percentage of the change pixel in all pixels is larger than a certain threshold value (typically taken at 80%) in the image obtained, the background will need to be extracted using frame difference method once more. the background image of Figure2(a) using frame difference method is shown in Figure 2(c).

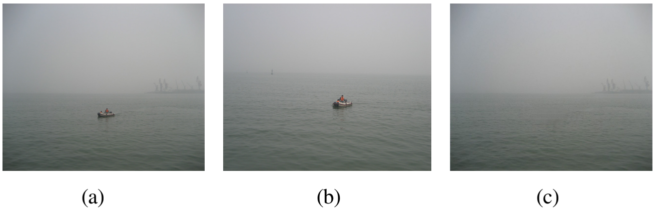


Fig. 2. These show how to separate the background of frame image form a video sequence. Fig. 2(a) is original a frame image, Fig. 2(b) is another frame image, Fig. 2(c) shows the background image.

4.2 Clarity Processing

Due to small changes of the concentration of the fog in the video frames with the same background, so firstly, the fog distribution $F(x)$ of a frame is obtained in all video frames with the same background using original image to subtract clear image:

$$F(x) = I(x) - J(x) \,. \tag{14}$$

And then this fog distribution subtracted by other frames can achieve the effect of video clarity. This can greatly reduce the time of video image processing, so as to improve the real time.

4.3 Video Image Defogging Experiment

In this section, a group of video frames with sea fog from a USV video clip are extracted. The first clip is from the fifth minute of video, shown as fig.3. The size of every frame is 352×288. Simulation is done using Matlab7.11 software, and computer is Intel double core, memory is 2GB.

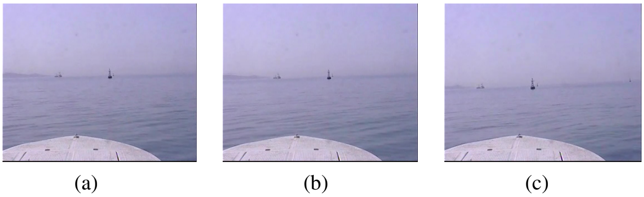


Fig. 3. These show the first group of original video frames, fig.3 (a) is the 256th frame, fig.3 (b) is the 495th frame and fig.4 (c) is the 832th frame in all 2357 frames

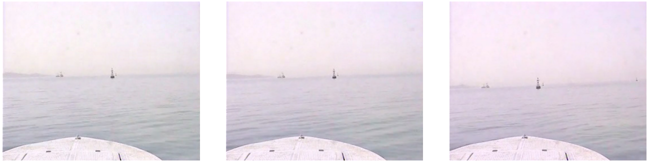


Fig. 4. These show image processing results based on Retinex's method

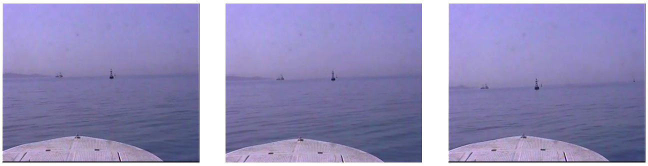


Fig. 5. These show image processing results based on He's method

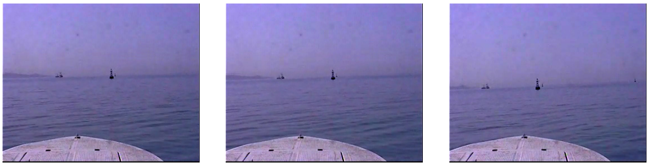


Fig. 6. These show image processing results based on our method

He's algorithm processing time of the video is 81.6724 seconds, Retinex algorithm processing time is 85.3288 seconds, while the proposed algorithm processing time is 72.2133 seconds. Analyzing the video processing effects of the Figures 4 to 6, the proposed algorithm is better, whether details or the color of the video image are restored relatively natural and clear; and the proposed algorithm can improve the video processing speed from the video processing time comparison.

5 Conclusions

In this paper, an improved algorithm based on the monochrome atmospheric scattering model was used to remove sea fog from a video of USV. The difference of proposed method is about sky brightness estimation, because the images of off shore have some important characteristics such as obvious the sea-sky-line and large sky area, so firstly image segmentation is done, and then on the base of analyzing sky area characteristics, estimation value of sky brightness is obtained, and the monochrome atmospheric scattering model is solved. Lastly considering the same background of video frames, the proposed method ran successfully on clarity processing video images under sea fog.

Compared with several defogging methods onshore, the proposed method is proved to be high-efficiency, precise and real-time when it is used to process heavy sea fog video images with obvious the sea -sky-line. And this research laid a good foundation for further research on precision surface target identification and tracking algorithm.

Next research will focus on how to improve the algorithm's processing efficiency to those fog images of off shore without obvious the sea-sky-line, and defogging method to uneven sea fog will be done.

Acknowledgments. The authors are grateful to College of Automation Harbin Engineering University; this project is supported by National Natural Science Foundation Youth Funding of China (No.51109047) and China Scholarship Council(No.2011307358). At the same time, thanks LI Wen for her technical help; at last, thanks our families for their understanding and support.

References

1. Chavez, P.: An improved dark-object substraction technique for atmospheric scattering correction of multispectral data. Remote Sensing of Environment 24, 459–479 (1988)
2. Keun, K.T., Ki, P.J., Soon, K.B.: Contrast Enhancement system using spatially adaptive histogram equalization with temporal filtering. IEEE Transactions on Consumer Electronics 44, 82–86 (1998)
3. Zhu, X., Tao, C.: Application of wavelet coefficient weighted algorithm to remote sensing image processing. Micoroelectronics & Computer 11, 141–149 (2008)
4. Land, E.: The Retinex Theory of color vision. Scientific American 237, 108–128 (1977)
5. Xu, D., Xiao, C., Yu, J.: Color-preserving defog method for foggy or haze scenes. In: Proc of the 4th Int'1 Conf on Computer Vision Theory and Applications (VISAPP), pp. 69–73 (2009)
6. Fattal, R.: Single image dehazing. ACM Transactions on Graphics 27, 1–9 (2008)
7. Tan, R.: Visibility in bad weather from a single image. In: Proc of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8 (2008)
8. He, K.M., Sun, J., Tang, X.O.: Single image haze removal using dark channel prior. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1956–1963. IEEE Press, Miami (2009)
9. Yu, J., Li, D.: Physics-based Fast Single Image Fog Removal. Acta Automatioca Sinica 37, 145–146 (2011)

10. Hu, C., Wang, X.: Research of equal scene depth model of sea fog degraded image clarity processing. Digital Communication 8, 63–65 (2010)
11. Narasimhan, S.G., Nayar, S.K.: Chromatic framework for vision in bad weather. In: Proceedings of IEEE CVPR, vol. 1, pp. 598–605. IEEE Computer Society, South Carolina (2000)
12. Mohanty, N.C.: Image enhancement and recognition of moving ship in cluttered background. IEEE Transactions on PAMI 3, 606–610 (1981)
13. Zhao, F., Yang, K., Cai, T.: Sea and sky boundary line detection based on the longest curve method. Ordnance Industry Automation 28, 82–84 (2009)
14. Wang, X., Wang, S.: Characteristic of ship target IR image. Journal of Applied Optics 5, 837–839 (2012)
15. Yang, Q.X., Tan, K.H., Ahuja, N.: Real-time fast bilateral filtering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 557–564. IEEE, Miami (2009)
16. Xie, B., Guo, F., Cai, Z.-X.: An image defogging algorithm based on the fog veil theory. Computer Engineering and Science 34, 83–87 (2012)
17. Gao, K.: A kind of moving target detection method based on frame difference method and background subtraction. Telecommunications Technology 51, 85–90 (2011)
18. Mendes, A., Bento, L.C., Nunes, U.: Multi-target Detection and Tracking with a Laser Scanner. In: Proc. of 2004 IEEE Intelligent Vehicles Symposium, vol. 796, pp. 14–17. IEEE Press, Italy (2004)

A Study of Vision-Based Lane Recognition Algorithm for Driver Assistance

Feng Ran[1], Zhoulong Jiang[2], Tao Wang[2], and Meihua Xu[2]

[1] Microelectronic Research and Development Center, Shanghai University, Shanghai, China
ranfeng@shu.edu.cn
[2] School of Mechatronics Engineering and Automation, Shanghai University, Shanghai, China
{jzl,alada,mhxu}@shu.edu.cn

Abstract. In this paper, a real-time lane detection algorithm based on vision is presented. This algorithm improves the robustness and real-time of processing by combining with the dynamic region of interest (ROI) and the prior knowledge. When the lanes detected from previous frames have little changes for several frames, we recognize the lane only in dynamic ROI. We also proposed an erosion operator to refine the edge and a Hough transform with a restrict search space to detect lines with a faster rate. Experiments in structured road showed that the proposed lane detection method can work robustly in real-time, and can achieve a speed of 30ms/frame for 720×480 image size.

Keywords: Lane detection, Machine vision, Intelligent vehicles, Driving assistance.

1 Introduction

Recently, extensive researches on ITS (Intelligent Transport Systems), aiming at road traffic efficiency and safety improvement, has been developed in many countries. The most important technology of the ITS is the driver assistance and automatic guidance, the LDWS (Lane Departure Warning System) is an important component. Most of LDWS are based on vision system, processing the images captured by a camera attached in the front of a car. By analyzing the captured image, it can detect the line and trigger an alarm if the vehicle is in danger of departing the road. The LDWS helps prevent the driver from unintended lane departure which is caused by driver's fatigue, drowsiness and improper driving maneuver. In recent, a significant amount of approaches for lane detection had been proposed. Yue Wang [1] proposed a B-Snake based lane detection and tracking algorithm without any camera's parameters. Yifei Wang [2] presented a lane-detection and tracking system based on a novel feature extraction approach and Gaussian Sum Particle filter (GSPF). It is able to improve most of the existing feature maps by removing the irrelevant feature points produced by unwanted objects in the scene. The GOLD [3] (Generic Obstacle and Lane Detection system) developed by Massimo Bertozzi remapped each pixel of the image toward a different position. The resulting image represented a top view of the road region in front of the vehicle, as it was observed from sky. With this image, it could

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 445–452, 2013.

detect the parallel lane boundary. Chris Kreucher [4] proposed a methold which is based on a novel set of frequency domain features that capture relevant information concerning the strength and orientation of spatial edges. The frequency domain features are combined with a deformable template prior, for the sake of detecting the lane makers of interest. Keith A. Redmill [5] adopted a Clothoid curve as the geometric model of the road and used a Kalman filter for the lane detection. The Kalman filter is applied to the estimated parameters to preserve smoothness and to predict lane parameters for the next frame of the image. Joon Woong Lee [6] proposed a lane boundary pixel extractor to obtain the candidate lane boundary pixels and utilized Hough transform to provide the lane-related parameters such as an orientation and a location parameter. Young Uk Yim [7] combined the starting position, orientation and the line intensity as the three features for a lane boundary to improve the rate of recognition. Out of the many possible lane boundary candidates, the best one is then chosen as the one at a minimum distance from previous lane vector.

In this paper, an algorithm based on edge detection and a simplified Hough Transform is proposed to reduce the storage requirement. To make a further reduction of the storage and the improvement of the detection performance, a dynamic ROI (Region of Interest) base on the parameters of the last detected lane markings and vanishing point is suggested. At the same time, to improve the performance of the edge detection, this paper has proposed two 1×2 operators. One is to dilate the edge to remove noise pixels around the edge, the other is to erode the edge to produce clearer edge contours.

2 Road Model

During the lane detection, linear model is used to fit the lane markings. Many researchers have proposed several models such as B-snake spline curve[1], Catmull-Rom spline curve[8] and least median square curve[9], improving the accuracy of mark line fitting. However, due to their increased complexity, the real-time detection is limited. And if the error generated by linear model is within the allowed range, using the linear model would greatly reduce the computation. According to the standard set by Technical Standard of Highway Engineering, 650m is the minimum radius of curve of expressway. Based on this standard, Yu Hou-yun[10] proposed a hypothesis to compute the maximum error ε, which represents the lateral distance between the real lane and the detected lane.

As shown in Figure 1, the solid line which is labeled as “Lane” represents the real lane while the dotted line which is labeled as “Reference Line” represents the detected lane based on the linear model. According to the hypothesis, The R which represents the radius of curve takes the value of 650m and the H which represents the length of the ROI takes the value of 2m. Finally, the maximum error ε can be calculated by

$$\varepsilon = R\text{-}\sqrt{R^2\text{-}\left(\mathrm{H}/2\right)^2} \approx 0.8\mathrm{mm} < 1\mathrm{mm} \tag{1}$$

The result showed that linear model could meet the accuracy requirements.

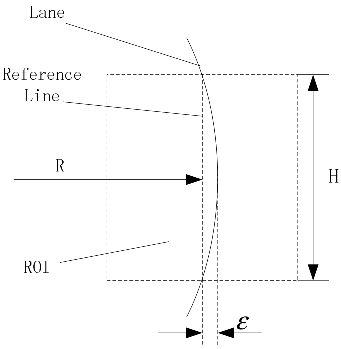


Fig. 1. Sketch of the error

3 Lane Detection

A CMOS camera is fixed on the front-view mirror to capture the road scene. In this paper, it was assumed that the input to the algorithm was a 720×480 RGB color image. Therefore the first thing the algorithm does is to convert the image to a grayscale image in order to minimize the processing time.

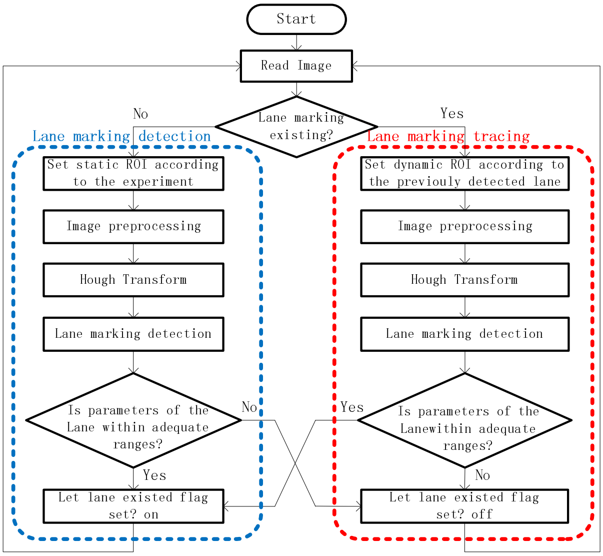
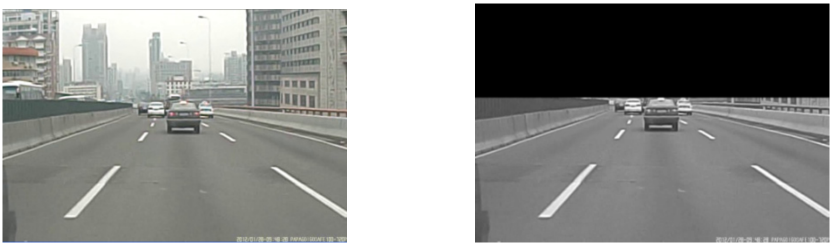

Start
Read Image
Lane marking existing?
No
Yes
Lane marking detection
Lane marking tracing
Set static ROI according to the experiment
Image preprocessing
Hough Transform
Lane marking detection
Is parameters of the Lane within adequate ranges?
No
Yes
Let lane existed flag set? on
Set dynamic ROI according to the previouly detected lane
Image preprocessing
Hough Transform
Lane marking detection
Yes
Is parameters of the Lanewithin adequate ranges?
No
Let lane existed flag set? off

Fig. 2. Alogrithm overview

The lane markings recognition algorithm proposed in this paper is divided into two parts, the lane-detection based on a single frame and the lane-tracking based on successive frames. The algorithm structure is shown in Fig.2. A region of interest (ROI)



(a) Original image

(b) Image after dividing

Fig. 3. Original image and interest region of image

is an area of image on which image analysis and processing are performed. During the setting of static ROI, We divide the picture and take the low part regard as Fig.3.

3.1 Edge Detection

Due to the sharp contrast between the road surface and painted lines, edge detection can be very useful in determine the location of lane boundaries. It also reduces the amount of candidate points required by simplifying the image considerably. Sobel edge detector is employed here to extract the lane boundaries. As shown in the Fig.4, sobel operator is composed of vertical core and horizontal core, and the result can be reference to Fig.6 (a). However, edge contours extracted by sobel operator are not clear enough, that is, there are many isolated weak edge-pixels surround the edge. As a result, it makes the extraction of inner and outer edges of lane become difficult. To solve this problem, we propose a 1×2 operator to refine the edge.

1	2	1
0	0	0
−1	−2	−1

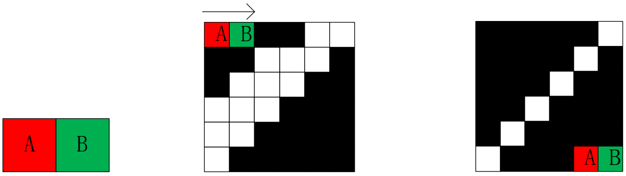
(a) Horizontal core

−1	0	1
−2	0	2
−1	0	1

(b) vertical core

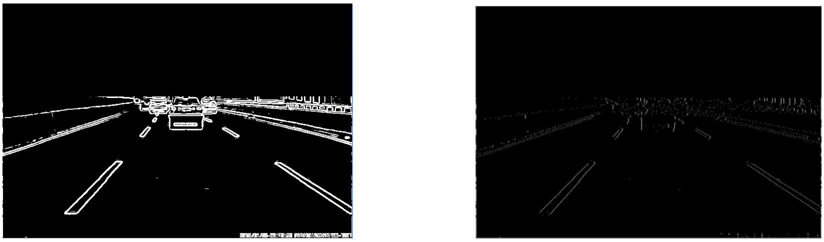
Fig. 4. Horizontal core and vertical core for edge enhancement

As shown in Fig.5 (a), the operator uses a sliding window with the size of 1×2. A and B are the name of the two pixels in the sliding window. Each image is densely scanned from the top left to the bottom right with the sliding window. If the pixel gray value of B equals 255, then set the pixel gray value of A to 0, that is ,if the original gray value of A is 255, it will be replaced by 0. On the contrary, if the gray value of B is 0, then the gray value remains unchanged, as illustrated in Fig.5. Inflation of edge is the inversion of erosion. Edge detection results are shown in Fig.6.



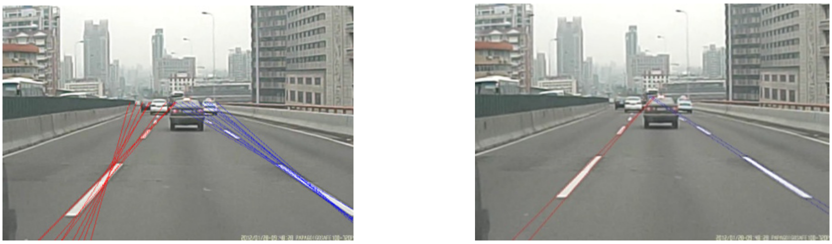
(a) Erosion operator (b) Original edge (c) Edge after erosion operator

Fig. 5. Erosion operator and its result



(a) Sobel edges (b) Erosion after sobel edges

Fig. 6. Example of edge detection result



(a) Hough transform after sobel edges (b) Hough transform after Erosion

Fig. 7. Hough transform after edge detection result

Figure7 shows the hough transform results after edge detection. For the sobel edges are not clear enough, there will be several lines detected in the same lane, as shown in Fig.7(a).It will consume extra time to decide which pairs to be the inner edge and outer edge of the lane. We use Erosion operator to refine the edges. It ensures that all the pixels are isolated. As a result, only inner edge and outer edge remained in a lane. Result can be find in Fig.7(b).

3.2 Hough Transform

The lane detector used is a standard Hough transform with a restrict search space.

Hough transform is used to get slopes and intercepts of the candidates. If there is a straight line y=kx+b, its parameter space polar coordinate lines can be expressed as the following equation:

$$\rho = x\cos\theta + y\sin\theta \tag{2}$$

Where x and y are the coordinates value of a pixel in a image,ρ is the distance from the origin to the fitted line, θis the angle between the normal line and x-axis. As illustrated in Fig.8, the parametersρandθare shown by ρ_L, ρ_R, θ_L andθ_R. Equation (2) is applied to all pixels of left boundary and right boundary. As a result, each point(x,y) in the image are mapped into a sinusoid in parameter space and we get two accumulate arrays $H_L(\rho_L,\theta_L)$ and $H_R(\rho_R,\theta_R)$. All elements in accumulators are initialized to zero and we set the increments of ρand θ to 1 and 1^o, respectively. Considering the likehood of locations of the road, we reduce the search rang of θ and to increase the search efficiency. Yu Tianhong [11] proposed that the angle between the normal line and x-axis usually falls in the range of $[15^o,75^o]$, so in this paper, we set the range of θ_L andθ_R to $[15^o,75^o]$ and $[105^o,165^o]$, respectively. The experiment results suggested that it did maintain the recognition rate while reducing the search time. At last, two parametersθand ρshould be estimated to fit the straight line model and their values are determined by maxima in each accumulator array.

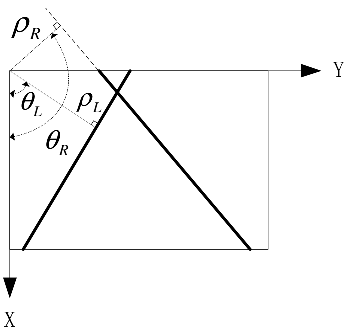


Fig. 8. Image coordinates

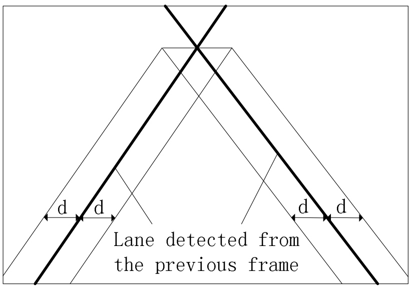
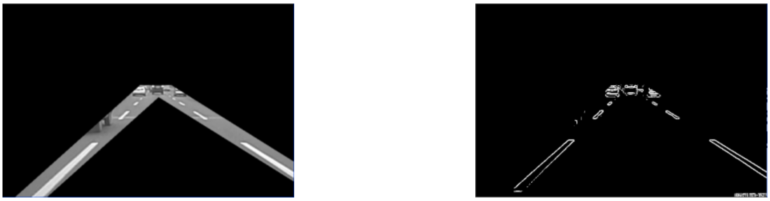


Fig. 9. Set up of dynamic ROI

3.3 Dynamic ROI

If the parameters θand ρof lanes detected from previous identification process have changed little for several times, it means that the vehicle has turned into the tracing phase and dynamic region ROI has been established. In this paper, the tracing algorithm is also the lane detection itself, but it searches the candidates in the established dynamic ROI. According to the identification result from the previous frame, we can get the slopes, intercepts and vanishing point. Vanishing point is defined as the intersection of the left and right lanes. Since the lane mark's succession, we can enlarge the lane boundary identified from the previous frame to set up the ROI. As shown in Fig.8, d is the distance between the boundary of ROI and the center of the lane detected from the previous frame. The Effect of dynamic ROI can refer to Fig.9 (a) and the result of edge



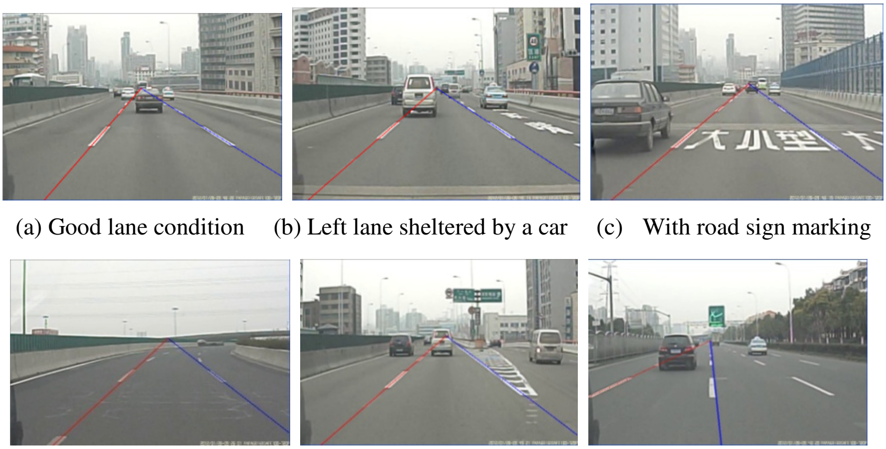
(a) Effect of dynamic ROI (b) Edge detection in dynamic ROI

Fig. 10. Effect of dynamic ROI and edge detection in the ROI

detection in dynamic ROI is shown in Fig.9 (b). Experiments showed that searching the edge points within the ROI will not only improve the speed of execution, but also eliminate abundance of useless disturbed message.

4 Experiment Results

In this paper, our proposed algorithm is developed and tested on the PC-based experimental platform with an Intel Pentium IV 2.2GHz CPU inside, using the C programming language. Due to the pure C programming language, it can be easily transplanted to embedded systems. In order to generalize the lane detection, we get the image in highways and normal roads, dashed marking, straight and curved roads. When shooting, the camera is fixed on the front-view mirror. The size of captured images is 720×480 pixels and the frame frequency is 30fps.The experiment results showed that the average image processing time was about 30ms/frame, which could meet real-time and stability request of the vision navigation system. Fig.11 shows the robust results of our algorithm on various road conditions.



(a) Good lane condition (b) Left lane sheltered by a car (c) With road sign marking

(d) Curve lane (e) Complex sign on right lane (f) Lane changes

Fig. 11. Result of our algorithm under various road conditions

5 Conclusion

In this paper, we have proposed a robust lane detection algorithm on marked roads which can be applied to Lane Departure Warning Systems or other navigation systems. This paper adopted a 1×2 erosion operator to refine the edge. By using a Hough transform with the restrict search space, the algorithm can well detect the lanes in the image with a faster rate. We recognize the lane only in dynamic ROI by using prior knowledge, and obtain a faster recognition rate, higher accuracy and better stability. However, in the real-time lane detection, there are many problems to be solved. For example, according to this algorithm, we must assure a relatively pure road surface must be provided at the very beginning to generate the initialization parameters, otherwise it would cause false identifications. And in order to improve its mobility, portability and convenience, we need to investigate the hardware system to compute the proposed algorithm and improve its real timing and precision.

Acknowledgments. The authors would like to acknowledge the financial support of Shanghai Economic and Information Technology Committee by the Annual Projects for Absorption and Innovation of Shanghai Imported Technology under Grant No.11XI-15.

References

1. Wang, Y., Teoha, E.K., Shen, D.: Lane detection and tracking using B-Snake. Image and Vision Computing 22(4), 269–280 (2004)
2. Wang, Y., Dahnoun, N., Achim, A.: A novel system for robust lane detection and tracking. Signal Processing 92(2), 319–334 (2012)
3. Bertozzi, M., Broggi, A.: GOLD: Aparallel real-timestereo vision system for generic obstacle and lane detection. IEEE Transactions on Image Processing 7(1), 62–81 (1998)
4. Kreucher, C., Lakshmanan, S.: Lane: A lane extraction algorithm that uses frequency domain features. IEEE Transactions on Robotics and Automation 15(2), 343–350 (1999)
5. Redmill, K.A., Upadhya, S., Krishnamurthy, A., Özgüner, Ü.: A lane tracking system for intelligent vehicle applications. In: IEEE Intelligent Transportation Systems, Oakland, pp. 273–279 (2001)
6. Lee, J.W., Yi, U.K.: A lane-departure identification based on LBPE, Hough transform, and linear regression. Computer Vision and Image Understanding 99(3), 359–383 (2005)
7. Yim, Y.U., Oh, S.-Y.: Three-feature based automatic lane detection algorithm (TFALDA) for autonomous driving. IEEE Transactions on Intelligent Transportation Systems 4(4), 219–225 (2003)
8. Wang, Y., Shen, D., Teoh, E.X.: Lane Detection Using Catmull-Rom Spline. In: IEEE International Conference on Intelligent Vehicles, pp. 51–57 (1998)
9. Bi, Y., Guan, X., Zhan, J.: A method of mark line searching in detection process. Automotive Engineering 28(5), 439–442 (2006)
10. Yu, H.-Y., Zhang, W.-G.: Lane tracking and departure detection based on linear mode. Processing Automation Instrumentation 30(11), 1–3 (2009) (in Chinese)
11. Yu, T.-H.: Study on Vision based Lane Departure Warning System. Univ. of Jilin, Jilin (2006) (in Chinese)

New Approach to Image Retrieval Based on Color Histogram

Muhammad Imran[1], Rathiah Hashim[1,⋆], and Noor Eliza Abd Khalid[2]

[1] FSKTM, University Tun Hussein onn Malaysia
[2] FSKTM, University Teknologi MARA Malaysia
malikimran110@gmail.com, radhiah@uthm.edu.my, elaiza@tmsk.uitm.edu.my

Abstract. Nowadays a lot of information in the form of digital content is easily accessible but finding the relevant image is a big problem. This is where the Content Based Image Retrieval (CBIR) comes in to solve the image retrieval dilemma. But a CBIR system faces certain problems such as a strong signature development. Also, one of the major challenges of CBIR is to bridge the gap between the low level features and high level semantics. Previously, several researchers have proposed to improve the performance of a CBIR system but they have only answered image retrieval problem to an extent. In this paper, we propose a new CBIR signature that uses color color histogram. The results of the proposed method are compared previous method from the literature. The results of the proposed system demonstrates high accuracy rate than the previous systems in the simulations. The proposed system has significant performance.

1 Introduction

To retrieve the similar images based on the query from an image database is a challenging task. Visual image content retrieval techniques are in focus for more than a decade. The two main approaches are available for the image retrieval. First is the text based image retrieval and the other based on image contents. In the text based image retrieval traditional database techniques are used to manage images. To searching in large image database the text based retrieval is not easy as the text based search is highly dependent on who adds it and also the same image can have different meanings for different peoples. the text based image retrieval aslo called as keyword based image retrieval (KBIR). The KBIR is a time consuming task as it retrieve the images manually. To search the image based on the content is called the content based image retrieval which is the second approach [1]. In this era of information technology the researchers are focusing on the content based image retrieval (CBIR). CBIR used the different features of the image to perform the search. The CBIR is suitable for the large and small size of databases. by analyzing the image content the CBIR aims at developing new robust techniques for retrieval of the matching images. the

[⋆] *IEEE member*

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 453–462, 2013.

exponentially increasing digital content databases motivated the researchers to proposed the best CBIR techniques.

Through CBIR the user can search the images using semantic contents such as object categories [2]. The object categories can be forest,vehicle, human and buildings. CBIR adopted the feature extraction and selection approaches. Mainly three types of low level features (Texture, Color and shape) are extracted from the image. The retrieval process generally perform in three steps. In step first the feature are extracted, and these features actually describe the content of the image. In the second steps the extracted features of the image database and the feature vector of the query image are used to find the distance. In short we can say that in the second step the similarity measure is performed between the query image and the database images. In step 3 the results are indexed and display to the user. The feature extraction step is more important because the only way to achieve the better result are the good signature. Therefore, development of effective and efficient feature extraction is the critical part of CBIR systems. The extracted features should represent the image in a way that complies with the human perception subjectivity. One of the major challenges of the CBIR is to bridge the gap between the low level features and high level semantics. Different users analyze the same images differently, even same user may analyze the same images differently in different circumstances. That why it is a difficult job for CBIR to understand that what user actually want. To bridge this semantic gap researchers proposed a variety of solutions. There are different CBIR techniques proposed in the literature. Few of them use local features while others use global features. The researchers also segment the image into regions based on color and texture. Many machine learning techniques are also applied in the CBIR system. The concept map of CBIR system can be seen in Figure 5 after references.

In coming section we will study about the previous work of CBIR,the proposed approach will be described in section 3, section 4 is reserve for the result and analysis, the end of this paper will be with conclusion in section 5.

2 Related Works

Rao et al.[3] used texture features for the image indexing and retrieval. they used the wavelet transformation to construct a feature vector. Euclidean distance was used to measure the image similarity. Authors used the Haar wavelet trans for signature development. To perform clustering they modified and implemented the ROCK clustering algorithm. The Wavelet signature (texture feature representation) is computed from sub image as follows,

$$f_r = \sqrt{\frac{\sum c_{ij}^2}{i \times j}} \tag{1}$$

Where f_r is the computed Wavelet signature (texture feature representation) of the sub image, C_{ij} is the representation of the intensity value of all elements of sub image and i j is the size of the sub image. the proposed technique was applied on soem garments images and get better results. Singhai et al. [4] performed the survey on the CBIR system. They discussed few CBIR techniques and introduced some new feature set to enhance the CBIR system. Fuzzy techniques sued by the author, achieved good performance than others. Abubacker [5] used color, texture and shape features for the image. For the color feature they used the spatial based color moments. First they divide the image into 25 blocks then calculate the Red Green Blue (RGB) values of each block. RGB values are converted to Hue, Saturation Intensity (HSV). According to the author the three color moments; mean, variance and skewness are effective and efficient for the color distribution of images. The formula for mean, variance and skewness are given below:

$$Mean(\mu_i) = \frac{1}{N}\sum_{j=1}^{N} f_{ij} \tag{2}$$

$$Varience(\sigma_i) = (\frac{1}{N}\sum_{j=1}^{N}(f_{ij} - \mu_i)^2)^{\frac{1}{2}} \tag{3}$$

$$Skewness(\S_i) = (\frac{1}{N}\sum_{j=1}^{N}(f_{ij} - \mu_i)^3)^{\frac{1}{3}} \tag{4}$$

where f_i is the value of the i^{th} color component of the image block j, and N is the number of blocks in the image. For the texture feature author used the Gabor filter. And applied the 2D Gabor function to obtain the set of Gabor filters with different scale and orientation. By using Gabor filter, he performs convolution on the image to obtain the Gabor transform. Invariant shape features are used to extract the shape features. Following are the steps taken by the author to extract the shape feature.

1. Based on the threshold value, the image is converted to binary image.
2. Using canny algorithm, the edges of the binary image are detected.
3. The centroid of the object is obtained by arranging the pixels in clockwise order and forming a closed polygon.
4. The centroid distance and complex coordinate function of the edges are found.
5. The farthest points are found and Fourier transform is applied on them.

Akgul et al [6] completed a survey of CBIR in the medical imaging. Authors discussed the current state of the art techniques of CBIR in medical imaging. They came up with the new challenges and opportunities of CBIR in medical diagnose process. Authors tried to focus the attention of the researcher on operation issues in medical CBIR and proposed certain strategies to tackle them. Huang et al [7] proposed the new technique of the CBIR using color moment and Gabor texture feature. To obtain the color moment, they convert the RGB image to HSV image. Then, by getting the equalized histogram of the three HSV components calculated the three moments for each color space. Modified form of the Euclidean distance was used to measure the similarity between the query image and the database image. The equation is given below;

$$D(q,s) = \frac{1}{L}\sum_{i=0}^{L-1}(1 - \frac{|q_i - s_i|}{max(q_i, s_i)}) \tag{5}$$

The global distance is computed as the weighted sum of similarities as

$$D(q,s) = \frac{\omega_c.D_c(q,s) + \omega_t.D_t}{\omega_i + \omega_t} \tag{6}$$

Through experiments Huang et al. [7] showed that the overall result of the proposed technique was better than other techniques. Relevance feedback combined with SVM by Zhao et al [8]. They also combine the different features. They extract three different kinds of texture feature. The three texture features were combined with the color feature in different combinations. Three different combinations were contracted having one color moment and two texture features. The effect of region based filtering tested by the [9]. Pujari used the wavelet based texture features. For the color images the texture features of each color space RGB are extracted separately. For similarity measure the integrated region matching used. The experiments performed using 0%, 3%, 6% and 8% filtering. There is no measurable difference between the 6% and 8%. Oliveira et al [10] used breast density for the image retrieval to help the radiologists in their diagnosis. Particle swarm optimization (PSO) with relevance feedback was used by the Broilo [11] to enhance the performance of CBIR. Author formulates the image retrieval process as an optimization problem and applied PSO on CBIR.

3 Proposed Approach

Color feature is one of the basic features used to retrieve the images. It is most intuitive and obvious feature of the image and color is easily extracted. To describe color information different methods are available. One of methods is color histogram. Color histogram has the advantages of speediness and not sensitive to images changes, such as translation, rotation,etc.

In this paper, we used center moment to describe histogram, the center moment and mean definition [12] is

$$\mu_n = \sum_{i=0}^{L-1}(Z_i - m)^n p(Z_i) \tag{7}$$

$$m = \sum_{i=0}^{L-1} Z_i p(Z_i) \tag{8}$$

where n is moment order Z_i is gray level, $p(Z_i)$ is normalized histogram, m is histogram mean and L is the gray level number. We use the HSV color model. Fallowing are the process of feature extraction

1. Divide the image space into 4x4 sub images which result into 16 sub images as shown in Fig 1.
2. convert the each sub image to the HSV image.
3. HSV sub image divided into three components (i.e. H,S,V).
4. Get the histogram equalized of the three components sub image. Histogram equalization is a method in image processing of contrast adjustment using the images histogram. The method can lead to better views of structure in images, conducive to subsequent treatment.
5. Calculate three moments (mean , skewness and variance)for the three components histogram respectively,
6. Combine the three components moment.
7. Combine the 16 feature vector in a single feature vector.

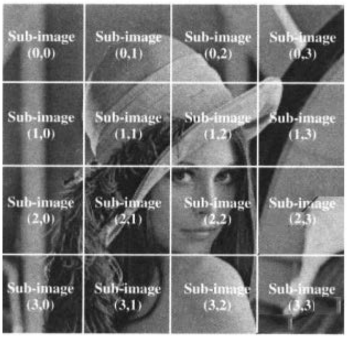


Fig. 1. Definition of sub image

4 Results and Analysis

We compared the result of the proposed system with previous proposed CBIR system namely Variance Segment Method [13] and Histogram based taken from [14]. We used the famous coral data set for our experiments. The database contains 10 classes and each class has 100 images. We have implemented the proposed system under Matlab R2010b.

4.1 Metrics Used for the Evolution

To measure the performance of the CBIR system different metrics are available. Precision is one of the metric which has been used in the several previous works such as Hiremath et al. [15], Banerjee et al. [14] and Wang et al [16] Precision can be calculated as;

$$Precision = \frac{NumberofTruePositive}{NumberofTruePositive + FalsePositive} \tag{9}$$

4.2 Performance in Terms of Precision

As illustrated above, precision is one of the metric used to check the performance of the CBIR system. Table-1 shows the performance of the proposed algorithm with different P @ n evaluation. The precision is calculated using the top most 40, 30 and 10 ranked results. In our experiments, we select 10 random images from each category and perform retrieval process, we repeat this process for 10 times and took the category wise average precision.

Table 1. Performance at different n

Class	**n=40**	**n=30**	**n=10**
Africa	.83	.88	.92
Beach	.41	.50	.62
Buildings	.45	.49	.57
Buses	.24	.26	.34
Dinosaurs	.96	.97	1
Elephant	.37	.42	.6
Flower	.28	.32	.49
Horses	.47	.55	.64
Mountains	.17	.19	.33
Food	.17	.18	.26
Avg	**.435**	**.476**	**.577**

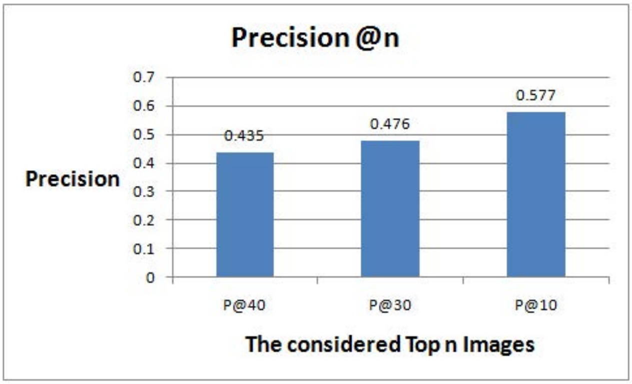


Fig. 2. Definition of sub image and image blocks

From Table 1 and the graph shown in Figure 2, it is clear that precision is affected, if we change the number of top most n images. As we can see that when the top retrieval kept 10 the precision was 0.577, but when the top most retrieval set as 40 the precision down to 0.435.

4.3 Comparison with Previous Methods

Comparison of the proposed method with Simple Histand Variance Segment Method in tabular form.

From Table 2, it is clear that proposed method has better results than Hist based and variance segment method.

In Figure 2, the proposed method is compared with Simple Hist and Variance Segment. the result of proposed method is taken for n = 40, 30, and 10. From the above graph, it is clear that the proposed method has better precision than both previous methods.

Table 2. Comparison of proposed method with previous methods

Class	**Simple Hist**	**Variance Segment**	**Proposed Method**
Africa	.30	.13	.88
Beach	.30	.26	.50
Buildings	.25	.11	.49
Buses	.26	.17	.26
Dinosaurs	.90	96	.97
Elephant	.36	.34	.42
Flower	.40	.49	.32
Horses	.38	.20	.55
Mountains	.25	.25	.19
Food	.20	.15	.18
Avg	**.435**	**.476**	**.577**

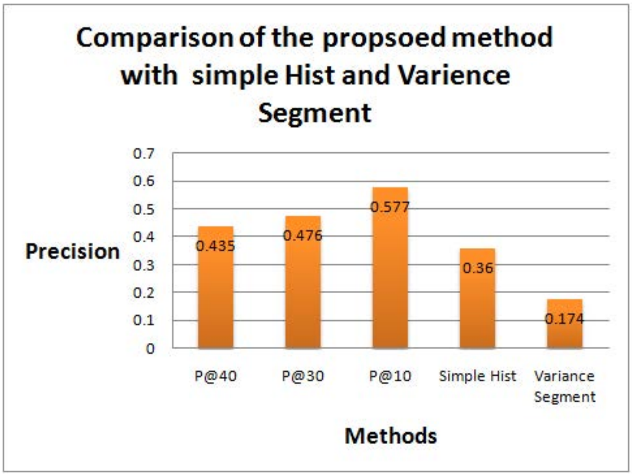


Fig. 3. Comparison of the proposed method with simple Hist and Variance Segment

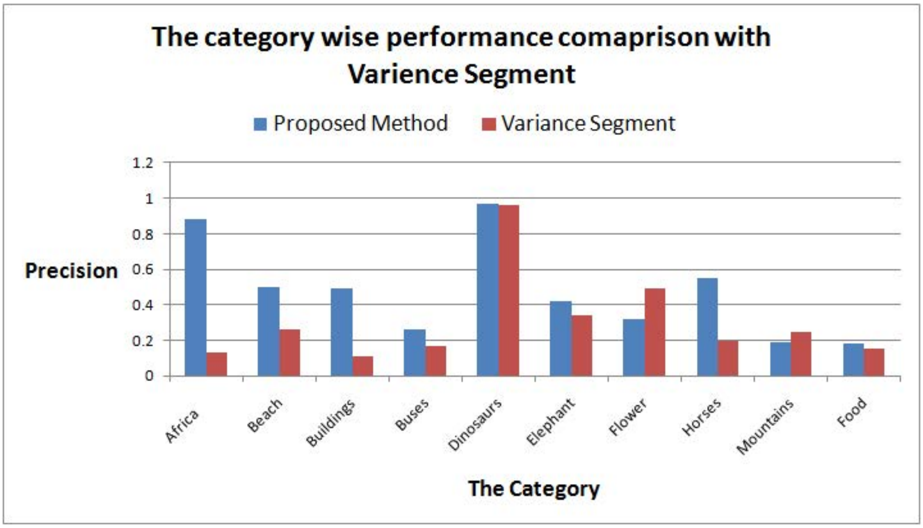


Fig. 4. Category-wise Comparison of Proposed Method (n @ 30) with previous methods

In the above graph, we can see that the proposed method has better result than Simple Hist method on all 10 categories. For the class dinosaurs the proposed method has 97% retrieval rate as the precision reaches 0.97.

5 Conclusion

A new CBIR system has been implemented to answer the shortcomings in the previous CBIR methods. Nowadays a lot of information in the form of digital content is easily accessible on the internet but finding the relevant image is a big problem using current CBIR systems. The proposed system used color features. The color features are extracted by using color histogram. The performance of the system has been evaluated with the previous proposed CBIR techniques. The system is also tested on the different top n image retrieval. It has been observed that the proposed system outperforms, if the *n* is equal to 30 or less than 30. From the results, it can be clearly seen that the performance of the proposed CBIR system has better then other competitors. As a future work to increase the performance of the proposed system texture feature can be combine with the color features.

Acknowledgment. The authors would like to thank University Tun Hussein Onn Malaysia for supporting this research under the Postgraduate Incentive Research Grant Vote No 1031.

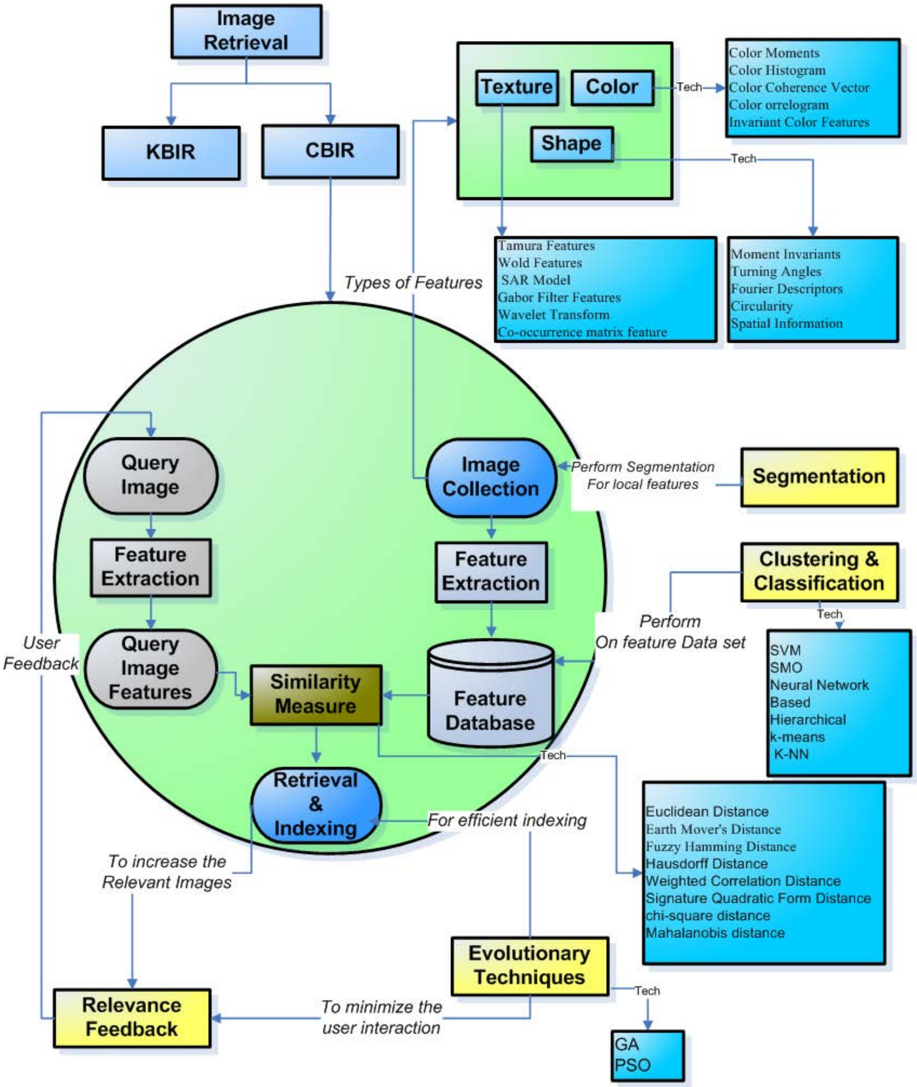


Fig. 5. Concept map of CBIR

References

1. Deshmukh, A., Phadke, G.: An improved content based image retrieval. In: 2nd International Conference on Computer and Communication Technology (ICCCT), pp. 191–195 (September 2011)
2. Sheikh, A., Lye, M., Mansor, S., Fauzi, M., Anuar, F.: A content based image retrieval system for marine life images. In: IEEE 15th International Symposium on Consumer Electronics (ISCE), pp. 29–33 (June 2011)

3. Gnaneswara Rao, N., Vijaya Kumar, V., Venkata Krishna, V.: Texture based image indexing and retrieval. IJCSNS International Journal of Computer Science and Network Security 9(5), 206–210 (2009)
4. Singhai, N., Shandilya, S.K.: A survey on: Content based image retrieval systems. International Journal of Computer Applications 4(2), 22–26 (2010)
5. Abubacker, K., Indumathi, L.: Attribute associated image retrieval and similarity re ranking. In: International Conference on Communication and Computational Intelligence (INCOCCI), pp. 235–240 (December 2010)
6. Akgul, C., Rubin, D., Napel, S., Beaulieu, C., Greenspan, H., Acar, B.: Content-based image retrieval in radiology: Current status and future directions. Journal of Digital Imaging 24, 208–222 (2011)
7. Huang, Z.C., Chan, P., Ng, W., Yeung, D.: Content-based image retrieval using color moment and gabor texture feature. In: International Conference on Machine Learning and Cybernetics (ICMLC), vol. 2, pp. 719–724 (July 2010)
8. Zhao, L., Tang, J.: Content-based image retrieval using optimal feature combination and relevance feedback. In: International Conference on Computer Application and System Modeling (ICCASM), vol. 4, pp. V4–436 –V4–442 (October 2010)
9. Pujari, J., Nayak, A.: Effect of region filtering on the performance of segmentation based cbir system. In: International Conference on Signal and Image Processing (ICSIP), pp. 292–295 (December 2010)
10. de Oliveira, J.E.E., Machado, A.M.C., Chavez, G.C., Lopes, A.P.B., Deserno, T.M., de Araújo, A.A.: Mammosys: A content-based image retrieval system using breast density patterns. Computer Methods and Programs in Biomedicine 99(3), 289–297 (2010)
11. Broilo, M., De Natale, F.: A stochastic approach to image retrieval using relevance feedback and particle swarm optimization. IEEE Transactions on Multimedia 12(4), 267–277 (2010)
12. Rafael, C., Gonzalez, R.E.W., Eddins, S.L.: Digital image processing using matlab. Publishing House of Electronics Industry (2009)
13. Bhuravarjula, H., Kumar, V.: A novel content based image retrieval using variance color moment. International Journal of Computer and Electronic Research 1(3), 93–99 (2012)
14. Banerjee, M., Kundu, M.K., Maji, P.: Content-based image retrieval using visually significant point features. Fuzzy Sets and Systems 160(23), 3323–3341 (2009)
15. Hiremath, P., Pujari, J.: Content based image retrieval using color boosted salient points and shape features of an image. International Journal of Image Processing 2(1), 10–17 (2008)
16. Wang, J., Li, J., Wiederhold, G.: Simplicity: semantics-sensitive integrated matching for picture libraries. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(9), 947–963 (2001)

Comparison and Evaluation of Human Locomotion Traits with Different Prosthetic Feet Using Graphical Methods from Control Area

Lulu Gong[1,⋆], Qirong Tang[2], and Hongwei Mo[3]

[1] School of Life Sciences and Technology, Tongji University, Siping Rd. 1239,
Shanghai 200092, China
[2] Institute of Engineering and Computational Mechanics, University of Stuttgart,
Pfaffenwaldring 9, 70569 Stuttgart, Germany
[3] Automation College, Harbin Engineering University, Harbin 150001, China
lulugong@tongji.edu.cn, qirong.tang@itm.uni-stuttgart.de, mhonwei@163.com

Abstract. This study investigates joint kinematics, joint angular positions, and orbital dynamic stability of human walking with different prosthetic feet by using graphical methods of phase plane portraits, Poincaré maps and Floquet multipliers, respectively. The Flex foot, SACH foot, Seattle foot and one non-specific optimized foot are taken as the research objects. Numerical experiments are performed to compare and evaluate human locomotion traits on several aspects by focusing on the concerned four kinds of prosthetic feet.

Keywords: human locomotion traits, Poincaré map, orbital stability, nonlinear dynamics.

1 Introduction

Prostheses play an important role in helping amputees whose lower limb or upper extremity has been amputated. The prosthetic foot is exactly one of the important components of lower limb prostheses. Some prosthetic feet are designed to improve gait quality. For example, the standard Solid Ankle Cushion Heel (SACH) foot is capable of storing energy, while the Seattle foot and Flex foot contain dynamic elastic responses. Lehmann et al. in [1] make comprehensive analysis of these three prosthetic feet concerning the forefoot compliance, the angle range of feet ankles, the shank flexibility, as well as dorsiflexion. Findings from Klute et al. indicate that SACH foot is stiffest in dorsiflexion, followed by Seattle foot, and then Flex foot in sequence [2]. Ackermann in [3] not only analyzes these three prosthetic feet, but also investigates and provides one optimized foot with optimal parameters for ankle joint stiffness and damping. He points out that plantar flexion moments at midstance, final stance and pre-swing lead to a greater dorsiflexion for Flex foot, compare to SACH foot and Seattle foot. Finally, he concludes that the optimized foot shows a significant improvement compared with other mentioned prosthetic feet and is approximate to the performance of normal human gaits.

⋆ Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 463–471, 2013.

The dynamic stability is a critical aspect for human walking with prosthetic foot, which is defined as the sensitivity of a dynamic system responding to perturbations [4,5]. In order to understand how dynamic stability is maintained during human locomotion, two approaches are frequently applied, namely, the local dynamic stability and orbital dynamic stability. Local stability quantifies the system's states respond to very small perturbations continuously in real time, while orbital stability quantifies the response of the system's states to local perturbations discretely [6]. Investigations have shown that a foot dynamic system can be orbitally stable despite of locally slight instability [6,7,8].

This study aims to compare and evaluate the performances of four different prosthetic feet by using graphical methods from control area. By this way, it is looking forward to providing some reference information for the design, selection and manufacturing of prosthetic feet for amputees.

2 Basic Concepts and Terminology of Human Locomotion

Walking is a motion mainly happens in the sagittal plane, see Fig. 1. Thus, 2-D models can deliver many insights into human locomotion gaits. The seven-body planar model shown in Fig. 2 is adopted to study normal and pathological locomotion gaits. The model is composed of 7 bodies, i.e., the HAT, the two thighs, the two shanks and the two feet. The human pelvis, upper part of the trunk, arms and head are modeled by a single rigid body referred as HAT. This model has 7 degrees of freedom without the consideration of the HAT horizontal and vertical motions. Thus, the motion of the model can be described by 7 generalized coordinates which are seven angles depicted in Fig. 2.

The foot motion during ground-level walking is usually divided into two main phases: the stance phase and the swing phase, see Fig. 3, for the right foot as an

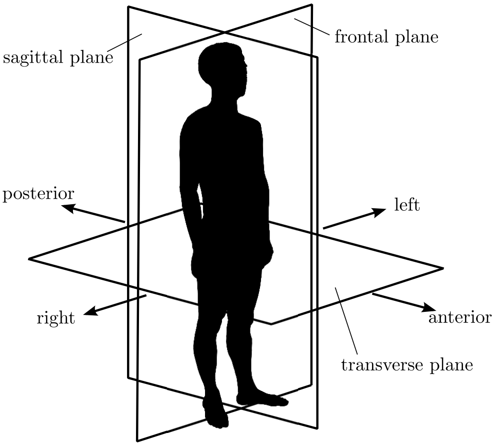


Fig. 1. Anatomical position with three reference planes

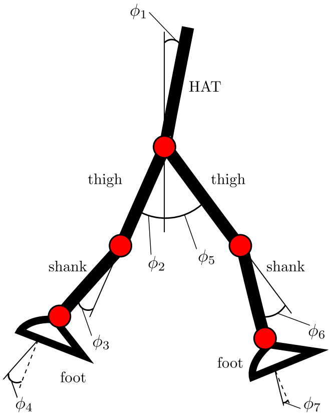


Fig. 2. The seven-body plane model

example. The ideal gait cycle is typically defined as starting with the heel strike (HS) of one foot and ending at the next heel strike of the same foot. The stance phase begins when the heel strikes the floor and ends at toe off (TO), i.e., the same foot rises from the ground. The swing phase corresponds to the part of the cycle when the heel is off (HO) the ground. The stance phase can be further divided into three sub-phases: controlled plantar flexion (contact sub-phase), controlled dorsiflexion (midstance) and powered plantar flexion [9]. Controlled plantar flexion begins at heel strike and ends at foot flat (FF). Controlled dorsiflexion begins at foot flat and continues until dorsiflexion reaches the maximum. Powered plantar flexion initiates after the controlled plantar flexion and ends at the leaving of the foot from the ground.

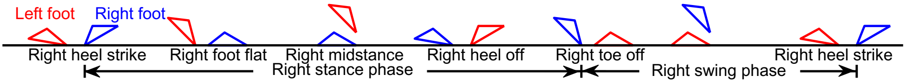


Fig. 3. Gait cycle

3 Methods

The walking human is often represented as a nonlinear dynamical system which can be modeled by nonlinear algebraic and/or nonlinear differential equations. Although the nonlinear multibody systems method can precisely describe the human locomotion, it usually cannot solve the question analytically and often shows complex dynamics. One way to solve the problem is the linearization. After linearization, the linear equations of motion are transformed to state equations which are easier for analyzing the locomotion traits.

However, we investigate this from another point of view. It is well known that trajectories traced by limb joints of healthy human are periodic in time. Thus, the graphical and analytical techniques like phase plane portraits, Poincaré maps and Floquet multipliers (FM), can be used to analyze the nonlinear dynamics of human motions [10,11]. These methods are also applied in this study for observing human walking with prosthetic feet.

The first step to analyze a nonlinear system is to identify a set of state variables (generalized coordinates and generalized velocities) that depict the dynamics fully[11]. In this study, the angular positions and angular velocities of body joints are taken at a given instant in time to represent the state of the system. We assume here the joints are rotational and all the elements are nondeformable. Seven angular positions and seven corresponding angular velocities are selected as the state variables in this study. The notation of seven angular positions corresponding to Fig. 2 is:

ϕ_1 – HAT angle with respect to the vertical,
ϕ_2 (ϕ_5) – flexion angle of right (left) thigh with respect to the vertical,
ϕ_3 (ϕ_6) – flexion angle of right (left) shank relative to right (left) thigh,
ϕ_4 (ϕ_7) – plantar flexion of right (left) foot relative to right (left) shank.

Hence, the state vector can be written as

$$\boldsymbol{x} = [\phi_1, \phi_2, \phi_3, \phi_4, \phi_5, \phi_6, \phi_7, \dot{\phi}_1, \dot{\phi}_2, \dot{\phi}_3, \dot{\phi}_4, \dot{\phi}_5, \dot{\phi}_6, \dot{\phi}_7]^T. \tag{1}$$

In our study, the Flex foot, SACH foot, Seattle foot and one non-specific optimized foot are taken as the research objects. The kinematic parameters value of the lower limb joints for walking with these prosthetic feet were obtained from numerical experiments by Ackermann [3,12] and Hoang [13]. For each prosthetic foot, there are data sets of four gait cycles at speed 1.33 m/s. We extract joint angular positions of lower limbs for walking with different prosthetic feet. The optimized foot is used as a reference and standard for comparison.

All angular positions are numerically differentiated to obtain corresponding angular velocities, and both positions and velocities are averaged over four gait cycles. Phase plane portraits of joints are obtained by plotting averaged angular positions (ϕ_k) against respective averaged angular velocities ($\dot{\phi}_k$), $k = 1, 2, ..., 7$. The averaged values at events HS, FF, HO and TO, can also be marked on the portraits, see Section 4 for details.

At a given instant in time, the joint angular positions and joint angular velocities of lower limbs are taken to represent the states of the system. How these states evolve in time defines the trajectories in state space. Quantifying orbital stability is based on evolution of the state at a discrete event across subsequent cycles. The discrete event is a hypersurface in the state space, known as Poincaré section. Poincaré section is the fictitious plane transversing the trajectories in phase space at regular intervals and obtaining points of intersection. During entire gait cycles, state at any discrete event, such as heel strike, can be used to define a Poincaré section. Poincaré map is the map from the current intersection to the subsequent intersection on a Poincaré section. In this context, Poincaré maps for each joint at a discrete event are obtained by plotting the value of $\boldsymbol{x}$ at i^{th} gait cycle $\boldsymbol{x}_i$, versus its value at $(i+1)^{th}$ gait cycle $\boldsymbol{x}_{i+1}$, i.e., form the equation of

$$\boldsymbol{x}_{i+1} = \boldsymbol{P}(\boldsymbol{x}_i). \tag{2}$$

The equilibrium state defines a periodic motion that returns to itself upon subsequent intersections, i.e.,

$$\boldsymbol{x}_{eq} = \boldsymbol{P}(\boldsymbol{x}_{eq}), \quad \text{where} \quad \boldsymbol{x}_{eq} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i. \tag{3}$$

The difference between $\boldsymbol{x}_i$ and equilibrium state $\boldsymbol{x}_{eq}$, is calculated by

$$\delta\boldsymbol{x}_i = \boldsymbol{x}_i - \boldsymbol{x}_{eq}. \tag{4}$$

Linearizing the map with respect to the equilibrium state yields a Jacobian matrix $\boldsymbol{J}_p$, which meets

$$\delta\boldsymbol{X}_{i+1} = \boldsymbol{J}_p\,\delta\boldsymbol{X}_i. \tag{5}$$

Here, the rows of matrix $\delta\boldsymbol{X}_i$ are the vector components of $\delta\boldsymbol{x}_i$ and the columns represent the separate gait cycles. Finally, the 14×14 constant Jacobian matrix $\boldsymbol{J}_p$, is calculated from the linear least-square fit using the pseudo-inverse routine in MATLAB [8].

By the Floquet theory, one can obtain the precise structure of fundamental matrix for the periodically time-variant system. The system is stable when the absolute values of all eigenvalues of Jacobian matrix (i.e. FM) are less than 1, which means that disturbances are vanished over subsequent cycles [14,15]. The more detailed description of the mathematical fundamentals can be found in [16].

4 Simulation and Results Comparison

Phase plane portraits for each joint were obtained by plotting the averaged positions against their respective velocities. The right foot contact events, including HS, FF, HO and TO were averaged and marked on the phase plane portraits. Poincaré maps were constructed by plotting the angular positions at a particular instant e.g., at the HS event of one step against the corresponding angular positions at a similar instant of the next step. Finally, the Floquet multipliers were calculated to determine the stability of human gaits with four prosthetic feet.

4.1 Investigate Joint Kinematics by Using Phase Plane Portraits

We only plot phase plane portraits of right lower limb joints because we assume in this study the subject walked with both left and right prosthetic feet. Figure 4(a) demonstrates that walking with Flex foot has the largest hip flexion during stance phase (Heel Strike→Toe Off) followed by optimized foot, Seattle foot, and SACH foot in sequence. Figure 4(b) exhibits that walking with Flex foot has the largest knee flexion during contact sub-phase (Heel Strike→Foot Flat), followed by Seattle foot, optimized foot, and finally SACH foot in sequence. However, the differences of respective maximums are very small. As to ankle joints, Flex foot has the greatest ankle dorsiflexion (i.e. plantar extension) between the instant of foot flat and toe off, see Fig. 4(c). The optimized foot, Seattle foot and SACH foot are with smaller dorsiflexions.

The averaged angular position ranges of right lower limb joints are obtained by subtracting the minimum from the maximum of averaged angular positions, see Fig. 5. For all prosthetic feet, ankle joints always display the smallest angle ranges, whereas knee joints often exhibit the largest angular ranges. Figure 5(a) shows the knee angle range of walking with Seattle foot is greatest, while SACH foot has close performance to Seattle foot. Walking with Flex foot has the smallest knee angle range. Figure 5(b) demonstrates walking with Seattle foot has the greatest hip angle range, followed by the optimized foot, Flex foot and SACH foot in sequence. Among four prosthetic feet, walking with Flex foot keeps the greatest ankle angle range, while SACH foot is within the smallest, see Fig. 5(c).

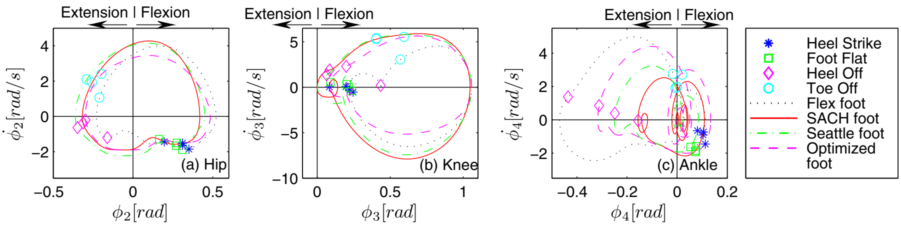


Fig. 4. Phase plane portraits of right hip, knee and ankle joints for walking with four different prosthetic feet

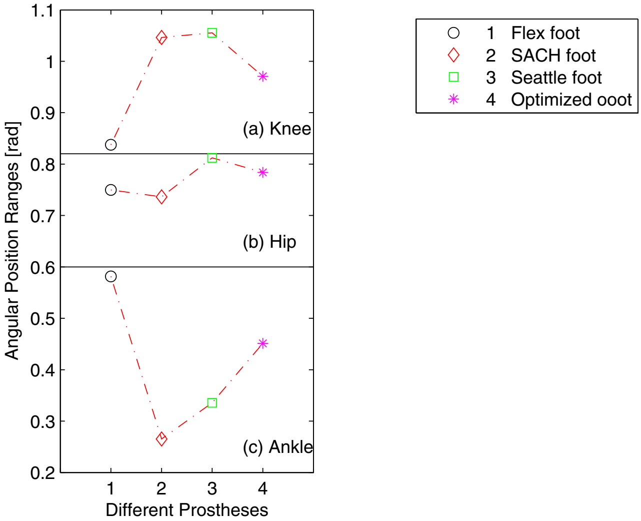


Fig. 5. Averaged angular position ranges of right lower limb joints for walking with different prosthetic feet

4.2 Investigate Joint Angular Positions by Poincaré Maps

Poincaré maps for walking with different prosthetic feet at right HS events show that all points cluster around the 45° diagonal line, see Fig. 6, which confirms the periodic nature of trajectories. Figures 6(a) demonstrates observable shift towards the lower left corner in hip flexion for walking with SACH foot, whereas a clear upper right shift for walking with Flex foot. And the characteristic of knee flexion is similar to that of hip flexion for walking with SACH foot and Flex foot from Fig. 6(b). There are no significant differences in ankle plantar flexion at right heel strike events for walking with four prosthetic feet, see Fig. 6(c).

4.3 Confirm the Gaits Stability by Using Floquet Multipliers

The stability of human gaits can be confirmed by Floquet multipliers, i.e., eigenvalues of the Jacobian matrix $\boldsymbol{J}_p$. Floquet multipliers for all prosthetic feet are calculated at all four sections, i.e. HS, FF, HO and TO instants. There are no differences for distributions and magnitudes of Floquet multipliers at these four instants, see Fig. 7. For all prosthetic feet in this study, they have the same feature. That means that Floquet multipliers do not depend on the choice of Poincaré sections. Due to stable locomotion, all Floquet multipliers for different prosthetic feet fall into the range between -1 and 1 as expected. Relative stability of prothetic gaits are quantified by comparing the magnitudes of their largest Floquet multipliers which are defined as the γ measure, see detailed definition in [10]. The measured value γ=1 for all conditions implies that the orbital stability of the investigated prosthetic feet consistent with the results from related experiments.

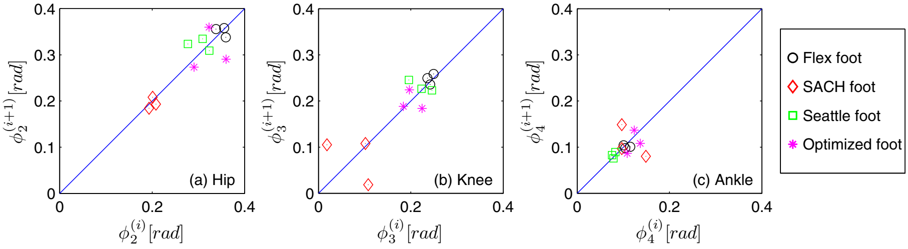


Fig. 6. Poincaré maps of angular positions of right lower limb joints at heel strike events for walking with four different prosthetic feet

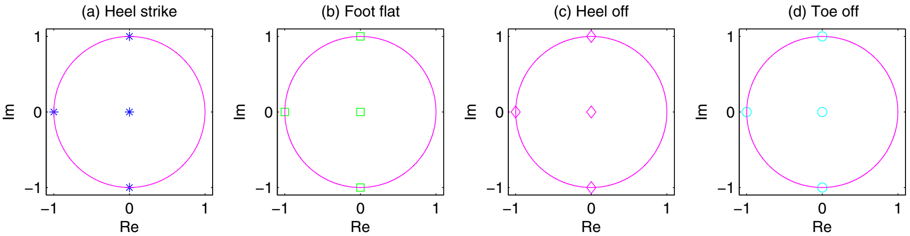


Fig. 7. Distribution of Floquet multipliers at right foot contact events for walking with Flex foot

5 Conclusions

This investigation gives quantified comparison and evaluation of human gaits considering different prosthetic feet by using methods of phase plane portraits, Poincaré maps and Floquet multipliers. Simulation of lower limbs walking with Flex foot, SACH foot, Seattle foot as well as one optimized foot are performed.

It turns out that human gaits using these four prosthetic feet display the orbital stability. However, there are still different effects on kinematics of lower limb joints for these investigated prosthetic feet according to phase plane portraits and corresponding Poincaré maps.

Walking with Flex foot always exhibits the greatest hip flexion during stance period, the greatest knee flexion during the contact phase and the largest ankle dorsiflexion from the foot flat to toe off section of stance phase. While walking with SACH foot has the smallest hip flexion, knee flexion and ankle dorsiflexion during the same periods. Human gaits using Seattle foot and the optimized foot display the intermediate performance compared with Flex foot and SACH foot. These observations partially may be explained by the lowest stiffness of Flex foot and highest stiffness of SACH foot in dorsiflexion.

Acknowledgments. The authors would like to express their thanks to Prof. Werner Schiehlen (University of Stuttgart) for his talent and valuable suggestions. This research is also partially supported by Chinese Scholarship Council (CSC).

References

1. Lehmann, J.F., Price, R., Boswell-Bessette, S., Dralle, A., Questad, K., de Lateur, B.J.: Comprehensive Analysis of Energy Storing Prosthetic Feet: Flex Foot and Seattle Foot Versus Standard SACH Foot. Arch. Phys. Med. Rehabil. 74, 1225–1231 (1993)
2. Klute, G.K., Kallfelz, C.F., Czerniecki, J.M.: Mechanical Properties of Prosthetic Limbs: Adapting to the Patient. J. Rehabil. Res. Dev. 38, 299–307 (2001)
3. Ackermann, M.: Dynamics and Energetics of Walking with Prostheses. Ph.D. Thesis, Institute of Engineering and Computational Mechanics, University of Stuttgart, Shaker-Verlag, No. 9, Aachen (2007)
4. Dingwell, J.B., Cusumano, J.P., Cavanagh, P.R., Sternad, D.: Local Dynamic Stability Versus Kinematic Variability of Continuous Overground and Treadmill Walking. J. Biomech. Eng. 123, 27–32 (2001)
5. Norris, J.A., Marsh, A.P., Granata, K.P., Ross, S.D.: Positive Feedback in Powered Exoskeletons: Improved Metabolic Efficiency at the Cost of Reduced Stability? In: Proceedings of the ASME 2007 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, Las Vegas, Nevada, September 4-7 (2007)
6. Dingwell, J.B., Kang, H.G., Marin, L.C.: The Effects of Sensory Loss and Walking Speed on the Orbital Dynamic Stability of Human Walking. J. Biomech. 40, 1723–1730 (2007)
7. Ali, F., Menzinger, M.: On the Local Stability of Limit Cycles. Chaos 9, 348–356 (1999)
8. Granata, K.P., Lockhart, T.E.: Dynamic Stability Differences in Fall-prone and Healthy Adults. J. Electromyogr. Kinesiol. 18, 172–178 (2008)
9. Jiménez-Fabián, R., Verlinden, O.: Review of Control Algorithms for Robotic Ankle Systems in Lower-Limb Orthoses, Prosthese, and Exoskeletons. Med. Eng. Phys. 34, 397–408 (2012)

10. Marghitu, D.B., Hobatho, M.-C.: Dynamics of Children with Torsional Anomalies of the Lower Limb Joints. Chaos Soliton Fract. 12, 2411–2419 (2001)
11. Hurmuzlu, Y., Basdogan, C.: On the Measurement of Dynamic Stability of Human Locomotion. J. Biomech. Eng. 116, 30–36 (1994)
12. Ackermann, M., Gros, H.: Measurements of Human Gaits. Interim Report ZB-144, Institute B of Mechanics, University of Stuttgart, Stuttgart (2005)
13. Hoang, K.-L.H.: Modellierung und Simulation von Oberschenkel-prothesen. Master Thesis DIPL-126, Institute of Engineering and Computational Mechanics, University of Stuttgart, Stuttgart (2008) (in German)
14. Müller, P.C., Schiehlen, W.O.: Linear Vibrations. Martinus Nijhoff Publishers, Dordrecht (1985)
15. Nayfeh, A.H., Balachandran, B.: Applied Nonlinear Dynamics: Analytical, Computational, and Experimental Methods. John Wiley & Sons, Inc., New York (1995)
16. Gong, L.: Stability Analysis of Human Locomotion with Processed Data. Institute Report IB-42, Institute of Engineering and Computational Mechanics, University of Stuttgart, Stuttgart (2008)

An Improved Intelligent Water Drop Algorithm for a Real-Life Waste Collection Problem

Mohammad Raihanul Islam and M. Sohel Rahman

AℓEDA Group, Department of CSE, BUET, Dhaka-1000, Bangladesh
{raihan2108,sohel.kcl}@gmail.com

Abstract. In this paper, we have proposed an improved Intelligent Water Drop (IWD) Algorithm. The IWD algorithm has been proposed by observing the dynamic flow of water in the river system and the actions of the water drops. The water drops act as agents to find the optimal solution. In this paper, we have modified the original IWD algorithm and proposed an improved variant of it. We have implemented our proposed algorithms to solve a real-life waste collection problem. Our algorithms have shown promising results.

Keywords: Intelligent Water Drop, Vehicle Routing, Waste Collection Problem, Swarm Intelligence, Artificial Dam.

1 Introduction

The past few decades have seen a thrive towards developing more nature-inspired algorithms especially in the field of Computational Intelligence. A substantial number of algorithms have been developed imitating the natural systems. One of the most recent nature-inspired algorithms is the Intelligent Water Drop (IWD) algorithm. This algorithm partially imitates the fluid mechanism of water in the river and its effect on soil of the riverbed. The IWD algorithm was first proposed in [1]. Subsequently it has successfully been applied on Traveling Salesman Problem (TSP) [2], Vehicle Routing Problem (VRP) [3], Economic Load Dispatch Problem [4] etc.

The Vehicle Routing Problem (VRP) aims at finding feasible routes of vehicles between customers and depot. An extension of VRP is Vehicle Routing Problem with Time Windows (VRPTW). Here each customer has a specific time window within which he/she must be served. In this paper, we have considered a real-life waste collection problem which is a variant of VRPTW. Here a number of clients, depot and a number of disposal facilities are present and all of them are associated with time windows. As a result, each customer is associated with a time window which defines when waste can be collected from that customer; each disposal facility has a definite time slot when they are open and finally all the vehicles have to return to the depot before it closes. Vehicles are used for collecting waste from the customers. Each vehicle has 3 types of capacity limitations as follows. Each vehicle is allowed to dispose off a limited amount of waste (C_{max}) and to serve a limited number of customers (S_{max}) throughout

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 472–479, 2013.

the day. Moreover, the total amount of waste a vehicle can carry at a time (C_{cap}) is also limited. Finally there is a lunch period for the drivers. The driver must take his lunch during that period. The goal of the problem is to find feasible routes for the vehicles satisfying all the customers without violating any of the constraints such that the distance traveled by the vehicles is as less as possible. Additionally, the number of vehicles used should also be as small as possible.

The contribution of this paper is as follows. Here, we first study the IWD algorithm and propose some modifications to it. Then we use both the versions to solve the real-life waste collection problem discussed above. Our algorithms have outperformed the state of the art algorithms in the literature.

The rest of the paper is organized as follows. Section 2 gives a brief description of previous works. Section 3 presents the concept of the basic IWD algorithm. In Section 4, we present our modifications to get a modified IWD (M-IWD) algorithm and discuss our approach to solve the waste collection problem using both IWD and M-IWD algorithms. In Section 5, we report our experimental findings.

2 Previous Works

The Vehicle Routing Problem is one of the most significant optimization problems. Since VRP is a hard combinatorial problem, the exact solution may not always be feasible, especially if the problem size is too large. Therefore, a number of meta-heuristics has been developed to compute the near-optimal solutions such as Simulated Annealing, Ant Colony Optimization, Tabu Search etc.

One of the earliest attempt to solve VRP using Tabu Search has been introduced in [5]. In [6], authors have described a VRP with intermediate facilities. Here, they have proposed a Tabu Search algorithm where initial solution is obtained by assigning each customer to a schedule by random choice. Another multi-depot VRP with intermediate facility has been proposed in [7]. Here, a Tabu Search heuristics is proposed. Authors have decomposed the problem into three sub-problems. After that, a solution pool is generated by executing a set partitioning algorithm. Finally a post-optimization procedure has been implemented. Perhaps, the most effective meta-heuristics to solve VRP is Ant Colony Optimization (ACO). In [8], authors have presented a novel ant system for VRP. In [9], authors have presented an Improved Ant Colony Optimization (IACO) which gives better results than the conventional ACO algorithms. In [10], a multi-ant colony system is proposed to solve a variant of VRP, where two types of customers are involved. In [11], authors have introduced an extension of VRPTW where a customer has multiple time windows. An ant colony system has been proposed for this problem. Another popular approach to solve the VRP is the Variable Neighborhood Search (VNS) heuristic, first proposed in [12]. Here local search for improving the existing solution is performed by interchanging the neighbors. In [13], a variant of VRP involving time windows and multiple depots is solved. In [14], authors have proposed an adaptive VNS technique for multi-depot delivery problem.

The waste collection problem of this paper has also been studied in the literature. In [15], authors have presented an extension of insertion heuristics to solve this problem. On the other hand, in [16], authors have presented a VNS based Tabu Search for the problem. Here, the solution is first obtained using a *nearest neighbor* algorithm. Subsequently, Tabu Search, VNS and Variable Neighborhood Tabu Search (VNTS) are applied to improve the solution.

3 Preliminaries

In this Section, we present the concept of the Intelligent Water Drop algorithm. In nature, when water drops flow through rivers or canals, they change the environment around them. On the other hand, the environment plays a vital role in the paths of rivers. If the water drops face no obstacles in its path it would reach the destination (lake or sea) following a straight path because of the gravitational force. But the path of the river is twisted because the obstacles of the path. Therefore, the water drops try to make a better path out of the ideal path (straight path) to reach the destination.

Consider an IWD (agent) in a complete graph with N nodes where it has to move from one node to another. The edge between two nodes holds an amount of soil. Assume that, the amount of soil between x and y can be expressed as $soil(x, y)$. This amount may be increased or decreased based on the IWD flowing through the edge. Let an IWD be at node i. The probability of the IWD to move to node j is as follows:

$$p^{IWD}(i,j) = \frac{f(soil(i,j))}{\sum_{k \in T(i)} f(soil(i,k))}, \tag{1}$$

where, $f(soil(i,j)) = \dfrac{1}{\varepsilon_s + soil(i,j)}$,
$T(i)$ is the set of nodes which can be traveled from node i,
ε_s is a small small fraction used to prevent division by zero in the above equation. Now if the velocity of IWD at time t is represented by $vel^{IWD}(t)$, then the following equation is used to update the velocity of IWD:

$$vel^{IWD}(t+1) = vel^{IWD}(t) + \frac{a_v}{b_v + c_v(soil(i,j))}, \tag{2}$$

where, a_v, b_v and c_v are constant parameters whose value can be set in the experiment initially. We need to measure the undesirability of an IWD to move from node i to node j. Consider a heuristic function HUD which can measure this undesirability. In our waste collection problem, we can denote the heuristic function as HUD_{WC} and define it as follows:

$$HUD_{WC}(i,j) = d_{ij}, \tag{3}$$

where, d_{ij} is the Euclidean distance between nodes i and j. The time needed to travel from node i to node j can be calculated using the following equation:

$$time(i,j;vel^{IWD}) = \frac{HUD_{WC}(i,j)}{\max(\varepsilon, vel^{IWD})} \tag{4}$$

As IWD moves from one node to another it erodes an amount of soil from that path. The amount of soil erosion depends on the velocity of IWD. If the velocity is high more soil will be removed. Alternatively we can say that, a faster river can remove more soil from its bed than a slower one. From that observation, we can calculate the amount of soil removal from the path of node i and j using the equation below:

$$\Delta soil(i,j) = \frac{a_s}{b_s + c_s \cdot time(i,j;vel^{IWD})}, \tag{5}$$

where, a_s, b_s and c_s are constant parameters. The value of $time(i,j;vel^{IWD})$ can be calculated from Equation (4). Now, the amount of soil on the path (edge) between node i and j can be updated by using the equation below:

$$soil(i,j) = \rho_0 \cdot soil(i,j) + (1 - \rho_n) \cdot \Delta soil(i,j), \tag{6}$$

where ρ_0 and ρ_n are constant parameters. The amount of soil IWD carry can be updated using the following equations:

$$soil^{IWD} = soil^{IWD} + \Delta soil(i,j) \tag{7}$$

The overall IWD algorithm continues in an iterative manner. Every IWD created in this algorithm moves from one node to another to build the solution pah. At the end an iteration, the performance of the solutions are measured. A detail description of the algorithm can be found in [17].

4 Our Approach

In this Section, we describe our approach to solve the Waste Collection Problem defined in Section 1. Consider a graph $G(V,E)$, where V is the set of nodes and E is the set of edges. We consider all the customers, disposal facilities and the depot as nodes in the graph and the paths among those are represented through the edges of the graph.

4.1 Dam Construction

In the basic IWD algorithm, the amount of soil in all the paths between the nodes are initially same. As a result, in reality selecting the next node is a random choice at the first iteration since all the probabilities are equal to one another (Equation (1)). However, we may have some information at the beginning which we could use to make a more informed choice. We introduce the idea of constructing dams to make some of the undesirable edges more costly. We adopt the following strategy. Initially, we construct a dam on the path between two nodes depending on the “undesirability” of the path. The main idea here is that, if there exists a dam the water flow will be hindered and hence the corresponding solution will be costly.

We apply the idea of a dam construction as follows. We adopt the idea of *nearest neighbor* [18] to construct dams on the paths. First, we apply the nearest neighbor algorithm on our input. At each instance, we choose the nearest neighbor from the location of the current customer and serve him/her. After constructing the solution T_{nn} using *nearest neighbor*, we initialize the soil on the path using the following equations:

$$soil(i,j) = INIT\_SOIL, \text{ if edge(i, j)} \in T_{nn}, \tag{8}$$

$$soil(i,j) = \beta \cdot INIT\_SOIL, \text{ if edge(i, j)} \notin T_{nn}, \tag{9}$$

where, $\beta \geq 1$. In this way, each path/edge not in the solution computed through the nearest neighbor algorithm will in effect have a dam constructed on it. This will ensure some extra cost if that path is selected. As a result, the IWD algorithm will tend to be more prone to avoid the paths with dams, i.e., costly paths while choosing a solution.

4.2 Modification in Selecting Next Customer

We have also modified the procedure for selecting the next node. Inspired by the selection procedure of the Ant Colony Optimization (ACO) technique, we have improved the selection procedure as follows: Let f_0 be a fixed number where, $0 \leq f_0 \leq 1$ and f be random number between 0 and 1. if $f \leq f_0$, then, IWD currently in node i, will select node j if,

$$j = arg \min_{k \in (T(i)} d_{ik}, \tag{10}$$

where, d_{xy} is the Euclidean distance between customer x and y and T(i) is the set of customers (nodes) which can be served after i.

On the other hand if, $f > f_0$ then the selection procedure will be as follows: Let q_0 be a fixed number where, $0 \leq q_0 \leq 1$ and q a random number between 0 and 1. if $q \leq q_0$, then, IWD will select node j if,

$$j = arg \max_{k \in T(i)} f(soil(i,k)) \tag{11}$$

Recall that, $f(soil(i,j)) = \dfrac{1}{\varepsilon_s + soil(i,j)}$.

If $q > q_0$ then we employ the conventional selection procedure of IWD algorithm according to Equation (1).

4.3 Overall Algortihm

The overall algorithm to solve the Waste Collection problem works as follows. At the beginning, we apply the *dam construction* procedure described in Section 4.1 to initialize the amount of soil to the paths between customers (nodes).

Then we apply our modified IWD to solve the waste collection problem. We assume that, there are unlimited vehicles stationed at the depot. We select a vehicle. The vehicle begins servicing the customers selected by the Equations (1), (10) and (11) until S_{max} or C_{max} exceeds. If the vehicle has finished its quota, then it returns to the depot. Additionally before servicing a customer we check whether there is enough space available for the vehicle to collect waste from the customer; if not then, it goes to the nearest disposal facility to dispose off the waste. Moreover, we also check, whether it is the lunch time; if yes, then the current time of the vehicle is updated by adding the time for lunch. After servicing each customer, the velocity and time of IWD and the soil of the paths are updated using Equations (2) to (6). We have also employed some local search (swapping among the routes) to get better routes for the vehicles.

5 Experiments

We have implemented our algorithm in C++ and run a simulation on 2.27 GHz PC (intel core $i5$) with 4.00 GB memory. We have used the data sets available in [19]. We have run initial simulation to find acceptable parameters for our methods. The satisfactory parameters for IWD were found as follows: $a_v = 1000$, $b_v = 0.01$, $c_v = 1$, $a_s = 1000$, $b_s = 0.01$, $c_s = 1$, $\epsilon_s = 0.01$, $\rho_0 = 0.4$ and $\rho_n = 1$. For each dataset, we have ran our algorithm $TotalIteration = 20$ times and report the best results in Table 1 along with average computational time and number of iterations needed to reach the best solution.

Table 1. Experimental Results

Problem Set	Best Solution		Average Computational Time (seconds)	Number of Iteration to Reach the Best Solution
	Vehicle	Distances (miles)		
102	3	170.161	20.3141	5
277	2	395.697	29.2329	17
335	6	189.318	69.2177	19
444	11	67.1903	49.7849	12
804	6	759.711	1127.29	8
1051	17	2121.99	104.314	16
1351	7	812.502	2208.6	17
1599	13	1144.89	727.969	18
1932	16	1042.22	703.41	1
2100	16	1581.45	845.615	1

We have compared our results with the previous works from [16] and [15]. We have picked the best results from [16] and [15] to compare against ours. We have also made a comparison between the results of IWD and and that of M-IWD.

Table 2. Comparison among results

Problem Set	M-IWD		Original IWD		Best Results from [16] and [15]	
	Vehicles	Distances (miles)	Vehicles	Distances (miles)	Vehicles	Distances (miles)
102	3	170.161	3	187.576	3	183.5
277	2	395.697	2	394.745	3	464.5
335	6	189.318	6	180.031	6	204.5
444	11	67.1903	11	68.6786	11	89.1
804	6	759.711	6	759.804	5	725.6
1051	17	2121.99	17	2142.66	17	2250.6
1351	7	812.502	7	808.89	7 (8)	1011.9 (915.1)
1599	13	1144.89	13	1109.2	13	1364.7
1932	16	1042.22	17	1520.15	16	1262.8
2100	16	1581.45	19	2464.41	16	1749

For*dataset 1351*, authors of [16] have reported two best results, one with less vehicles and other with less traveled distances. Therefore, we have shown our comparison with two results (one in parentheses). From Table 2, we can see that, our M-IWD algorithm has outperformed the results from [16] and [15] in every datasets except *dataset 804*. Moreover, M-IWD has also shown good performance against the original IWD. Notably, the M-IWD have shown better results in last two datasets (largest dataset) compared to all other methods. From the results, we can conclude that, our proposed IWD algorithm has shown better results in large datasets compared to original IWD.

Acknowledgment. Our research was partially supported by CodeCrafters-Investortools Research Grant for CSE BUET (http://www.codecraftersintl.com/researchgrant.html), a Commonwealth Fellowship and an ACU Titular Fellowship. We are very grateful for their sincere effort.

References

1. Duan, H., Liu, S., Lei, X.: Air robot path planning based on Intelligent Water Drops optimization. In: IEEE World Congress on Computational Intelligence, pp. 1397–1401 (2008)
2. Kesavamoorthy, R., ArunShunmugam, D., ThangaMariappan, L.: Solving Traveling Salesman Problem by Modified Intelligent Water Drop Algorithm. In: International Conference on Emerging Technology Trends, vol. 2, pp. 18–23 (2007)
3. Intelligent Water Drops a new optimization algorithm for solving the Vehicle Routing Problem (2010)
4. Rayapudi, S.R.: An Intelligent Water Drop Algorithm for Solving Economic Load Dispatch Problem. International Journal of Electrical and Electronics Engineering 5(2), 43–49

5. Willard, J.A.G.: Vehicle Routing Using r-Optimal Tabu Search. Master's thesis, Imperial College, London (1989)
6. Angelelli, E., Speranza, M.: The periodic vehicle routing problem with intermediate facilities. European Journal of Operational Research 137(2), 233–247 (2002)
7. Crevier, B., Cordeau, J.F., Laporte, G.: The multi-depot vehicle routing problem with inter-depot routes. European Journal of Operational Research 176, 756–773 (2007)
8. Bullnheimer, B., Hartl, R.F., Strauss, C.: An Improved Ant System Algorithm for the Vehicle Routing Problem. Annals of Operations Research 89, 319–328 (1997)
9. Bin, Y., Zhong-Zhen, Y., Baozhen, Y.: An improved ant colony optimization for vehicle routing problem. European Journal of Operational Research, 171–176 (2009)
10. Gajpal, Y., Abad, P.: Multi-ant colony system (MACS) for a vehicle routing problem with backhauls. European Journal of Operational Research (2008)
11. Favaretto, D., Moretti, E., Pellegrini, P.: Ant colony system for a vrp with multiple time windows and multiple visits. Journal of Interdisciplinary Mathematics 10(2), 263–284 (2007)
12. Mladenovic, N., Hansen, P.: Variable neighborhood search. Computers and Operations Research 24(11), 1097–1100 (1997)
13. Polacek, M., Hartl, R.F., Doerner, K., Reimann, M.: A Variable Neighborhood Search for the Multi Depot Vehicle Routing Problem with Time Windows. Journal of Heuristics 10(6), 613–627 (2004)
14. Stenger, A., Vigo, D., Enz, S., Schwind, M.: An Adaptive Variable Neighborhood Search Algorithm for a Vehicle Routing Problem Arising in Small Package Shipping. Transportation Science 46(4) (2012)
15. Kim, B.I., Kim, S., Sahoo, S.: Waste collection vehicle routing problem with time windows. Comput. Oper. Res. 33, 3624–3642 (2006)
16. Benjamin, A.M., Beasley, J.E.: Metaheuristics for the waste collection vehicle routing problem with time windows, driver rest period and multiple disposal facilities. Comput. Oper. Res. 37, 2270–2280 (2010)
17. Shah-Hosseini, H.: Optimization with the Nature-Inspired Intelligent Water Drops Algorithm, ch. 16 (2009)
18. Solomon, M.M.: Algorithms for the vehicle routing and scheduling problems with time window constraints. Operational Research 35, 254–265 (1987)
19. `http://www.postech.ac.kr/lab/ie/logistics/WCVRPTW_Problem/benchmark.html` (last accessed: December 21, 2012)

The Extension of Linear Coding Method for Automated Analog Circuit Design

Zhi Li and Jingsong He[*]

Department of Electronic Science and Technology,
University of Science and Technology of China, Hefei, China
zhili@mail.ustc.edu.cn, hjss@ustc.edu.cn

Abstract. Encoding method is one of the key factors of evolutionary design of analog circuit. Due to the adaptability, convenience and relatively short length of linear coding method, it has been widely used for automation of analog circuit design. Evolutionary design of analog circuits, which is not limited to traditional knowledge, could generate circuits with novel structures and parameters. The novel structures provide more possible solutions for fault-tolerance design of analog circuits. While, the current linear coding method based on five connection ways limits the number of possible circuit structures. So in this paper, we improve the existing linear coding method by expanding the instruction set. The experimental results show that the improved linear coding method can generate richer circuit structures, and it opens up a new way for the fault-tolerance design of analog circuits.

Keywords: Analog circuit, evolutionary design, linear encoding.

1 Introduction

Analog circuits, which are important parts of modern electronic systems, have been widely used in many fields. With the expanding scale and increasing complexity of analog circuits, the design tasks become more and more difficult for manual methods. Evolutionary design methods, which are not based on traditional knowledge, provide new ways for the automation synthesis of analog circuits with novel structures [1].

Evolutionary design methods use chromosomes to represent the real netlists of analog circuits. The performance of evolutionary design is highly related to the encoding and decoding methods. Many researchers have carried out some researches on the encoding methods of analog circuits. Koza et al. have proposed a tree encoding method, in which a circuit is represented by a tree structure. This method can produce rich circuit structures. But, the encoding and decoding processes of tree-codes are very complex [2] [3]. Grimbleby proposed a netlist-based encoding method, in which a component is represented with its types, parameters and pins. Although this coding method reduces

[*] Corresponding author.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 480–487, 2013.

the complexity of encoding and decoding, it increases the complexity of genetic operations at the same time [4]. Mattiussi et al. proposed the analog genetic coding method (AGE), in which the connections between the components are determined by the predefined mutual relationship between the different genes [5]. Lohn et al. proposed a linear coding method [6] [7], which is widely applied to the evolutionary design of analog circuit [8] [9]. It is beneficial for reducing the length of codes and convenient for genetic operation. The current linear coding methods based on five connection ways limits the number of possible analog circuits. To address this issue, in this paper, we improve the existing linear coding method for analog circuits by expanding the instruction set. Through the analog circuits evolutionary design experiment, the results show that the improved linear coding method can produce richer circuit structures, and open up a new search for the evolution of fault-tolerant analog circuits.

2 The Existing Linear Coding Method for Analog Circuit Design

By using linear coding method, proposed by Lohn [7], the variable parts of analog circuits are transferred into individual chromosomes. In the existing linear coding method, each component is represented by three-bit genes. The location of each component is not directly defined in the chromosome. They use one bit gene to represent the connection ways of new components. In this way, the growing method of the circuits is defined by the chromosome. During the decoding process, an active node is used to mark the latest junction node, and adding components to construct the circuit in order on the basis of embryonic circuit. Each component is encoded with its type, parameter and connection instructions. The connection instructions are used for guiding the growing process of analog circuits. Each instruction has a specific meaning, which specifies the access way of new component and guides the changes of the active node. So, the different association sequences of connection instructions represent different circuit structures. In the existing linear coding method, there are five different kinds of connection instructions (x-move-to-new, x-cast-to-previous, x-cast-to-ground, x-cast-to-input and x-cast-to-output). X represents a component type. X may be the R (resistance), L (inductance) and C (capacitance) or T (triode). The x-move-to-new instruction represents access of component x between the active node and the newly created junction and changes the active node into the newly-created node. The x-cast-to-previous instruction, x-cast-to-ground instruction, x-cast-to-input instruction and x-cast-to-output respectively represents access of component x between the active node and the previous node, the ground node, the input node, the output node, and the active node remains unchanged. Changes in the output node and the active node after the execution of each instruction are shown in Table 1.

However, the five kinds of connection instructions limit the numbers possible circuit structures. There are several partial structures, such as those partial structures shown in Fig 1, cannot be generated by using the existing linear coding method. But these structures provide possible solutions for analog circuit design. What's more, such structures are beneficial to enhance the fault-tolerant performance of a circuit. Then we make a deep analysis on each of these structures shown in Fig. 1.

Table 1. The five different kinds of connection instructions in existing linear coding method

Instruction	Outgoing Node	Active Node
x-move-to-new	newly-created node	becomes the newly-created node
x-cast-to-previous	previous node	unchanged
x-cast-to-ground	ground node	unchanged
x-cast-to-input	input node	unchanged
x-cast-to-output	output node	unchanged

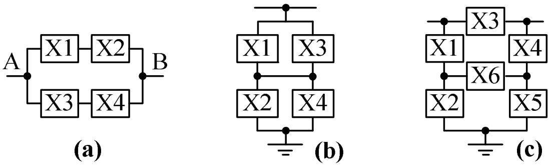


Fig. 1. Three different circuit structure segments

1) In Fig. 1 (a), there are two branches between node A and node B. Each branch consists of two components in series. Even though there is single point short and single point disconnect failure in one single component, the connection between node A and node B could still remain. This structure is conducive to enhance the fault-tolerance performance of the circuit for single point failure.

2) In the existing linear coding method, the branch to the ground only has one component. Once a fault occurs in the branch to the ground, the performance of the circuit is likely to decline seriously. In Fig. 1 (b), the structure has strong resistance for a single point failure in the branch to the ground.

3) In Fig. 1 (c), this network structure can enhance the connection between nodes, and reduce the dependence of node to single component. Thus, this kind of structures is helpful to improve fault-tolerant performance of the circuit.

Circuit structures represented by the existing linear coding method are not rich enough, as its instruction set are constrained. So, in this paper, we extend its instruction set in order to generating a variety of circuit structures.

3 The Extended Linear Encoding for Analog Circuit

The linear coding method employs circuit-construction robot (cc-robot) instructions to construct analog circuits. The research space of topology is influenced by cc-robot instructions. In order to enhance the richness of circuit structures, we will extend the cc-robot instruction set. We define instructions of x-series-to-previous and x-parallel-to-previous for adding new component to the previous in series or parallel. In addition, the active node remarked as active pnode, x-cast-to-asnode instruction and x-casnode-to-pasnode instruction are introduced to enhance the connection of branches, The detailed definitions of these instructions are shown as follow.

1) x-series-to-previous

This instruction expresses adding a component to the previous component in series. A new node should be created firstly named active snode. Then, disconnects one of the ports of the previous component and connects it with active snode. The new component will be added between the active snode and the other port of the previous component. In Fig. 2, x-series-to-previous instruction can add new component to the branch connect to ground.

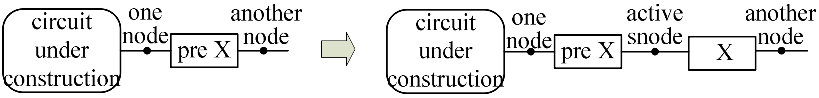


Fig. 2. Result of *x-series-to-previous* instruction

2) x-parallel-to-previous

New component is paralleled to the previous component by using this instruction. Shown in Fig. 3, a new component is paralleled to the previous x component. X-parallel-to-previous instruction could generate parallel component to the latest component in the circuit. It looks like that the x-parallel-to-previous instruction has the same functions as the x-cast-to-previous instruction. Actually, the x-cast-to-previous instruction could only generate parallel component to the latest component in the main branch. For example, the x-cast-to-previous instruction cannot add component parallel to the previous component connected to the ground. However, x-parallel-to-previous instruction can make up this shortage, which is the difference between x-parallel-to-previous instruction and x-cast-to-previous instruction.

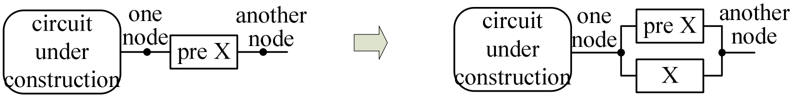


Fig. 3. Result of x-prallel-to-previous instruction

3) x-cast-to-asnode

This instruction connects x component between active pnode and active snode, shown in Fig. 4 (a). The connections between main branch and ground branches can be enhanced by x-cast-to-asnode.

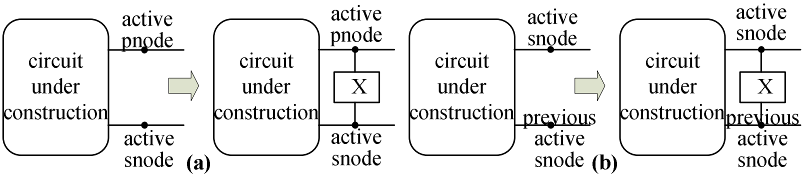


Fig. 4. Results of x-cast-to-asnode instruction and x-casnode-to-pasnode instruction

4) x-casnode-to-pasnode

X component is connected between current active snode and previous active snode, details is showed in Fig. 4 (b). This instruction is benefit for adding connections between different branches, which can produce novel circuit structures.

After extension, the cc-robot contains 9 instructions. The meanings of each instruction are summarized in Table 2.

Table 2. Extended instruction set of linear coding(x expresses component type)

Instruction	Incoming Node	Outgoing Node	Active Pnode	Active Snode
x-move-to-new	active pnode	newly-created node	newly-created node	unchanged
x-cast-to-previous	active pnode	previous node	unchanged	unchanged
x-cast-to-ground	active pnode	ground node	unchanged	unchanged
x-cast-to-input	active pnode	input node	unchanged	unchanged
x-cast-to-output	active pnode	output node	unchanged	unchanged
x-series-to-previous	one node of previous x	newly-created node	unchanged	newly-created node
x-parallel-to-previous	one node of previous x	another node of previous x	unchanged	unchanged
x-cast-to-asnode	active pnode	active snode	unchanged	unchanged
x-casnode-to-pasnode	active snode	previous active snode	unchanged	unchanged

4 Experimental Results and Analysis

In the experiments, we use analog filters as design example for making comparisons with other researches on coding methods for analog circuit design. We also make analysis on the performance of evolutionary design method of analog circuit based on the proposed coding method. For the specifications of the low-pass filter in the following experiments, the pass-band is (1,1K) Hz, and the stop-band is (2K,∞) Hz, the maximum attenuation in the pass-band is -3dB and the minimum attenuation in the stop-band is -60dB. We don't take the transition band into consideration.

4.1 Results of Evolutionary Design

In this part, in order to verify the ability of the extension of linear coding method, we use the extended linear coding method and genetic algorithm to design low-pass filters and then make a comparison between the results. In the experiments, the population size is 100. Maximum evolution generation is 300. Elite rate is 0.3. Crossover rate and mutation rate are respectively 0.8 and 0.1. For the component parameter, we adopt E-12 serial parameter values. Fig. 5 records the fitness curves of the ten random runs. It shows that the evolutionary algorithm based on the extended linear coding method is able to converge to design goal, even though the search space is extended at the same time. Fig. 6 shows the two typical circuits in the ten evolved circuits. We can see that the evolutionary design method based on the extended linear coding could not only generate circuits which could be generated by existing linear coding method (such as circuit in Fig. 6 (b)) but also generate novel structures (such as circuit in Fig. 6 (a)).

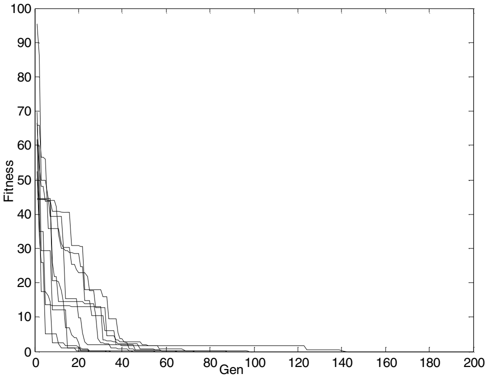


Fig. 5. The fitness curves of ten random runs

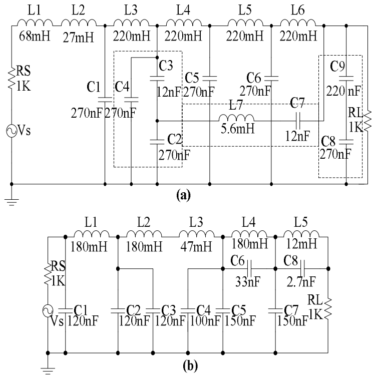


Fig. 6. The circuits generated by the extended linear coding method

4.2 Results of Fault-Tolerance Design

In this part, we are trying to check whether the extended linear coding can bring us some practical circuits. So, for single point short and single point disconnect two fault models, we employ the original and the extended linear coding methods to evolution low-pass fault-tolerant filter. We want to improve the average performance of the circuit under the single point of failure. In order to search fully in the coding space, we set 1000 as the maximum evolution generations. The rest of the evolutionary parameters are the same as the previous experiments.

The two filters in Fig. 7 are evolved for single point short and single point disconnect two fault models with the original linear encoding, and Fig. 8 shows the amplitude-frequency curves of the two circuits under single point failure. From the Fig. 8, we can see that there have three disaster points in circuit of Fig. 7 (a) (Fig. 8 (a) have three red lines), and one disaster point in circuit of Fig. 7 (b). Once the fault happened in such disaster points, circuits will lost their original function. The two filters in Fig. 9 are evolved for single point short and single point disconnect two fault models with the extended linear encoding, and Fig. 10 shows the amplitude-frequency curves of the two circuits under single point failure. From the Fig. 10, we can see that there is no disaster point in the two circuits. Failure occurs in any one element, circuits can still keep the original function only with performance decrease slightly. Comparing the circuits evolved with different coding methods, we can see that the extended linear encoding can bring new structures for evolution and generate some practical circuits for fault tolerance.

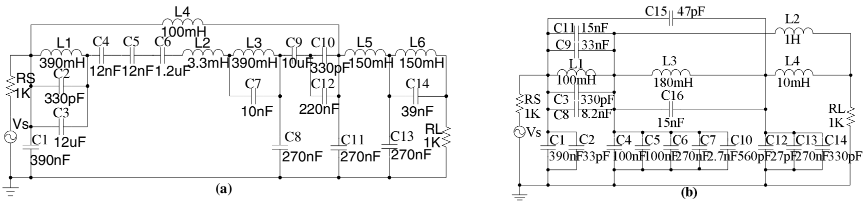
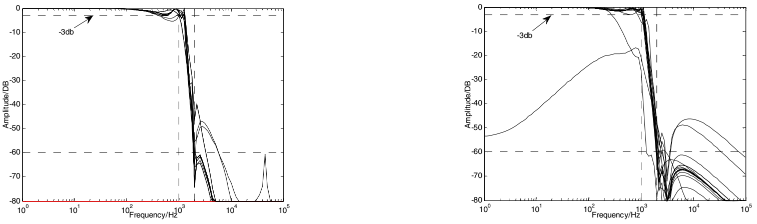


Fig. 7. The two low-pass filters which evolved by using the original linear coding method, (a) is evolved for single point short failure, (b) is evolved for single point disconnect failure



(a) (b)

Fig. 8. The output responses of circuits shown in the Fig.7 under single point failure

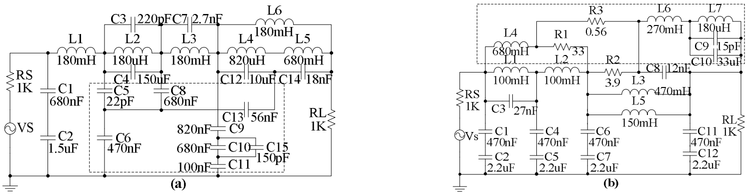
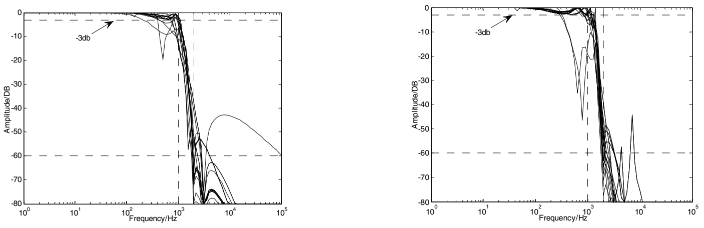


Fig. 9. The two low-pass filters which are evolved by using the extended linear coding method, (a) is evolved for single point short failure, (b) is evolved for single point disconnect failure



(a) (b)

Fig. 10. The output responses of circuits shown in the Fig.9 under single point failure

5 Conclusion

Linear coding method has been widely used for the evolutionary design of analog circuit, due to its relatively short length and convenience. Current linear coding method based on five connection ways limits the number of possible circuit structures. In this paper, we improve the linear coding method by expanding its instruction set. The experimental results show that the improved linear coding method can not only generate structures, which could be designed by the current linear coding method, but also generate analog circuits with novel structures. What's more, in the fault-tolerant experiments, it is shown that the proposed linear coding method could generate analog circuits with different fault-tolerant abilities. These analog circuits with novel structures provide new possible solutions for the fault-tolerant design of analog circuits.

Acknowledgments. This work is supported by National Nature Science Foundation of China under Grant 60975051 and 61273315.

References

1. Yao, X., Higuchi, T.: Promises and Challenges of Evolvable Hardware. IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews 29, 87–97 (1999)
2. Koza, J.R., Bennett III, F.H., Andre, D., et al.: Automated WYWIWYG Design of both the topology and component values of analog electrical circuits using genetic programming. In: Koza, J.R. (ed.) The first Annual Conference on Genetic Programming, Stanford University, CA, USA, pp. 123–131. MIT Press, Massachusetts (1996)
3. Koza, J.R., Bennett III, F.H., Andre, D., et al.: Automated synthesis of analog electrical circuits by mean of genetic programming. IEEE Transactions on Evolutionary Computation 1, 109–128 (1997)
4. Grimbleby, J.B.: Automatic Analogue Network Synthesis Using Genetic Algorithms. In: The First Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, Sheffield, UK, pp. 53–58 (1995)
5. Mattiussi, C., Floreano, D.: Analog Genetic Encoding for the Evolution of Circuits and Networks. IEEE Transactions on Evolutionary Computation 11, 596–607 (2007)
6. Lohn, J.D., Colombano, S.P.: Automated Analog Circuit Synthesis Using a Linear Representation. In: Sipper, M., Mange, D., Pérez-Uribe, A. (eds.) ICES 1998. LNCS, vol. 1478, pp. 125–133. Springer, Heidelberg (1998)
7. Lohn, J.D., Colombano, S.P.: A Circuit Representation Technique for Automated Circuit Design. IEEE Transactions on Evolutionary Computation 3(3), 205–219 (1999)
8. Vieira, P.F., Sa, L.B., Botelho, J.P.B., Mesquita, A.: Evolutionary synthesis of analog circuits using only MOS transistors. In: Proceedings of 2004 NASA/DoD Conference on Evolvable Hardware, pp. 38–45 (2004)
9. Liu, M., He, J.: An Evolutionary Negative-Correlation Framework for Robust Analog Circuit Design under Uncertain Faults. IEEE Transactions on Evolutionary Computation 99, 1–27 (2012)

The Design and Implementation of Motor Drive for an Electric Bicycle

Tze-Yee Ho[1], Mu-Song Chen[2], Wei-Chieh Chen[1], and Chih-Hao Chiang[1]

[1] Dept. of Electrical Engineering, Feng Chia University, 100 WenHwa Road, Seatwen, Taichung, Taiwan, R.O.C.
[2] Dept. of Electrical Engineering, Da-Yeh University, No.168 University Rd., Dacun, Changhua, Taiwan, R.O.C.
tyho@mail.fcu.edu.tw, chenms@mail.dyu.edu.tw,
alex94279@gmail.com, K869869@hotmail.com

Abstract. In recent years, the highly growth and development of world economy results in the natural resources being gradually run out and the environment further directly and indirectly being polluted more severe. Consequently, any kind of alternative energy resource have been developed, harvested and designed. An electric bicycle based on a blushless dc motor drive which has high efficiency, zero pollution, clean and convenient, is then designed and implemented in this paper. The hardware design based on a microcontroller is analyzed and discussed. The software programming is developed in MPLAB integrated development environment from the Microchip Technology Inc. Finally, a prototype of blushless dc motor drive for an electric bicycle is realized and demonstrated. The experimental results show the feasibility and fidelity of the complete designed system.

Keywords: electric bicycle, microcontroller, blushless dc motor drive.

1 Introduction

In order to avoid global warming from becoming more serious, many countries in the world agree to follow the Kyoto Protocol that was addressed in conference of the United Nations Climate Change Framework Convention held in Japan at December, 1997. Since then, lots of the alternative energy resources have been developed and harvested. So that the exhaust gas of the automobile is strictly required to meet the standard of protocol. The electric bicycle which does not produce the exhaust gas and generate the air pollution has become one of the most efficient and economic traffic vehicles in the metropolitan city [1]. Recent years, due to the highly development of power electronic devices and microcontrollers, the design and implementation of the electric bicycle have been studied in the academic field and widely applied to the industry as well. In order to realize the complete system, the basic configuration of an electric bicycle drive which consists of a controller, battery charger, battery and electric motor is firstly addressed in this paper. A controller is basically based on a microcontroller that controls the power from the battery to the electric motor and the

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 488–495, 2013.

auxiliary devices such as the brake, the signal lights as well. Since the battery is usually rechargeable and connected serially by two 12-volt batteries, the battery charger is required when the low battery signal occurs. There are commonly two types of battery available in the market. One is lead-acid battery and the other is NiMH battery. The former is inexpensive and heavy. The latter is light and possesses good performance, however, it costs high. Since the electric bicycles are a clean and zero pollution vehicles, thus have been gaining increasing attentions worldwide, especially in China, Taiwan, Europe, Japan, and the United States [2]. Therefore, in order to design and implement an electric bicycle, a brushless dc motor drive based on a microcontroller is presented in this paper. Both design criterion and requirements of the designed system meet the standard of electric-auxiliary bicycles CNS14126 in Taiwan [3].

The hardware of the designed system includes a microcontroller, power break circuit, voltage regulator, half bridge gate drive, current sensing circuit, and photo isolator circuit. The 120-degree conduction pulse width modulation (PWM) control method is employed to the blushless dc (BLDC) motor drive according to the rotor position from the measurements of Hall effect magnetic pole sensing. The PI control algorithm is applied to achieve the best performance of system response. The software programs are written in C language and programmed based on the MPLAB integrated development environment (IDE) tool by Microchip technology incorporated [4]. Finally, the complete system is successfully designed and implemented in this paper. The experimental results show the system realization and verify the system performance.

2 The System Hardware Structure of Electric Bicycle

The system hardware of an electric bicycle based on a BLDC motor drive is shown in Fig. 1. It consists of a dsPIC30F4011 microcontroller, protection circuit, optical coupling isolation, inverter, current sensor, encoder, and communication interface.

The microcontroller dsPIC 30F4011 manufactured by Microchip technology incorporate is the core controller of the electric bicycle. It is a 16-bit CPU with the capability of digital signal processing. Moreover, it supports many powerful modules such as built-in PWM module, addressable encoder interface module, and input capture module, these make the design friendly and thus shorten the development schedule. The three-phase bridge inverter comprises six power MOSFETs for switching. The photocoupler TLP250 manufactured by Toshiba semiconductor company, is used for electrical isolation between the microcontroller system and bus voltage. The motor currents are sensed through the current detection circuit. The magnet pole and rotor position are detected by the Hall effect sensor. In such a way, the speed and rotor position can be calculated and precisely controlled, subsequently. The 120-degree conduction of pulse width modulation technique for MOSFETs switching, is applied to drive the thee-phase inverter.

The equivalent circuit of a BLDC motor is shown in Fig. 2. The stator phase voltage equations (V_{an}, V_{bn}, V_{cn}) related to the stator phase currents (i_a, i_b, i_c) and back electromotive force (e_a, e_b, e_c) for a BLDC motor, can be expressed by (1) [4-5].

$$\begin{aligned}
V_{an} &= R_a i_a + L_{aa}\frac{di_a}{dt} + L_{ab}\frac{di_b}{dt} + L_{ac}\frac{di_c}{dt} + e_a \\
V_{bn} &= R_b i_b + L_{ba}\frac{di_a}{dt} + L_{bb}\frac{di_b}{dt} + L_{bc}\frac{di_c}{dt} + e_b \quad . \\
V_{cn} &= R_c i_c + L_{ca}\frac{di_a}{dt} + L_{cb}\frac{di_b}{dt} + L_{cc}\frac{di_c}{dt} + e_c
\end{aligned} \tag{1}$$

Where R_a, R_b, R_c, represent the phase resistance for each phase, L_{aa}, L_{bb}, L_{cc} represent the self inductance for each phase and L_{ab}, L_{bc}, L_{ca} represent the mutual inductance between either of two phases, e_a, e_b, e_c, represent the back EMF for each phase.

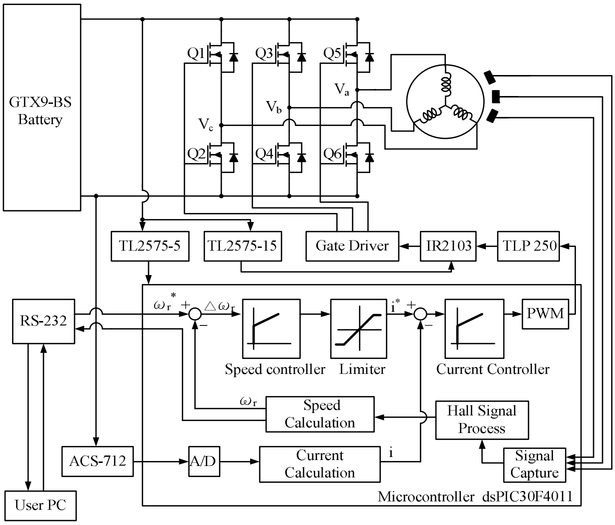


Fig. 1. The system hardware of electric bicycle

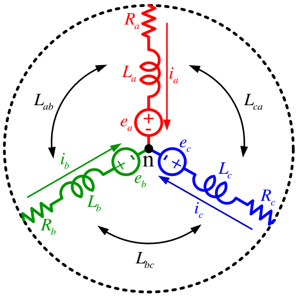


Fig. 2. The equivalent circuit of a BLDC motor

If a three-phase balanced system is considered, the stator voltage in (1) can be rearranged to

$$\begin{bmatrix} V_{an} \\ V_{bn} \\ V_{cn} \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} L_s & 0 & 0 \\ 0 & L_s & 0 \\ 0 & 0 & L_s \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} . \tag{2}$$

In steady state, the air gap power is expressed in terms of the electromagnetic torque and speed as

$$e_a i_a + e_b i_b + e_c i_c = T_e \omega_m . \tag{3}$$

Hence the electromagnetic torque can be represented as

$$T_e = \frac{e_a i_a + e_b i_b + e_c i_c}{\omega_m} . \tag{4}$$

Rearrange (4), the electromagnetic torque can be expressed by

$$J \frac{d\omega_m}{dt} + B\omega_m + T_L = T_e . \tag{5}$$

The load model can be expressed in terms of a moment of inertia, J, in kg-m^2/sec^2 with a viscous friction B, in N-m/rad/sec. The electromagnetic torque, T_e, in N-m then drives the load torque, T_L, in N-m as represented in (5) [6-7].

3 The System Software Development

The system software program is developed under MPLAB IDE software platform and written in C language. Most of the functions of electric bicycle are programmed in the microcontroller firmware which includes the circuit protection mechanism, PWM generation, motor currents calculation, rotor position and speed calculation and rotor pole position [7-8]. The flowchart of main program for microcontroller firmware is shown in Fig. 3. The initializations for I/O configuration, Timer 1, Timer 2, ADC and PWM settings are firstly processed in the main program.

Fig. 4 shows the PWM interrupt routine. The sensing current is firstly calculated and fed to the current controller. Since the PWM frequency is 20 kHz, the current controller is updated for every 50 us. After the calculation of current controller, the speed calculation is then performed by speed controller. The speed is constrained by the limiter for the operating speed range.

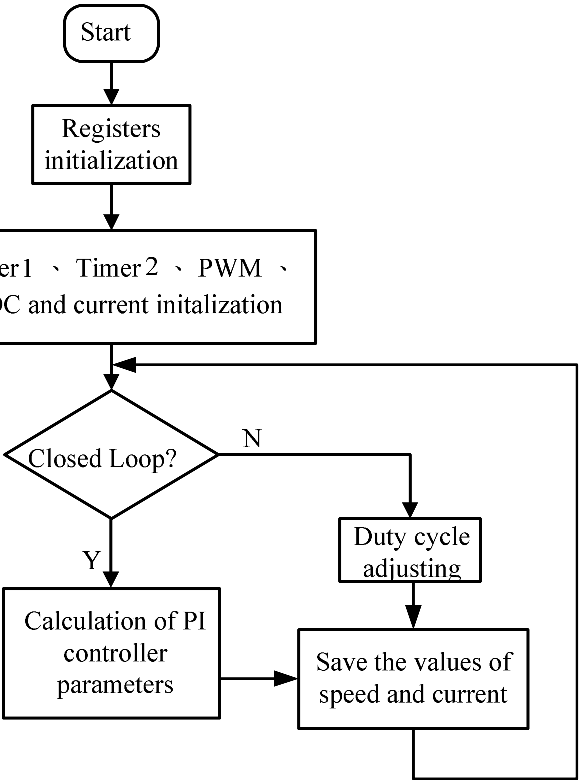

Start
Registers initialization
Timer1 、Timer2 、PWM 、 ADC and current initalization
Closed Loop?
N
Y
Duty cycle adjusting
Calculation of PI controller parameters
Save the values of speed and current

Fig. 3. The flowchart of main program

4 The Experimental Results

The prototype of an electric bicycle based on the dsPIC microcontroller with PI control, is tested under different load conditions which are fulfilled with the dynamometer. In order to test and verify the step response of different speed for the designed motor drive, different speed command is given for every 5 seconds. The step speed response profile is shown in Fig. 5. For the phase voltages of BLDC motor during the operation, Fig. 6 shows the phase voltages when 40 % duty cycle of PWM is applying to the switching MOSFETs, where ch1, ch2 and ch3 represent the phase A, phase B and phase C, respectively. Fig. 7 shows the step response of the speed from zero to 200 rpm under the load of 5 kg-cm. It can be observed that the motor drive can reach the command speed within the 1.5 seconds. The phase currents under the load of 5 kg-cm is shown in Fig.8, where ch1, ch2 and math represent phase currents A, B, and C, respectively.

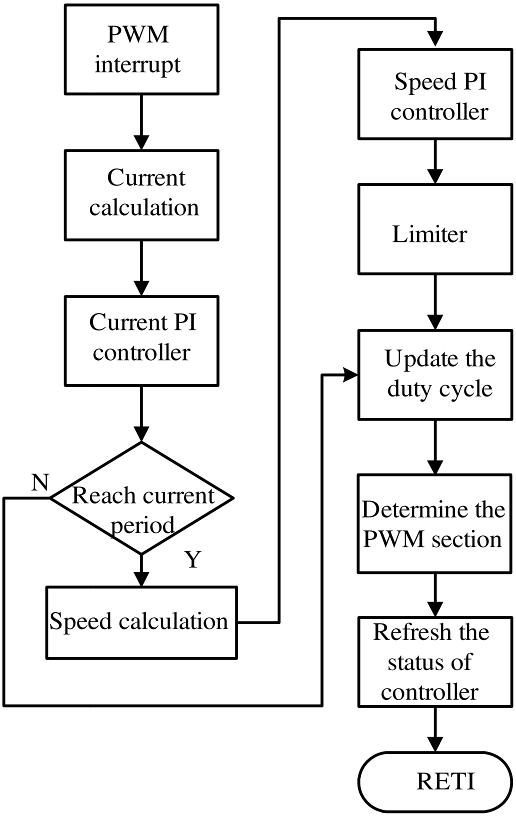


Fig. 4. The PWM interrupt routine

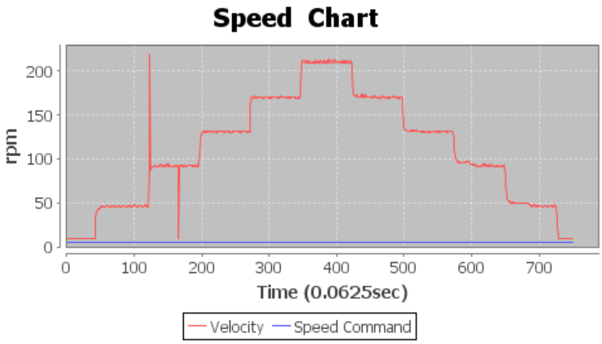


Fig. 5. The step speed response profile

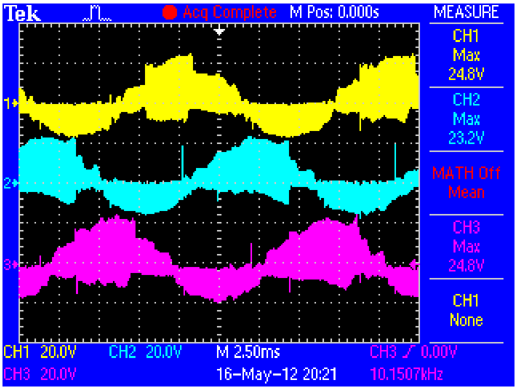


Fig. 6. The phase voltages

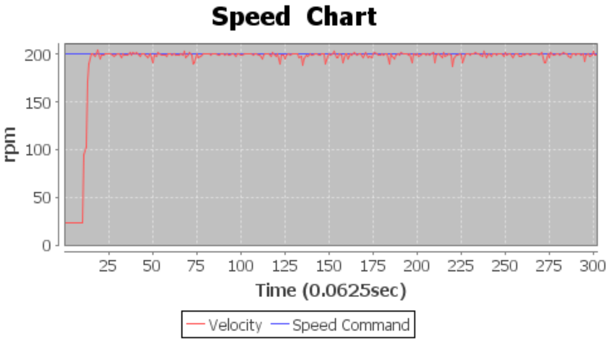


Fig. 7. The step response of speed under the load of 5 kg-cm

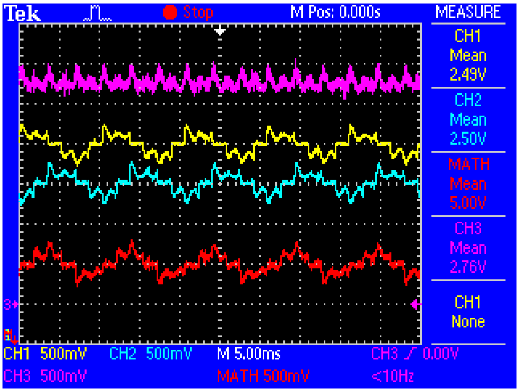


Fig. 8. The phase currents under the load of 5 kg-cm

5 Conclusions

The hardware structure including a microcontroller, protection circuit, optical coupling isolation, three-phase inverter, current sensor and communication interface is well designed. Further, the system software and firmware of microcontroller are programmed and described in detail. From the experimental results, it can be seen that the good performance of speed response profile verified the designed system over the operating speed range. Finally, a BLDC drive for an electric bicycle is realized and demonstrated in this paper. The experimental results show the feasibility and fidelity of the complete designed system.

References

1. Eichenberg, D.J., Kolacz, J.S., Tavernelli, P.E.: Baseline Testing of the Global E-Bike SX, NASA/TM -2001-210972, Glenn Research Center, Cleveland, Ohio (June 2001)
2. Muetze, A., Tan, Y.C.: Electric Bicycle – A Performance Evaluation. IEEE Industry Applications Magazine (July/August 2007)
3. Chinese National Standards CNS14126 (1998)
4. Ho, T.-Y., Chen, M.-S., Yang, L.-H., Lin, J.-S., Chen, P.-H.: The Design of a High Power Factor Brushless DC Motor Drive. International Journal of Advancements in Computing Technology (IJACT) 4(18), P141–P149 (2012) ISSN: 2005-8039
5. Krishnan, R.: Selection criteria for servo motor drives. IEEE Transactions on Industry Applications IA-23(2), 270–275 (1987)
6. Ziegler, J.G., Nichols, N.B.: Optimum settings for automatic controllers. Trans. Amer. Soc. Mech. Eng. 64, 759–768 (1942)
7. Hang, C.C., Astrom, K.J., Ho, W.K.: Refinements of the Ziegler-Nichols tuning formula. In: Proceeding IEE Pt. D, vol. 138, pp. 111–118 (1991)
8. Pillay, P., Krishnan, R.: Modeling, simulation, and analysis of permanent-magnet motor drives. IEEE Transactions on Industry Applications 25, 265–273 (1989)

The UML Diagram to VHDL Code Transformation Based on MDA Methodology

Chi-Pan Hwang[1] and Mu-Song Chen[2]

[1] National Changhua University of Education
[2] Da-Yeh University
cphwang@cc.ncue.edu.tw, chenms@mail.dyu.edu.tw

Abstract. The Model Driven Architecture (MDA) methodology requires several intelligent operation stages, such as the computation independent model transformation (CIMT), the platform independent model transformation (PIMT), and the platform specific model transformation (PSMT), to progressively transform an abstract model to a physical system. The special Unified Modeling Language (UML) or StarUML is the core tool of CIMT that models a digital system in a diagram paradigm. PIMT uses the Python language with *minidom* object to perform a series translation from UML diagram to VHSIC Hardware Description Language (VHDL) code. Finally, the PSMT imports an *os* object to Python for running a series of synthesis command script to get bit stream that is finally downloaded into FPGA device to complete the realization of the digital logic circuit.

Keywords: Model Driven Architecture, Unified Modeling Language, CIMT, PIMT, PSMT.

1 Introduction

The design of a digital system starts from a transformation of specification to a set of truth tables or state diagrams, and then transforms to a set of Boolean algebraic expressions by another set of theoretic operations that are manipulated by experienced engineers [1]. If the implementation is realized in terms of VHDL, the specifications have to be manually programmed to describe the system behaviors. The VHDL statements must be also compiled into a bit stream and then downloaded into a programmable logic device with special designated synthesis tools to complete the system realization. However, interdisciplinary engineers could not be skilled in programming, especially when they design the hardware circuit using the VHDL. The MDA methodology permits better consistency between models and programming codes and increases productivity between automated mapping of models to implementations [2][3]. In the digital of logic circuit, the MDA methodology is based on a series of transformation stages supported by experienced knowledge to gradually convert an abstract source model to the physical target circuits.

The rest of the paper is organized as follows. In the section 2, we briefly introduce the MDA methodology. Section 3 describes the system design and its implementation. Finally, section 4 concludes the results of the system development.

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 496–503, 2013.

2 MDA Methodology for Digital Logic Circuit Realization

The MDA is based on the perspectives with *computation independent*, *platform independent*, and *platform specific* to develop the abstract models about *qualitative*, *numeric*, and *physical* [4]. The operational stages of CIMT, PIMT, and PSMT are shown in Fig. 1 that can gradually and progressively transform an abstract model to the desired system. The function of CIMT is based on the unified modeling language (UML) to build the composite and statechart diagrams as models corresponding to the digital system specifications.

The conversion of an UML diagram to VHDL code is undertaken by the PIMT that has been supported by the design expertise of digital circuits. The PSMT integrates the compiling and synthesis tools for specific FPGA devices to generate the configuration bit stream that would be downloaded to the FPGA chip for completing the digital system [5].

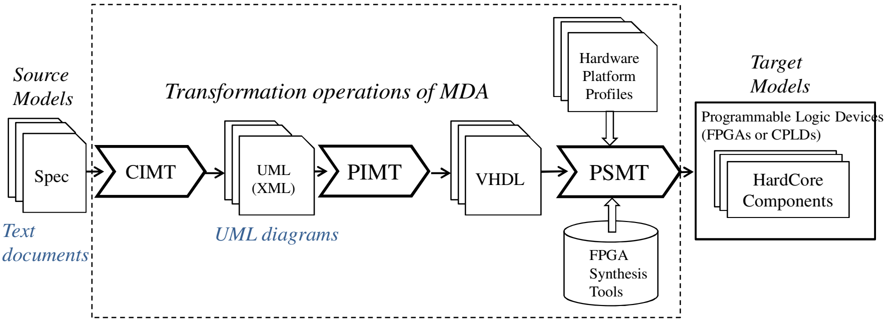


Fig. 1. Model transformation operations of the MDA

3 System Design and Implementation

The starUML provides GUI and model management for the CIMT to construct the composite and statechart diagrams as digital circuit model. The PIMT and PSMT are realized by Python language supported by *minidom* and *os* objects that are applied to develop the xml parser in the PIMT to transform the UML diagrams to VHDL codes, and execute synthesis command script to generate the configuration bit stream to download to the FPGA device for completing the digital circuit realization.

3.1 CIMT Design and Implementation

The digital circuits include the combinational and sequential logic circuits that will be modeled into composite diagram and state diagram by the starUML. Fig. 2 illustrates a hierarchical composite diagram for a 4-bit ripple adder by the CIMT. Fig. 3 also shows a XPD document piece of composite diagram in Fig. 2. It has included the model elements of *class instance*, *port*, and *interface* corresponding to *function block*, *I/O pin*, and *connection* of the 4-bit ripple adder. The class instance name is bipartite connected by an underscore. The former part denotes a class name and the next part indicates an instance of the class. In Fig. 3, **XOR2_1** is the value of ATTR name field in an OBJ element. The **XOR2** is a class name that denotes a 2-input exclusive-OR gate. The accompanied number after underscore indicates an instance of XOR2 class. The name of Port and Interface consists of a label and a data type that have been enclosed in the square brackets. The interface **A**[**bool**] in Fig. 4 names a connection port **A** and defines its data type to **bool**. The *mode* of Port and Interface is determined by the connections of *Realization* and *Dependency*. The interface **A**[**bool**] is connected to **A**[**bool**] port of **XOR2_1** by *Dependency* described in in Fig. 3 that implies the Interface **A**[**bool**] to be an external input port and the **A**[**bool**] port of **XOR2_1** is an input port. Similarly, if an Interface is connected by *Realization* that represents it to be an external output port, and the connected port of a class instance to be an output port. Otherwise, if an Interface is simultaneously connected by *Realization* and *Dependency* then it is defined to be an internal connection.

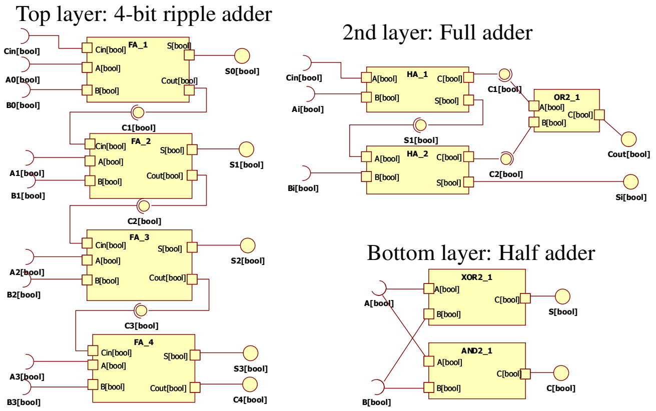
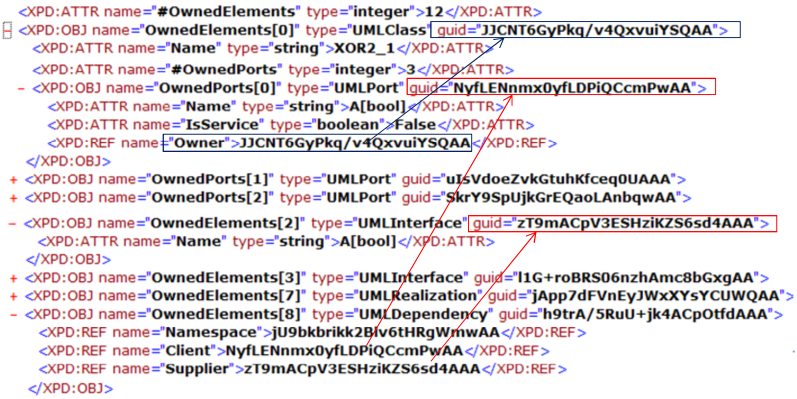


Fig. 2. The UML composite diagrams modeling the 4-bit ripple adder



```
 <XPD:ATTR name="#OwnedElements" type="integer">12</XPD:ATTR>
 <XPD:OBJ name="OwnedElements[0]" type="UMLClass" guid="JJCNT6GyPkq/v4QxvuiYSQAA">
   <XPD:ATTR name="Name" type="string">XOR2_1</XPD:ATTR>
   <XPD:ATTR name="#OwnedPorts" type="integer">3</XPD:ATTR>
 - <XPD:OBJ name="OwnedPorts[0]" type="UMLPort" guid="NyfLENnmx0yfLDPiQCcmPwAA">
     <XPD:ATTR name="Name" type="string">A[bool]</XPD:ATTR>
     <XPD:ATTR name="IsService" type="boolean">False</XPD:ATTR>
     <XPD:REF name="Owner">JJCNT6GyPkq/v4QxvuiYSQAA</XPD:REF>
   </XPD:OBJ>
 + <XPD:OBJ name="OwnedPorts[1]" type="UMLPort" guid="uIsVdoeZvkGtuhKfceq0UAAA">
 + <XPD:OBJ name="OwnedPorts[2]" type="UMLPort" guid="SkrY9SpUjkGrEQaoLAnbqwAA">
 - <XPD:OBJ name="OwnedElements[2]" type="UMLInterface" guid="zT9mACpV3ESHziKZS6sd4AAA">
    <XPD:ATTR name="Name" type="string">A[bool]</XPD:ATTR>
   </XPD:OBJ>
 + <XPD:OBJ name="OwnedElements[3]" type="UMLInterface" guid="l1G+roBRS06nzhAmc8bGxgAA">
 + <XPD:OBJ name="OwnedElements[7]" type="UMLRealization" guid="jApp7dFVnEyJWxXYsYCUWQAA">
 - <XPD:OBJ name="OwnedElements[8]" type="UMLDependency" guid="h9trA/5RuU+jk4ACpOtfdAAA">
    <XPD:REF name="Namespace">jU9bkbrikk2Blv6tHRgWmwAA</XPD:REF>
    <XPD:REF name="Client">NyfLENnmx0yfLDPiQCcmPwAA</XPD:REF>
    <XPD:REF name="Supplier">zT9mACpV3ESHziKZS6sd4AAA</XPD:REF>
   </XPD:OBJ>
```

Fig. 3. The XPD file fragment of composite diagram in Fig. 2.

Fig. 4 exemplifies a composite and statechart diagram that is generated by the CIMT to model a sequential logic circuit. The composite diagram depicts the profile of sequential logic circuit, and names the function class and I/O ports. The statechart diagram illustrates the discrimination of "110" bit sequence from input port *x*, and generate the result to output port *y*. The **State0** is the initial state. It transits to **State1** when *x*=1, otherwise stays in **State0** and output *y*=0. In state **State1**, it transits back to **State0** when input *x*=0. Otherwise, transits to **State2** to represent "11" haven been encountered. In the state **State2**, it transits to **State0** when input *x*=0, and output *y* is 1 because the "110" has been discriminated. Otherwise, it still stays in **State2** when input *x*=1 and output *y*=0.

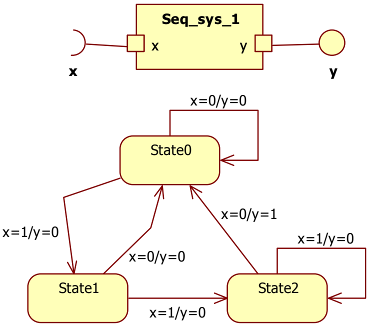


Fig. 4. The UML statechart diagram for "110" discrimination

3.2 PIMT Design and Implementation

The PIMT consists of a model parameter filter and a hardware description language translator that are implemented by Python language. The combinational logic circuit translation must iteratively filter out the model parameters from composite diagram document to incrementally construct a set of lists for conducting VHDL code generation. The translation is started to open the composite diagram document, then calls the *parsestring*(XPD_document) in *minidom* object to generate a **dom** object that is a tree structure in main memory. Furthermore, there six pass operations on the **dom** object are implemented by Python to perform the combinational logic circuit translation.

The first pass on dom object gets name and guid of UMLClass, Interface, Dependency, and Realization to build corresponding lists. In the second pass, the Ports associated with an UMLClass are extracted and stored in its corresponding lists. The third pass determines the *mode* and *type* of Ports and Interfaces that refers to the generated Port, Dependency, and Realization lists, based on the rules as mentioned in former section. The VHDL code consists of entity and architecture parts that are incrementally generated from fourth to six operational pass. The type and mode of Ports and Interfaces are determined in third pass that are used to generate entity part of VHDL code in the fourth pass. Referring to Fig. 2, **Cin**[**bool**], **Ai**[**bool**], and **Bi**[**bool**] are determined to the external input port list, and **Cout**[**bool**] and **Si**[**bool**] are defined into the external output port list in the third pass. Fig. 5 illustrates the generated VHDL entity part.

```
entity FADDER is
   port (
      Cin : in std_logic;
      Ai : in std_logic;
      Bi : in std_logic;
      Cout : out std_logic;
      Si : out std_logic
   );
end entity;
```

Fig. 5. VHDL entity statement of the Full adder

The architecture of VHDL is divided into declaration and concurrent statement sections. The declaration of internal connections and components are generated in the fifth pass that refers to the internal connection and UMLClass lists. Fig. 6 reveals the translation results of declaration section for Full adder. The concurrent statement section is generated in the six pass. Each concurrent statement has *label*, *class instance name*, and *port map* fields. The label field is the class instance name of the UMLClass. The class name is the former part of label before the underscore. The port map part is a set of name associations to represent the interconnections between ports of function blocks in a combinational logic circuit. Fig. 7 shows the translated concurrent statements.

```
signal C1 : std_logic;
signal S1 : std_logic;
signal C2 : std_logic;
component HA is
   port (
      A : in std_logic;
      B : in std_logic;
      C : out std_logic;
      S : out std_logic
   );
end component;
component OR2 is
   port (
      A : in std_logic;
      B : in std_logic;
      C : out std_logic
   );
end component;
```

Fig. 6. Declaration section of full adder in the architecture part

```
HA_1: HA port map(A=>Cin,B=>Ai,C=>C1,S=>S1);
HA_2: HA port map(A=>S1,B=>Bi,C=>C2,S=>Si);
OR2_1: OR2 por map(A=>C1,B=>C2,C=>Cout);
```

Fig. 7. The Concurrent statement section of full adder in the architecture part

The sequential logic circuit is composed of composite and statechart diagrams in the UML. The composite diagram is only used to generate the entity part of VHDL code for the sequential logic circuit. The statechart diagram is realized to VHDL code in the architecture part. The entity part of composite diagram illustrates in Fig. 8, where **clk** and **reset** signals are involved automatically.

```
entity Seq_sys_1 is
   port (
      clk : in std_logic;
      reset : in std_logic;
      x : in std_logic;
      y : out std_logic
   );
end entity;
```

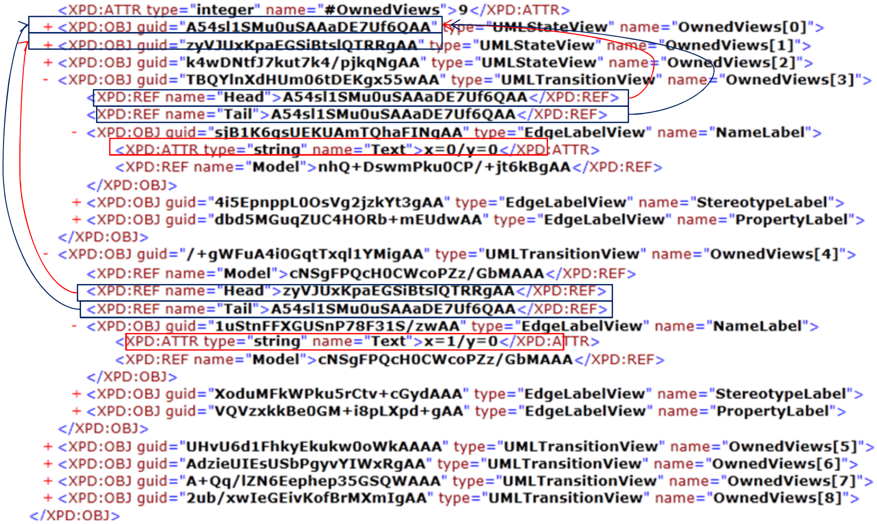
Fig. 8. The entity part of the composite diagram in Fig. 4

The architecture part translation is manipulated by two passes. The first pass generates the state list and declaration section in architecture, based on the UMLStateView in UML file of statechart diagram. Referring to Fig. 9, the *guid* and label of the state are obtained from UML state view where *guid* is extracted from Model of REF tag. The label is retrieved from name of ATTR tag in NameLabel OBJ that is a descendent of NameComponent OBJ involved in state view. Fig. 10 shows the translated VHDL code of declaration section.

In the second pass, a process statement of VHDL is generated to realize the behavior model of sequential logic circuit. For each state in the state list, it scans the Tail of REF tag in every UMLTransitionView. If the *guid* is same as the current state, then stores it temporarily to a transition list. At the same time, an if-elsif statement is generated that condition part comes from the former part before slash of name ATTR element in the NameLabel OBJ involved in UMLTransitionView OBJ. Referring to Fig. 10 and based on State0 in state list, the guid of State0 is found in two transition views that are **OwnedViews**[3] and **OwnedViews**[4]. Their labels are *x*=0/*y*=0 and *x*=1/*y*=0 that are extracted from name ATTR element in the NameLabel OBJ. It generates a piece of VHDL code for **State0** in Fig. 11.

3.3 PSMT Design and Implementation

The PSMT is dependent on a selected programmable logic device. It executes a series of parsing and synthesis commands for generating downloadable bit stream file that is in turn downloaded to the programmable device for completing the realization of a digital application system. In this paper, Xilinx FPAG devices have been chosen to a target to implement the digital logic circuit. An *os* object is imported into Python for running a series of synthesis processes denoted in a command script as shown in Fig. 12 to get bit stream file.



```
   <XPD:ATTR type="integer" name="#OwnedViews">9</XPD:ATTR>
 + <XPD:OBJ guid="A54sl1SMu0uSAAaDE7Uf6QAA" type="UMLStateView" name="OwnedViews[0]">
 + <XPD:OBJ guid="zyVJUxKpaEGSiBtslQTRRgAA" type="UMLStateView" name="OwnedViews[1]">
 + <XPD:OBJ guid="k4wDNtfJ7kut7k4/pjkqNgAA" type="UMLStateView" name="OwnedViews[2]">
 - <XPD:OBJ guid="TBQYlnXdHUm06tDEKgx55wAA" type="UMLTransitionView" name="OwnedViews[3]">
     <XPD:REF name="Head">A54sl1SMu0uSAAaDE7Uf6QAA</XPD:REF>
     <XPD:REF name="Tail">A54sl1SMu0uSAAaDE7Uf6QAA</XPD:REF>
   - <XPD:OBJ guid="sjB1K6qsUEKUAmTQhaFINgAA" type="EdgeLabelView" name="NameLabel">
       <XPD:ATTR type="string" name="Text">x=0/y=0</XPD:ATTR>
       <XPD:REF name="Model">nhQ+DswmPku0CP/+jt6kBgAA</XPD:REF>
     </XPD:OBJ>
   + <XPD:OBJ guid="4i5EpnppL0OsVg2jzkYt3gAA" type="EdgeLabelView" name="StereotypeLabel">
   + <XPD:OBJ guid="dbd5MGuqZUC4HORb+mEUdwAA" type="EdgeLabelView" name="PropertyLabel">
   </XPD:OBJ>
 - <XPD:OBJ guid="/+gWFuA4i0GqtTxql1YMigAA" type="UMLTransitionView" name="OwnedViews[4]">
     <XPD:REF name="Model">cNSgFPQcH0CWcoPZz/GbMAAA</XPD:REF>
     <XPD:REF name="Head">zyVJUxKpaEGSiBtslQTRRgAA</XPD:REF>
     <XPD:REF name="Tail">A54sl1SMu0uSAAaDE7Uf6QAA</XPD:REF>
   - <XPD:OBJ guid="1uStnFFXGUSnP78F31S/zwAA" type="EdgeLabelView" name="NameLabel">
       <XPD:ATTR type="string" name="Text">x=1/y=0</XPD:ATTR>
       <XPD:REF name="Model">cNSgFPQcH0CWcoPZz/GbMAAA</XPD:REF>
     </XPD:OBJ>
   + <XPD:OBJ guid="XoduMFkWPku5rCtv+cGydAAA" type="EdgeLabelView" name="StereotypeLabel">
   + <XPD:OBJ guid="VQVzxkkBe0GM+i8pLXpd+gAA" type="EdgeLabelView" name="PropertyLabel">
   </XPD:OBJ>
 + <XPD:OBJ guid="UHvU6d1FhkyEkukw0oWkAAAA" type="UMLTransitionView" name="OwnedViews[5]">
 + <XPD:OBJ guid="AdzieUIEsUSbPgyvYIWxRgAA" type="UMLTransitionView" name="OwnedViews[6]">
 + <XPD:OBJ guid="A+Qq/lZN6Eephep35GSQWAAA" type="UMLTransitionView" name="OwnedViews[7]">
 + <XPD:OBJ guid="2ub/xwIeGEivKofBrMXmIgAA" type="UMLTransitionView" name="OwnedViews[8]">
</XPD:OBJ>
```

Fig. 9. The UML file fragment of a statechart diagram

```
type state_type is (State0, State1, State2);
signal state : state_type;
```

Fig. 10. The declaration section of architecture part for statechart diagram in Fig. 6

```
case state is
     when State0 =>
        if x='0' then
            y <= '0';
            state <= State0;
         elsif x='1' then
            y <= '0';
            state <= State1;
         end if;
```

Fig. 11. A state transition code in architecture part for statechart diagram in Fig. 6

```
import os
def vhdlcompile():
 os.chdir("d:\\ISE_Projects\\xsttest")
 os.system('c:\\Xilinx\\13.4\\ISE_DS\\ISE\\bin\\nt\\xst ...)
 os.system('c:\\Xilinx\\13.4\\ISE_DS\\ISE\\bin\\nt\\ngdbuild ...)
 os.system('c:\\Xilinx\\13.4\\ISE_DS\\ISE\\bin\\nt\\map ...)
 os.system('c:\\Xilinx\\13.4\\ISE_DS\\ISE\\bin\\nt\\par ...)
```

Fig. 12. Example code of PSMT

4 Conclusions

The digital system is the core of many electronic application systems. The FPGA devices are also frequently applied to construct the parallel circuits for satisfying the requirements of massively computing applications. The MDA methodology has devised intelligent interactions between CIMT, PIMT, and PSMT operational stages to gradually and progressively transform an abstract system model to the physical system. The StarUML is adopted to perform the functions of the CIMT that effectively captures the abstract model parameters of a digital system in a diagram paradigm. The PIMT performs a series of translations from the UML to the VHDL that uses the Python language with *minidom* object. It parses the XPD documents from starUML to transform to VHDL code incrementally. An *os* object is imported to Python in the PSMT stage for running a series of synthesis command script to get bit stream that is finally downloaded into FPGA device to complete the realization of digital logic circuits. The developed software tools have been integrated into the MDA operational stages that have been proved to effectively shorten the gap between specification and physical system and the turnaround time of the application system development.

Acknowledgments. This work is supported by National Science Council (R.O.C) under Grant NSC98-2220-E-018-001 and NSC101-2221-E-018-015.

References

1. Roth, C.H.: Fundamentals of Logic Design. Thomson, Singapore (2004)
2. Soley, R.: OMG Staff Strategy Group: Model Driven Architecture. White Paper Draft 3.2, Object Management Group (2000)
3. Mellor, S.J., Scott, K., Uhl, A., Weise, D.: The IT-Architecture Professionals–Model Driven Architecture. Resource Sharing, http://www.lcc.uma.es
4. Czarnecki, K., Helsen, S.: Classification of Model Transformation Approaches. In: Workshop on Generative Techniques in the Context of Model-Driven Architecture (2003)
5. Rieder, M., Steiner, R., Berthouzoz, C., Corthay, F., Sterren, T.: Synthesized UML, a practical approach to map UML to VHDL. In: Guelfi, N., Savidis, A. (eds.) RISE 2005. LNCS, vol. 3943, pp. 203–217. Springer, Heidelberg (2006)

Generating Mask from the Structural Layer of Micro Device

Zheng Liu

School of Mechatronic Engineering, Xi'an Technological University, Xi'an China
zheng.liumail@gmail.com

Abstract. Traditional design flow is not efficient enough for the complex surface micromachined device because of its lack of perceptual intuition. The feature technology was introduced into micro device area to provide the comfortable design environment. The problem solving of generating mask from the 3D model of micro device plays the key role to implement the advanced design way, which emphasized on the structural design method instead of beginning with the issues of fabricating processes. In this paper, the algorithm to evaluate the data of mask based on the 3D model of surface micromachined device is presented. With respect to the etching process, the etched solids were built up by particular operations of the layer models to indicate the etched parts. After that, the mask is derived on the basis of the etched solids.

Keywords: Mask generation, Etched solid, Solid element operation, Surface micromachining.

1 Introduction

Currently, the popular design tools of micro device begin with the processes design, which is the bottom-up flow [1-3]. The characteristic of the design way is summarized as mask-to-shape-to-verify [4]. For surface micro machined device, this mode has been used for simple structure for many years, but not suitable for the increasingly complex structure any more. Although the processes of the surface fabricating technology are similar with the integrated circuit in some respect, the design tools evolving from the integrated circuit area are not efficient for the design work of micro device because the model generally consists of several 3D layers sometime with cantilever or intersection structure to build up. For example, as the multiple layer processes, the MUMPs (Multi-User MEMS Processes) is adopted to fabricate the micro devices [5,6]. In this case, considering all the details of the masks in advance is obviously not intuitive for the designer. Therefore, the more reasonable way is construct the 3D model with components like features technology [7-9]. Furthermore, with this method, the relationship between the 3D structure and the system level modeling is more likely to build up [10-12]. Among the challenges faced to implement the advanced designing way, the mask synthesis plays the key role to complete the cycle of function-to-shape-to-mask [13].

To sum up, there have been many efforts to make the design work of micro device more efficient. The advanced design flow can be summarized as function-to-shape-to-mask, which is inverse to the traditional way to a great extent. Because of the different

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 504–510, 2013.

flow, many challenges emerged. The mask synthesis is the crucial problem to solve, so far as the 3D modeling method is adopted. Especially for surface micromachining, the multiple layers make things more tricky. In this paper, the method to generate mask from the solid model of surface micromachined device is presented, which is the key link to implement the top-down design flow.

2 The Description of Commonly Used Variables and Operations

For ease of reading, the description of commonly used variables and operations in this paper is as follows:

1) Boolean operation with material attributes

Boolean operation of Subtraction for solids is abbreviate to BS (solid A, solid B). Boolean operation of Intersection for solids is abbreviate to BI (solid A, solid B). Boolean operation of Union for solids is abbreviate to BU (solid A, solid B). These operations are all involve the assignment of material.

2) The etched solid

The etched solid set of the ith structural layer is abbreviate to $Ess\left(L_i\right)$. The etched solid set of the sacrificial layer close under it is abbreviate to $Ess\left(Sac_i\right)$. The etching features are represented as $Etc\left(Str\right)_{i,m}$ and $Etc\left(Sac\right)_{i,m}$ respectively.

3) Cantilever structure

Cantilever face is a kind of face of the cantilever structure, whose normal vector is vertically downward and there is no other structure of lower layers to support it on the whole face (see Figure 1). Because the structural layers are deposited on the lower layers in surface micromachining, the cantilever structure indicates that the material under it is removed. In fact, that is the sacrificial layer. Cantilever face is used to judge the existence of cantilever structure and then the existence of sacrificial layer. The height of the cantilever face is represented with the parameter “ h ”.

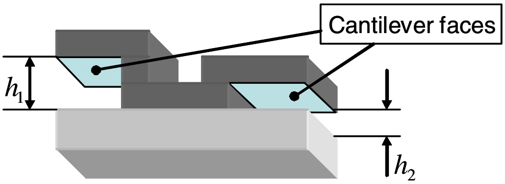


Fig. 1. Cantilever faces illustration

The etched solid set has a close relationship with the etching feature. Hence, there is not a clear separation between the deduction of Ess and the etching feature of the process model. To simplify the question, this paper lays special emphasis on the deduction process concerning with the mask set information.

The process model of the ith layer involves the process information of the ith structural layer and the corresponding sacrificial layer (if existing). For a certain layer, the existence of the sacrificial layer is judged by the cantilever structure firstly. In surface micromachining, the space taken by the material etched will be filled with the deposited part of the upper layers. It is the basis for the geometric model deduction with the Boolean operation. However, for the derivation of the sacrificial layer, not all the depositions are thick enough to fill these spaces. The solution method is to construct a temporary solid that depends on the upper structural layer and is thicker than the deposition model of the sacrificial layer. In this situation, the temporary solid is the real second parameter of the operation and is not displayed in the process model. Only to illustrate it simply, we will not restate in the later sections.

3 The Algorithm to Evaluate the Etched Solid

The algorithm to derive the model of etched solid is as follows:

1) The deposition model of the ith sacrificial layer $Dep(Sac)_i$

The max "h" in the cantilever structure is got as the thickness of the deposition. It is represented as $h_{\max}(Sac_i)$. A protrusion operation is executed with the parameters of $h_{\max}(Sac_i)$ and $USML$ of M_{i-1} to get $Dep(Sac)_i$. It is represented as $Dep(Sac)_i$ $= UEO\ (USML(M_{i-1}), h_{\max}(Sac_i))$.

2) The model remained after the etching of the ith sacrificial layer Sac_i

The deposition model of the sacrificial layer executes Boolean operation of Subtraction with the model of the ith structural layer to get Sac_i. It is represented as $Sac_i = BS(Dep(Sac)_i, L_i)$.

3) The etched solid set of the ith sacrificial layer $Ess(Sac_i)$

The deposition model of the sacrificial layer executes Boolean operation of Intersection with the model of the ith structural layer to get $Ess(Sac_i)$. It is represented as $Ess(Sac_i) = BI\ (Dep(Sac)_i, L_i)$.

4) The deposition model of the ith structural layer Dep_i

Firstly, the etching feature of the kth sacrificial layer (k<i) is revised if there is an intersection relation between L_i and Sac_k. Then, the distance of HFP in the design features is calculated. The max value ($h_{\max}(L_i)$) is as the parameter of thickness for deposition. Finally, a protrusion operation is executed with the parameters of $h_{\max}(L_i)$ and $USML$ of the combined solid of M_{i-1} and Sac_i to get $Dep(Sac)_i$. It is represented as $Dep_i = UEO\ (USML(BU(M_{i-1}, Sac_i)),\ h_{\max}(L_i))$.

5) The etched solid set of the ith structural layer $Ess(L_i)$

The deposition model of the structural layer executes Boolean operation of Subtraction with L_i to get $Ess(L_i)$. It is represented as $Ess(L_i) = BS\,(Dep_i, L_i)$. $Ess(L_i)$ is the set of the etched solid { $Es_{i,1}, Es_{i,2} \ldots\ldots Es_{i,n}$ }.

4 Derivation of the Mask Set

The geometric information of the mask set includes "vertex", "edge" and "loop", etc. The shape of the mask is constructed with these geometric elements. Moreover, it is the basis of the CIF file. The deduction of the mask is a process of determining these elements. It includes four steps mainly:

1) The division of the etched solid:

All the etched solids with same height are treated as one time etching process except the "step" structure as shown in Figure 2. The judgment of the "step" structure is dependent on whether there is more than one face ($Face_i$) whose normal vector is vertically downward in the etched solid and the distances btween the horizontal faces pair $HFP(Face_i)$ are different. The "step" structure means that there is more than one time etching process. In this case, the solid is separated into several parts (P, Q) with corresponding height h_1 , h_2 , etc. P and Q are treated respectively as individual etched solid to derive mask set together with other etched solids. In addition, there is a proper order to etch them according to their vertical locations. The upper hole is always manufactured in advance. Actually, the etched solid A is just a normal instance for illustration. Other instance is regarded as the combination of A (e.g., B is composed of two A).

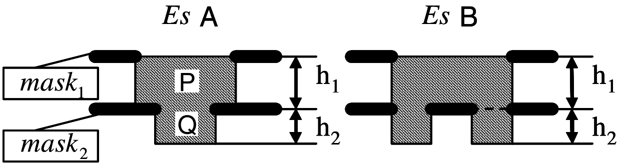


Fig. 2. The "step" structure of the etched solid

2) The grouping of the etched solids:

The etched solids are divided into groups according to the thickness. One group indicates one time etching process with etching feature and mask accordingly. After grouping, the number of the etching features and the masks of this layer are confirmed.

3) The extraction of the "loop" information:

The "loop" of top face of the etched solid is picked up. Figure 3 illustrates this process. If there is more than one top face in the etched solid, a trimming and joining operation for the "half edge" are needed with the projection of these faces on the horizontal plane.

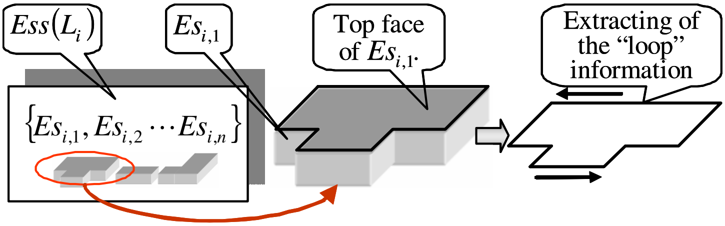

$Ess(L_i)$
$Es_{i,1}$
Top face of $Es_{i,1}$.
Extracting of the "loop" information
$\{Es_{i,1}, Es_{i,2} \cdots Es_{i,n}\}$

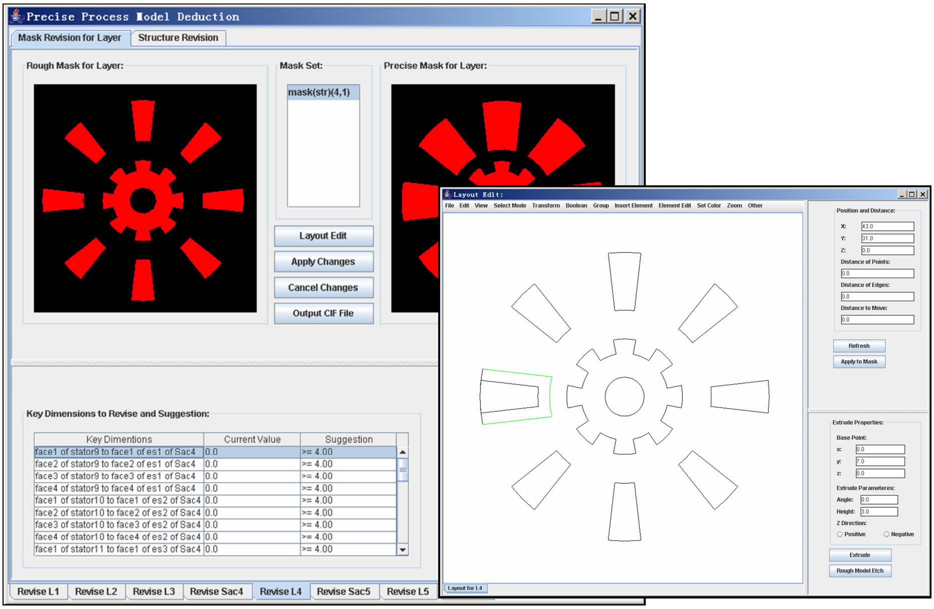
Fig. 3. The extraction of the "loop" information

4) The "Inverse" operation of the "loops":

Based on L_i and the information of the "loops" from Es, mask information is derived by the "inverse operation". Firstly, the "start vertex" and "end vertex" of each "half edge" in these loops are exchanged to make the "outer loops" into the "inner loops". Then, the "outer loops" are obtained by means of projecting the outer side faces of L_i on the horizontal plane. Finally, a trimming and joining operation are needed if the "half edges" have an overlapping relation on the horizontal plane.

5 Implementation

To illustrate the implementation of the method, the 4th layer the micro motor is used as the instance, which is designed according to MUMPs technology. The mask generated is shown in Figure 4.



Key Dimensions to Revise and Suggestion:

Key Dimentions	Current Value	Suggestion
face1 of stator9 to face1 of es1 of Sac4	0.0	>= 4.00
face2 of stator9 to face2 of es1 of Sac4	0.0	>= 4.00
face3 of stator9 to face3 of es1 of Sac4	0.0	>= 4.00
face4 of stator9 to face4 of es1 of Sac4	0.0	>= 4.00
face1 of stator10 to face1 of es2 of Sac4	0.0	>= 4.00
face2 of stator10 to face2 of es2 of Sac4	0.0	>= 4.00
face3 of stator10 to face3 of es2 of Sac4	0.0	>= 4.00
face4 of stator10 to face4 of es2 of Sac4	0.0	>= 4.00
face1 of stator11 to face1 of es3 of Sac4	0.0	>= 4.00

Position and Distance:

X: 43.0
Y: 31.0
Z: 0.0
Distance of Points: 0.0
Distance of Edges: 0.0
Distance to Move: 0.0

Extrude Properties:

Base Point:
x: 0.0
y: 7.0
z: 0.0

Extrude Parameteres:
Angle: 0.0
Height: 3.0
Z Direction: ○ Positive ○ Negative

Fig. 4. The mask generation of the 4th layer of micro motor

6 Conclusion

This paper presents the synthesis on how to evaluate the data of mask from the layered model of surface micromachined device. It solves the etching information deriving problem without the mask design in advance. The etched solid plays a key role in building the relationship between the structural layer model and the mask set. With respect to the overall top-down design system, this work is only one part of the rough processes generating module. The algorithm is the foundation to build up the fabricating features and for the following constraint based revising process.

Acknowledgments. This work was financially supported by the Scientific Research Program Funded by Shaanxi Provincial Education Department (No. 11JK0864), Science and Technology Development Plan Foundation of Shaanxi Province (No. 2011K07-11), President Fund of Xi'an Technological University (No. XAGDXJJ1007) and Shaanxi Major Subject Construction Project.

References

1. Chong, M.H., Jaafar, H., Hasan, W.Z.W., Hafie, S.S., Hamidun, M.N., Isa, M.M., Hawary, A.F.: An Experiment of Thick Film Force Sensor Using MEMS Simulation Software. Journal of Theoretical and Applied Information Technology 47(3), 902–910 (2013)
2. Zhang, H., Guo, H., Zhang, D., Xu, J., He, Y.: Research on Computer Aided MEMS Process Integration Technology. Nanotechnology and Precision Engineering 2(3), 229–233 (2004)
3. Chang, H., Xie, J., Xu, J., Yan, Z., Yuan, W.: One Novel MEMS Integrated Design Tool with Maximal Six Design Flows. Chinese Journal of Sensors and Actuators 19(5), 1323–1326 (2006)
4. Schlipf, M., Bathurst, S., Kippenbrock, K., Kim, S.G., Lanza, G.: A structured approach to integrate MEMS and Precision Engineering methods. CIRP Journal of Manufacturing Science and Technology 3(3), 236–247 (2010)
5. Khan, F., Bazaz, S., Sohail, M.: Design, Implementation and Testing of Electrostatic SOI Mumps Based Microgripper. Microsystem Technologies 16(11), 1957–1965 (2010)
6. Hu, F., Yao, J., Qiu, C., Ren, H.: A MEMS Micromirror Driven by Electrostatic Force. Journal of Electrostatics 68(3), 237–242 (2010)
7. Li, J., Liu, Y., Ling, H., Guo, W., He, G.: Systematic Direct Solid Modeling Approach for Surface Micromachined MEMS. Advanced Materials Research 433, 3130–3137 (2012)
8. Gao, F., Hong, Y.S.: Function-Oriented Geometric Design Approach to Surface Micromachined MEMS. In: Technical Proceedings of the 2004 NSTI Nanotechnology Conference and Trade Show, Boston, USA, March 7-11, vol. 1, pp. 319–322 (2004)
9. Li, J., Gao, S., Liu, Y.: Solid-Based CAPP for Surface Micromachined MEMS Devices. Computer-Aided Design 39(3), 190–201 (2007)
10. Shaikh, M.Z., Kodad, S.F., Jinaga, B.C.: Performance Analysis of Piezoresistive Mems for Pressure Measurement. Journal of Theoretical and Applied Information Technology 4(3), 227–231 (2008)

11. Xu, J., Yuan, W., Chang, H., Yu, Y., Ma, B.: Angularly Parameterized Macromodel Extraction for MEMS Structures with Large Number of Terminals. Journal of System Simulation 22(3), 748–751 (2010)
12. Shaikh, M.Z., Kodad, S.F., Jinaga, B.C.: Modeling and simulation of MEMS characteristics: A Numerical Integration Approach. Journal of Theoretical and Applied Information Technology 4(5), 415–418 (2007)
13. Fan, Z., Wang, J., Achiche, S., Goodman, E., Rosenberg, R.: Structured Synthesis of MEMS Using Evolutionary Approaches. Applied Soft Computing 8(1), 579–589 (2008)

Parallel Process of Virtual Screening Result File Based on Hadoop

Ning Ma, Rongjing Hu, and Ruisheng Zhang

School of Information Science and Engineering,
Lanzhou University, Lanzhou, 730000, China

Abstract. Virtual screening (VS) is a advanced technology to find some potential drugs from a large scale of small molecules, which will generate intensive results data. The data processing in VS is a key problem because of its scale. In the present work, we performed a study of exploiting MapReduce computing model to Parallel Process of Virtual Screening Result File. The results showed that it costs much less time using the combined methods. Hence, we recommend the efficient methods to process the results from large scale virtual screening.

Keywords: Hadoop, MapReduce, Virtual Screening, Dock, Bigdata.

1 Introduction

Drug virtual screening is a pharmaceutical technology based on drug design theory using computer simulation technology and professional software applications. With the technology development potential compounds can be easily discovered from a large number of compounds and the activity of these compounds can be predicted[1]. Thus, virtual screening is widely by chemists and pharmacologist to help reduce the number of the experimental screening compounds greatly, shorten the development cycle, and save the expenditure of funds.

DOCK is one of the most popular software to do docking in virtual screening[6]. This software can automatically simulate the role of the ligand in the receptor active sites and record the best way of interaction ultimately. DOCK program uses score function to evaluate match degree between ligand and receptor. In result files of docking the score value includes atom contact score and energy score. In the whole process, chemists concern about the score values most, because they will select molecules with the top grid scores to be the most potential leading compounds. To extract the grid score records, sort and analyze them is a time-consuming task. A large scale of virtual screening study means millions of molecules and enormous calculation. In our previous study of anti-H5N1 drug virtual screening, we docked tens of millions compounds, which means tens of millions docking result files need to process. The traditional stand-alone process mode cannot meet the demand for large data process, thus we need a simple, extended method.

In order to deal with the calculation of big data, the common solution is distribute parallel computing. Distributed computing mode divides a large-scale

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 511–517, 2013.

data calculation into many small-scale data calculaton. It must be mentioned that distributed processing cannot improve the physical resources utilization efficiency. On the contrary, it will consume more physical resources compared with large server mode. It is just a trade-off between physical resource and time. In other words it uses physical resources in exchange for the performance improvements of time. In the past ten years, how to process the intensive data from virtual screening efficiently is still a key problem. Since the appearance of Hadoop framework, chemists attempted to introduce it to solve the problem of intensive data processing and parallel computing[7]. Our team has done some research in this field[5]. We are trying to transplant our virtual screening work from chemist grid to Hadoop platform and process results data by MapReduce of Hadoop framework. MapReduce is a simplified program model of parallel computing. it makes those who have little parallel compute experience can easily develop parallel applications. Hadoop realize Google File System and MapReduce model, It is not just a distributed storage system but also a framework of distributed applications which running on large clusters. In this paper, we based on MapReduce compute model and exploit Hadoop framework process dock's results file. We can quickly screen higher gridscore molecular file which facilitate chemist for further process.

2 Hadoop and MapReduce

Hadoop is a distributed system infrastructure[4]. It takes full advantage of cluster's high-speed and storage capacity. The core component of Hadoop is HDFS(Hadoop Distributed File System) and MapReduce[2]. HDFS is a distributed file system. It provides the high throughput rate to access the application data, which is suit for those applications that have large data sets.

MapReduce is a programming model and an associated implementation for processing and generating large datasets that is amenable to a broad variety of real-world tasks[3]. MapReduce programming model originated from functional languages, mainly through "Map"and "Reduce" two-step to parallel process massive data sets. Data sets (or task) handle by MapReduce must have this feature: The pending data sets can be divided into many small dataset, and each small data sets can parallel process. The datasets in our work are full compliance with this feature. Each MapReduce task is initialized for a Job, each Job can be divided into two stage: map and reduce. Two stages correspond two functions, map function and reduce function. map function receive input of a <key,value >form, And then produce a <key,value > form intermediate output. reduce function also receives an input form of <key,(list of values)>, and process each value according to the reduce function, Each reducer have 0 or 1 output. reduce 's output is also in the form of <key,value >. MapReduce's work principle as follows Fig.1.

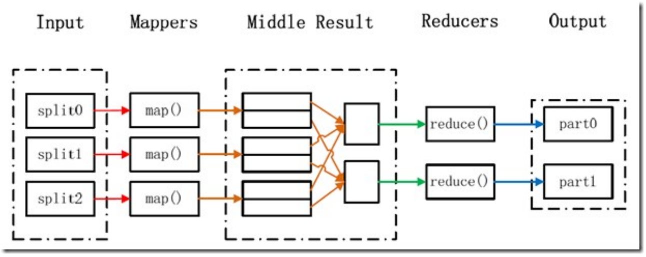


Fig. 1. MapReduce working principle

3 Traditional Methods

In virtual screening, massive small molecules and proteins are docked by DOCK program, and it will produce massive results files. The result files contain a lot of information, such as molecule ID number of conformations, conformation coordinate and grid score values(Fig.2). Chemists are concerned about score values most . The higher the grid score value is, the better the activity is. Hence, its necessary for the chemists to extract and sort the grid score value.

In the traditional stand-alone process mode, we used the method: Heap-based TopN to filter out dock result file. In the methods the time complexity is calculated by the formula of M * logN, here, M is the total number of molecules; N is the number you want to filter out. The concrete steps are follows:

maintain a N-size heap, it is used for saving the sets of docking result files. The Minimum is the root of the heap ptop[0], Then traverse tens of millions records, and inserted these records into the heap sequentially, If the current records is greater than the root node, then replace the root node by the current data.

Then do SiftDown operation on the root node based on the heap's structural property, so that to keep the result set is still a mini root heap.

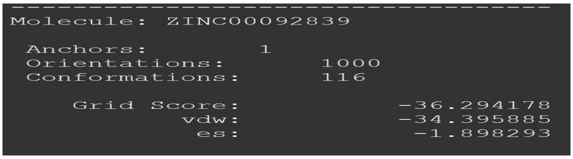
With the heap structure, we can complete the search and move task in log level. Such time complexity is M * logN. Pseudocode as follows:

1. Post-process: extracted molecule ID and grid score from docking results files line by line.
2. Create a N-size heap to save dock result file sets

```
for(i=1; i<n;i++)
{
ptop[i] = pdata[i];
siftup_rear(ptop,i+1); \ \ In addition to the last element, the other part [0 .. n-2] is already heap. By the following function, exchange the last element to the forward and adjust the entire array into a heap
}
```

3. Traverse records, inserted these records into the heap sequentially, and adjust structure according to the heap's characteristics

   ```
   for(i=n;i<m;i++)
   {
   register t = pdata[i];
   if(ptop[0]>= t) continue;
   ptop[0] = t;
   siftdown_head(ptop,n);\ \ In addition to the first element, the other part [1
   .. n-1] is already heap. By the following function,exchange the first element
   to the subsequent and adjust the entire array into a heap
   }
   ```



```
------------------------------------
Molecule: ZINC00092839

 Anchors:          1
 Orientations:        1000
 Conformations:       116

    Grid Score:              -36.294178
             vdw:            -34.395885
              es:             -1.898293
```

Fig. 2. The content in the docking result files

4 The Processing of Virtual Screen Results File Based on Hadoop Framework

In order to deal with the calculation of big data, the common solution is parallel computing. In this paper, we exploit MapReduce compute model and Hadoop framework processing dock results file. Because of the high-speed computing ability and extended storage capacity of Hadoop, we can quickly screen the results files and search the ones with higher grid score, which facilitate chemists for further processing.

The MapReduce job will be divided into two parts. In the first stage, the task is extracting score function values and small molecules ID from massive docking results file (see Fig.3). In the second stage, the task is filtering out those small molecules which have higher score values. It will use the sequential combination to organize MapReduce job. The output from the task of the first stage serves as the input data of the second stage.the operation details are described as follows:

1. The first job 's main goal is quick parallel extracted ZincID and gridscore from massive dock result file.

   In the Map stage, the data were divided into into M blocks. Small molecule ID and correspond score values were extracted in each map task. The Reduce stage summarized the various tasks. According to the characteristics of small molecule file(see Fig. 2), every small molecule 's dock information account 11 lines. so we must re-implement the hadoop 's input class to tell hadoop how to processing input correctly. than we can use the regular expression or other string processing technology to extract zincid and gridscore in each record. Map and Reduce 's input and output forms as follow: Input: <LongWritable key, Text text>output: <FloatWritable Gridscore , Text ZincID >

2. The second job 's main goal is filter out those small molecules with higher value and output those record

   The basic principle is exploit the hadoop's key automatically sort order, then we just need output the first N records with higher gridscore. However, in order to further speed up processing, we can pre-execute local TopN, This method can further cut down the number of Mapper output and relive Reducer's pressure. On the other hand ,we can still use the mentioned method :Heap-based TopN way to screening locally.(see Fig.4) Just the scale of the problem is divided into multiple parts which parallel processed by each maper.

   Among this task the Map's Input format is <FloatWritable Gridscore Text ZincID >and the output is <FloatWritabe Gridscore Text zincID>Reduce 's input is <FloatWritable Gridscore Text ZincID>Reduce 's output format is <FloatWriable gridsocre, Text text>

5 Further Improvements

1. Custom combiner
   We can customized our own combiner according to the needs, In Map stage to merge the same key value of score can reduce the amount of intermediate results data. This measure can cut down some network transmission overhead.
2. Adjust the number of reducer
   In general, there is only one reducer that means all intermediate data will be sent to this unique reducer. This situation is another reason that lead to the task execution becomeing very slow. This design idea can guarantee a high degree of parallelism in the map phase, but in the reduce stage, there will not be parallel. We can add the number of the reducer appropriately in the first part.
3. Merge small files into large files
   HDFS are not suitable for dealing with small files, in this paper every dock result file is about 5M. we can merge small files into a large files but not over the hadoop block size. Because Our process is based on multi-line ,if file size is over hadoop block size, hadoop will spilit it and it will result in incorrectly parse out the molecular information. on the other hand we can increase the value of the block size in order to cut down the number of mapper's input. Further more we can establish index for the original file to improve the access efficiency of massive small-file.[2]

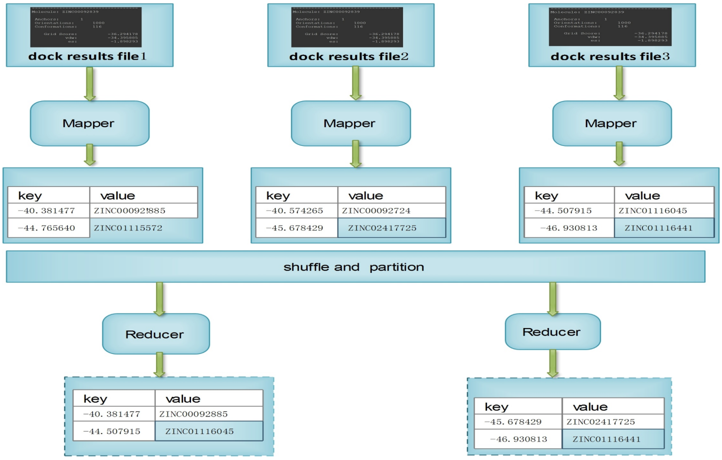


Fig. 3. First Stage of MapReduce Job

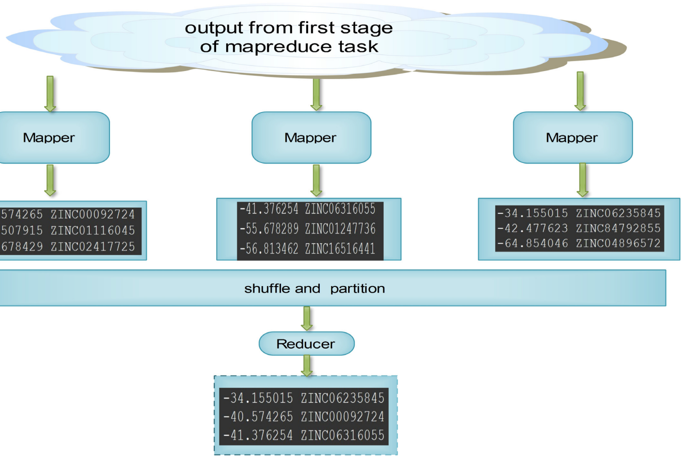


Fig. 4. Second Stage of MapReduce job

6 Performance Contrast

Experimental design: there are 16G Dock result files which contains nearly 60 million small molecule record. we will extract 10000 record with higher score. The Hadoop parallel computing platform consists of 5 servers. The configuration of each server is redhat5.3 CPU2.5GHz Memery4GB Harddisk 1800G. Dock result file is divided into four groups, the comparison of two ways can be seen table1.

As can be seen from table1. we can conclude that compared with traditional methods the efficiency of processing the data is evidently improved with Hadoop method, especial when the data becomes larger, Hadoop's advantage is amost overwhelming!

Table 1. The contrast of hadoop method and traditional method

Group	Data Size	Record Numbers	Traditional Methods	Hadoop methods	save time
1	2GB	7.6Million	250s	299s	-49s
2	4GB	15Million	657s	575s	82s
3	8GB	30Million	1410s	750s	660s
4	16GB	60Million	2960s	1347s	1613s

7 Conclusion

In this paper, we exploit MapReduce compute model and exploit Hadoop framework to process virtual screening results files. We can screen docking results files rapidly by the advantage of the high-speed compute ability and storage capacity of Hadoop. We performed two types of experiments, relatively by traditional methods and our new methods. From the results, we can concluded that the news methods can improve the processing efficiency greatly.

Additionally, the improvement is more obvious when the data becomes larger. The results of the experiment further verify that Hadoop framework is suitable to deal with the big data. In conclusion, the MapReduce computing model and Hadoop framework is significant in solving the problem of large data processing in drug virtual screening and will promote the development of drug discovery. Although this way consume more physical resources compared with large server mode. But it saves time. it is a trade-off between physical resource and time. with the cost of the hardware is becoming less and less, it is still a better way to solve large-scale data problem.

References

1. Ai-lin, L.I.U., Guan-hua, D.U.: Research progress of virtual screening aided drug discovery. Acta Pharmaceutica Sinica 06, 566–570 (2009)
2. Chunming, Z.J.R., Tingting, H.: An approach for storing and accessing small files on hadoop. Computer Applications and Software 29(11) (2012)
3. Dean, J., Ghemawat, S.: Mapreduce: simplified data processing on large clusters. Commun. ACM 51(1), 107–113 (2008)
4. Apache Hadoop. hadoop, http://hadoop.apache.org/
5. Hu, R., Barbault, F., Maurel, F., Delamar, M., Zhang, R.: Molecular dynamics simulations of 2-amino-6-arylsulphonylbenzonitriles analogues as hiv inhibitors: Interaction modes and binding free energies. Chemical Biology & Drug Design 76(6), 518–526 (2010)
6. Kuntz. dock, http://dock.compbio.ucsf.edu/DOCK_6/index.htm
7. Watson, P., Leahy, D., Cala, J., Sykora, V., Hiden, H., Woodman, S., Taylor, M., Searson, D.: Cloud Computing for Chemical Activity Prediction, Computing Science, Newcastle University (2011)

3D Modeling Environment Development for Micro Device Design

Zheng Liu

School of Mechatronic Engineering, Xi'an Technological University, Xi'an China
zheng.liumail@gmail.com

Abstract. Along with the development of fabricating processes, the structure of micro device becomes more and more complex. The traditional design tools begin with the processes design, which is not in a perceptually intuitive way. The 3D modeling method is presented to improve design efficiency. Above all, the data structure of the feature-based model is illustrated, by which the inner data of the 3D components to build up micro device is organized. Then, the 3D visualization environment is constructed to make the joint between the inner data and the interactive scene that receives the commands from the designers. By using this method, designer can build up the 3D model of micro device with the more convenient way.

Keywords: 3D modeling, Micro device, Data structure, Interactive environment.

1 Introduction

Nowadays, the structure of micro device becomes more and more complex. Especially for surface micromachined device, multiple masks for the structural and sacrificial layers made the designing work boring. Because of the similar fabricating processes with the integrated circuit, the traditional design way of surface micromachined device begins with the mask design, which is the 2D designing method in some sense [1]. The disadvantage of this flow is that the designers have to consider the fabricating issue from the beginning of the mask design. It is even more depressing that the structure of micro device is not the exact one they desired after finishing a lot of designing work layer by layer. Moreover, there is not sufficient consideration of the issue to meet the requirements of the function and performance because of the fabricating constraint. Therefore, how to find the more intuitive and reliable way to build up the model of micro device becomes a critical problem.

To avoid the long iterative design cycle of the bottom-up design flow, the top-down design method was proposed to provide the more intuitive and comfortable way to design micro device, which is summarized as function-to-shape-to-mask [2]. The former link is called function-to-shape that involves the modeling and simulation on system level [3]. The latter link is the guarantee of manufacturability, which indicates the advanced way to get the mask from the structure of micro device [4]. However, the popular tools still follows the traditional way beginning with the processes design, which doesn't conform to the advanced flow [5]. As the effective way to construct

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 518–523, 2013.

model of mechanic products, the feature technology has been developed for a long time [6]. To improve design efficiency, the feature technology was introduced into micro device design area [7,8]. However, the fabricating feature recognition is still a problem whether for micro devices or for mechanical products [9]. Meanwhile, for the convenience of the 3D modeling, the mechanical design tools are used to verify the modeling theory [10]. However, because of the different character of micro fabrication, the mechanical design tool is not suitable for the proposed design flow. For MUMPs (Multi-User MEMS Processes), one of the standardized processes of micro device, the fabricating procedure is performed layer by layer with the cantilever structures supported by the sacrificial layers, which is quite different from the mechanical product [11]. Therefore, building the 3D designing environment of micro device that conform to the top-down design flow is the key issue to improve the designing efficiency. In this paper, a 3D modeling method is presented. By means of JAVA technology, the design environment is built up, which is also the foundation for the following cooperative activities.

2 The Information Framework to Construct 3D Model

The hierarchical organization of the modeling information is shown in Figure 1. There are three levels organizationally to deal with the data and operations of different types. The data of the voxel primitive is on the bottom level. Above it is the feature level, which involves the features, operations and the relationships. It is the device level that is on the top level and related to the scene operations.

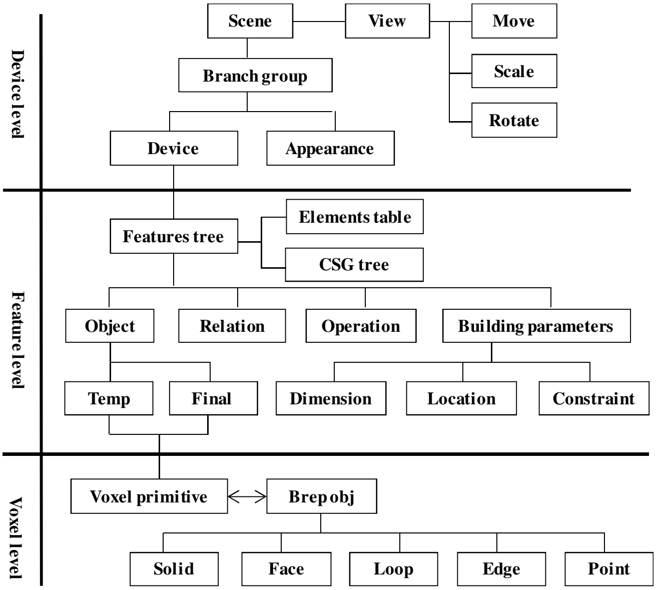


Fig. 1. The hierarchical organization of the modeling information

3 The Data Structure of Feature Based Model

Figure 2 illustrates the data structure of the feature based model. The data structure is based on the hybrid modeling technology, which means that the constructing procedures are described by CSG method for both the features and the layers, while the inner data structure is boundary representation.

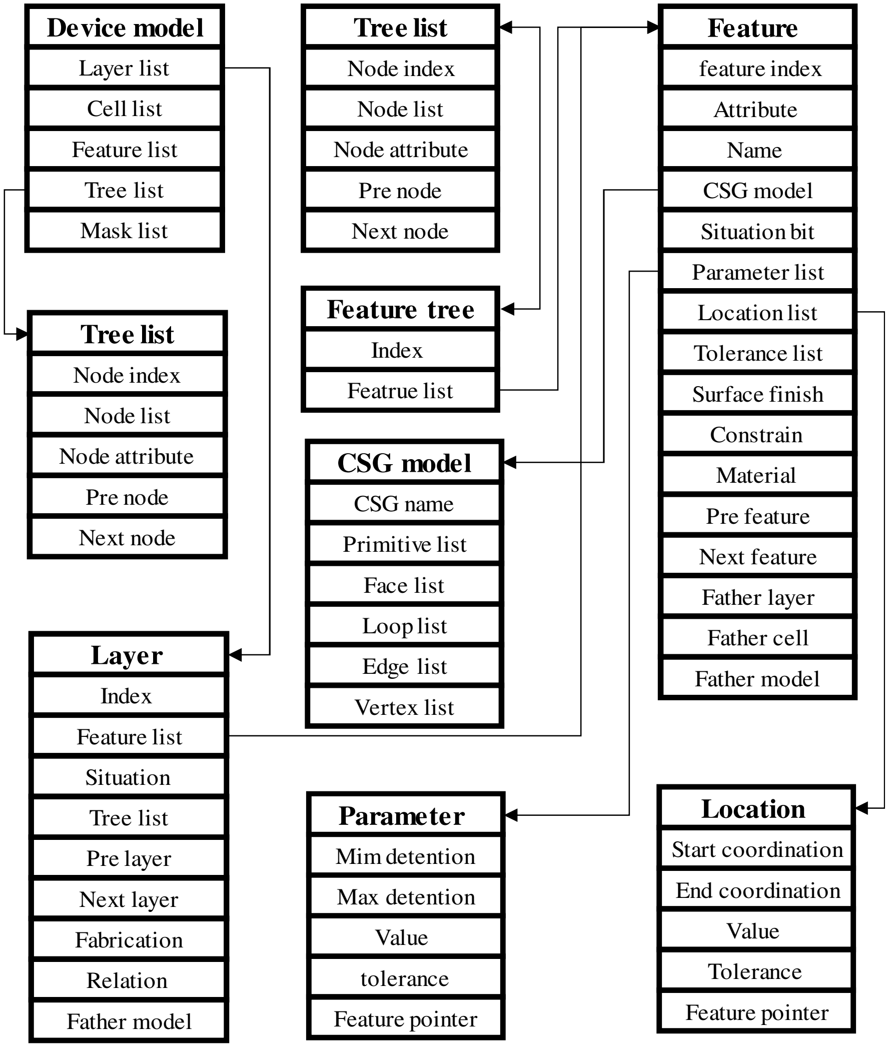


Fig. 2. The data structure of the feature based model

The data structure is implemented with JAVA technology, which is invisible to users. To build up the interactive visualization environment, the 3D display kernel is indispensable. Here, JAVA 3D is adopted as the linkage between the designer application and the low-level data.

4 JAVA 3D Based Visualization

Java 3D is the application programming interface based on the scene graph mode, which is seamlessly integrated with the Java platform. The visualization of 3D micro device model by JAVA3D technology is shown in Figure 3. The key point of modeling visualization lies in the relationship built between the elements of features and the components of 3D scene.

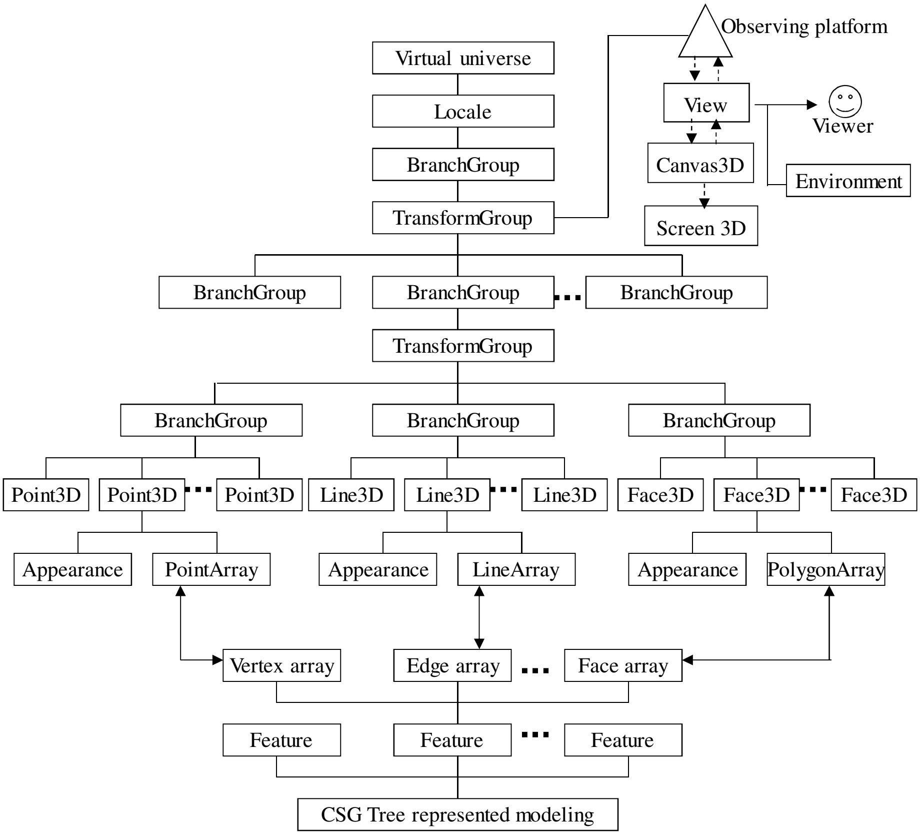


Fig. 3. The modeling visualization by JAVA3D technology

To construct JAVA3D environment involves several steps:

1) Building up relationship between the features and the elements of 3D nodes
The features are represented by boundary representing method. Each of the features has the geometric and topological information involving the solids, the faces, the loops, the edges and their relationships. To display the features in 3D scene, the relationship between the geometric elements and the elements of the 3D nodes in JAVA3D scene is build up. For example, the object of line3D has the attributes of appearance and line array that is mapped to the edge array of the features. These 3D

objects have the ability to detect the interactive operations of users. Based on the relationship, the inner data of the model can be retrieved and updated for particular requirements to revise the model such as the interactive fabricating parameters revising procedure.

2) The organization of the 3D scene

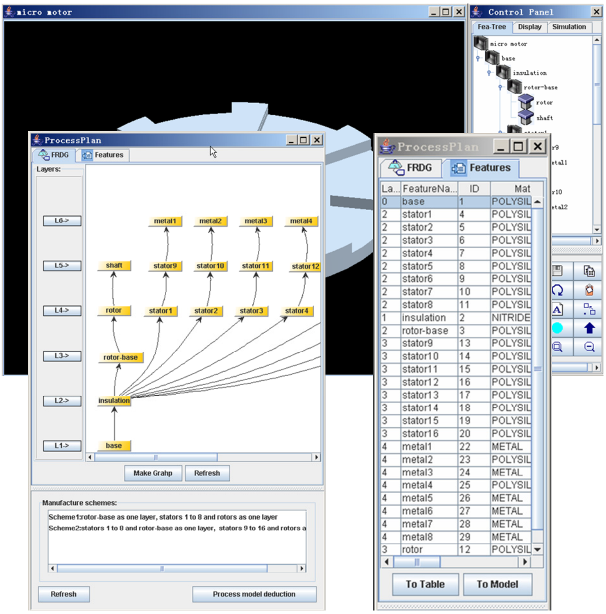
The virtual universe is the root and the only root of the 3D scene. There are many local nodes under the root. Only one locale node is displayed at a time. The locale node involves many branch group nodes, on which the transform group node and shape3D node are built up. The appearance of the shape is stored in the appearance node.

3) Constructing the interactive environment

The observing platform deals with the observation and operation of the designer. To realize the operations like scale and transform, the events of mouse are monitored and the data are transmitted to corresponding module to control the scene as the designer wanted. The designer can retrieve the low level data such as selecting certain faces of one structural layer, which is the common operation for the algorithm of etching simulation.

5 Implementation

The 3D modeling environment of micro device is shown in Figure 4. The system is implemented by JAVA technology. The primitive elements and features are parametric and constructed by CSG. By the locating and transforming operations, the 3D model is built up conveniently.



Manufacture schemes:

Scheme1:rotor-base as one layer, stators 1 to 8 and rotors as one layer

Scheme2:stators 1 to 8 and rotor-base as one layer, stators 9 to 16 and rotors a

La...	FeatureNa...	ID	Mat
0	base	1	POLYSIL
2	stator1	4	POLYSIL
2	stator2	5	POLYSIL
2	stator3	6	POLYSIL
2	stator4	7	POLYSIL
2	stator5	8	POLYSIL
2	stator6	9	POLYSIL
2	stator7	10	POLYSIL
2	stator8	11	POLYSIL
1	insulation	2	NITRIDE
2	rotor-base	3	POLYSIL
3	stator9	13	POLYSIL
3	stator10	14	POLYSIL
3	stator11	15	POLYSIL
3	stator12	16	POLYSIL
3	stator13	17	POLYSIL
3	stator14	18	POLYSIL
3	stator15	19	POLYSIL
3	stator16	20	POLYSIL
4	metal1	22	METAL
4	metal2	23	POLYSIL
4	metal3	24	METAL
4	metal4	25	POLYSIL
4	metal5	26	METAL
4	metal6	27	METAL
4	metal7	28	METAL
4	metal8	29	METAL
3	rotor	12	POLYSIL

Fig. 4. The 3D modeling environment

6 Conclusion

The 3D modeling method of micro device is proposed in this paper. It provides the foundation for construction of the design environment conforming to the advanced shape-to-mask designing flow. With this method, the designer can build up the model of micro device with the 3D features or primitive elements just like the mechanical design tools, which follows the popular designing conventions. With respect to the overall top-down theory, this work is related to the fundamental data structure of the information framework. It is also the foundation to implement the following work of simulation and verification.

Acknowledgments. This work was financially supported by the Scientific Research Program Funded by Shaanxi Provincial Education Department (No. 11JK0864), Science and Technology Development Plan Foundation of Shaanxi Province (No. 2011K07-11), President Fund of Xi'an Technological University (No. XAGDXJJ1007) and Shaanxi Major Subject Construction Project.

References

1. Schlipf, M., Bathurst, S., Kippenbrock, K., Kim, S.G., Lanza, G.: A structured approach to integrate MEMS and Precision Engineering methods. CIRP Journal of Manufacturing Science and Technology 3(3), 236–247 (2010)
2. Fedder, G.K.: Top-Down Design of MEMS. In: Proceedings of the 2000 Int. Conf. on Modeling and Simulation of Microsystems Semiconductors, Sensors and Actuators, San Diego, USA, vol. 3, pp. 7–10 (March 2000)
3. Shaikh, M.Z., Kodad, S.F., Jinaga, B.C.: Modeling and simulation of MEMS characteristics: A Numerical Integration Approach. Journal of Theoretical and Applied Information Technology 4(5), 415–418 (2007)
4. Fan, Z., Wang, J., Achiche, S., Goodman, E., Rosenberg, R.: Structured Synthesis of MEMS Using Evolutionary Approaches. Applied Soft Computing 8(1), 579–589 (2008)
5. Chang, H., Xie, J., Xu, J., Yan, Z., Yuan, W.: One Novel MEMS Integrated Design Tool with Maximal Six Design Flows. Chinese Journal of Sensors and Actuators 19(5), 1323–1326 (2006)
6. Li-an, C., Ying, X., You-mei, Z.: Automatic feature recognition based on entity model. Journal of Shanghai Normal University 2(1), 161–165 (2010)
7. Gao, F., Hong, Y.S., Sarma, R.: Feature Model For Surface Micro-machined MEMS. In: Proceedings of ASME Design Engineering Technical Conferences, Chicago, USA, vol. 1, pp. 149–158 (2003)
8. Li, J., Gao, S., Liu, Y.: Feature-based process layer modeling for surface micro-machined MEMS. Journal of Micromechanics and Microengineering 15(3), 4620–4635 (2005)
9. Hayasi, M.T., Asiabanpour, B.: Extraction of manufacturing information from design-by-feature solid model through feature recognition. The International Journal of Advanced Manufacturing Technology 44(11), 1191–1203 (2009)
10. Zhang, C., Jiang, Z., Lu, D., Ren, T., Wang, J.: Design for Micro-Electro-Mechanical Systems Devices Based on Three-Dimensional Features. Journal of Xi'an Jiaotong University 41(5), 571–575 (2007)
11. Hu, F., Yao, J., Qiu, C., Ren, H.: A MEMS Micromirror Driven by Electrostatic Force. Journal of Electrostatics 68(3), 237–242 (2010)

Data Reconciliation of Release Mechanism Research of LDH-Based Drug

Xiaoxia Liu

Information Technology Department of Qingdao Vocational & Technical College
of Hotel Management, Qingdao, Shandong, China, 266100
gaolin0619@126.com

Abstract. Sebacate pillared layered double hydroxides (LDH) was prepared via co-precipitation method. And then the drug 10-hydroxy-camptothecin (10-HC) was intercalated into the gallery of LDH to form the drug–LDH composites. Three types of dissolution- diffusion kinetics models were used to make clear the drug release mechanism of the LDH composites. For the first time data reconciliation was used to make clear the drug release kinetics mechanism of the LDH composites, which was carried out by using a data filter algorithm based on first-order delay filter and time-delay principle to diminish random experimental errors resulting from the factors including stirring speed, solid content, and so on. Simulation results indicated that the data reconciliation adequately degraded the interactions between drug release rates at different times and made the data satisfy the kinetics models for the drug release mechanism more accurately.

Keywords: 10-HC, LDH, Release mechanism, Data reconciliation, Data filter.

1 Introduction

10-hydroxyl-camptothecin (10-HC), isolated from the Chinese tree, *Camptotheca acuminata,* inhibits the activity of topoisomerase I and has a broad spectrum of anticancer activity in vitro and in vivo[1]. But it has poor water solubility, leading to difficulties in efficient dose delivery and unwanted side effects. Layered double hydroxides (LDHs) are a family of layered inorganic materials with structurally positively charged layers and interlayer balancing anions [2-3]. The interlayer gallery of LDHs may be considered as a molecules vessel and be potentially used in drug controlled release systems. For studying the release mechanism, data processing is needed according to the dynamic model[4-6], and then suitable dynamic equation is chosen by the related coefficient, so release mechanism is determined. But, during experimental process, there are many factors such as stirring speed, solid content which influence the final results. These factors can introduce errors to release percentage and result in unfavorable experimental results. So, in order to make the results more accurate, data filter algorithm is often used to diminish random error[7].

The conventional data filter algorithms include recursive mean filter, first-order delay filter and least mean square filter[8-9], but these methods have a common

Y. Tan, Y. Shi, and H. Mo (Eds.): ICSI 2013, Part II, LNCS 7929, pp. 524–530, 2013.

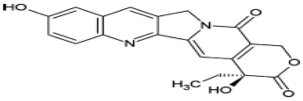
limitation, that is, the parameters of filter need to be set manually for different problems, which will lead in-and-out filter results for a dynamic process. On this occasion, a modified first-order delay algorithm is proposed[10].

Conventional first-order delay filter has two limitations, the first is the delay constant need to be set different value according to instantaneous anticipant fluctuant value; the second is delay constant must be adjusted to a bigger value to control signal fluctuation but this will lead to obvious delay and can not track real value effectively. Aiming at the above two limitations, a parameter (dynamic degree index) dyn is introduced to make delay constant be adjusted automatically[10]. In the next sections, a new data filter algorithm will be proposed and used in release mechanism research of LDH-based drug.

2 Materials and Methods

2.1 Reagents

10-HC (99% purity) is purchased from Haobo Co. Ltd. (Hubei, China) and used as received. Its structure is as follows. The other chemicals and solvents of analytical grade are purchased from National Medicines Co. Ltd. (China) and used without further purification.



2.2 Preparation of 10-HC/Se/LDH Composites

The mixed salt solution of $Mg(NO_3)_2 \cdot 6H2O/Al(NO_3)_3 \cdot 9H2O$ in a molar ratio of 2:1 is prepared with a total metal ion concentration of about 0.5 mol/L^{-1}. Then the co-precipitating agent, diluted ammonia water (6 wt%) which contained 0.05mol sodium sebacate, is added to the mixed salt solution at a speed of 25 mL/min and the mixture is stirred till the final pH value reached 9.5. The precipitate is aged in the mother solution at room temperature for 0.75h, 3h and 6h, respectively, and then the precipitates are filtered and washed with deionized water until the pH value reached about 7.0. The filter cake held in a glass bottle is peptized at a constant temperature of 80℃ in an oven for about 24h to obtain sebacate pillared Zn–Al–NO_3 LDH (Se/LDH) sol sample. 0.5 g of the wet Se/LDH sample is dispersed in 250ml saturated ethanol solution of 10-HC. The mixture is stirred for a desired time at a desired temperature. After that, the mixture is centrifuged and the precipitate is washed twice with ethanol, and the obtained 10-HC/Se/LDH composites after being dried at 60℃ in a vacuum oven for 45min, 3h and 6h are noted as 10-HC/$Se_{0.75h}$/LDH, 10-HC/Se_{3h}/LDH, 10-HC/Se_{6h}/LDH, respectively.

2.3 Determination of Release Rate

The drug release studies are performed at 37℃ in a pH=7.2 buffer solution (0.1M). The composite sample of 50mg is placed into the buffer solution of 500ml and the suspensions are stirred at 37℃. 4mL of solutions is withdrawn at predetermined time intervals and filtered through a 0.45μm syringe filter. The absorbances are measured at λ_{max}=370 nm by UV–vis absorption spectrophotometer to obtain the 10-HC concentrations in solutions, in turn to calculate the release percentage (X_t) of 10-HC . The tests are repeated three times and the final values are an average of measurements. The release percentage values of 10-HC from the composites are plotted versus time to examine the release rates of the drug from the controlled release system.

2.4 A New Data Filter Algorithm

Our primary filter algorithm based on first-order delay filter and time-delay principle is proposed in [10]. For dynamic data reconciliation, algorithm inertia should be strengthened so as to swell algorithm disturbance resistibility. On the other hand, algorithm inertia should be weakened so as to swell algorithm fast tracking ability. So, the new algorithm in this paper is designed as follows:

Suppose X_k is current sample value, X_{k-1} is previous sample value, let $d_k = X_k - X_{k-1}, \Delta d_k = d_k - d_{k-1}$. And set function rules as shown in Eqs. (1)-(4):

$$\begin{cases} up_k = up_{k-1} \cdot C_0, & if\ \Delta d_k > 0 \\ up_k = up_{k-1} \cdot C_1, & otherwise \end{cases} \tag{1}$$

$$\begin{cases} dn_k = dn_{k-1} \cdot C_0, & if\ \Delta d_k < 0 \\ dn_k = dn_{k-1} \cdot C_1, & otherwise \end{cases} \tag{2}$$

$$tr_k = (1-\alpha) \cdot tr_{k-1} + \alpha \cdot \max\{up_{k-1}, dn_{k-1}\} \tag{3}$$

$$dyn_k = 1 - \exp(-tr_k) \tag{4}$$

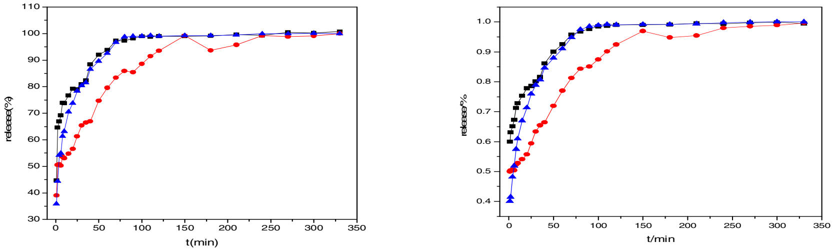
Where up_k is up-trend factor, dn_k is down-trend factor, the two factors would augment observably and restrain each other when system state changed evidently. C_0 and C_1 are constant values confirmed through experiments. tr_k is dynamic degree factor with some delay which could reflect system change degree in time. Because tr_k would augment when system signal changed observably, delay factor is introduced to avoid tr_k leap. dyn_k is system dynamic degree index. The new algorithm output is described by Eq. (5):

$$Y_k = Y_{k-1} + \frac{dyn_k}{dyn_k / 2 + 1} * (X_k - Y_{k-1}) \tag{5}$$

Where Y_k is filter value of X_k.

3 Results and Discussion

The in vitro release curves of 10-HC from the nanohybrids aged for different times is shown in Fig.1. (a). It can be seen that a rapid release of 10-HC occurs at the initial stage, which is followed by a slower release of 10-HC. The time for release of 80% 10-HC is 30min for 10-HC/$Se_{0.75h}$/LDH or 10-HC/Se_{6h}/LDH, 60min for 10-HC/Se_{3h}/LDH. Obviously, the Se/LDH sample aged for 3h shows better release effect, indicating that the 10-HC/Se_{3h}/LDH composites could be used as a potential drug controlled release system, due to the restricted motion of 10-HC molecules either by the steric effect of LDH, or by the hydrogen bonds between layers of LDHs and 10-HC molecules intercalated[11-12]. In order to diminish the influence of random errors on release rate (X_t), the proposed data filter algorithm is used. Algorithm parameters are set as C_0=1.5, C_1=1, α=0.5, and the raw results and filtered results are shown in Fig. 1. (a) and Fig. 1. (b), respectively. Compared with Fig. 1. (a), the release curves in Fig. 1. (b) are smoother, suggesting that release percentage(X_t) fits the expectation curve more accurately and the new algorithm is valid.



(a) (b)

Fig. 1. Release curves (a)before and (b) after data reconciliation for 10-HC/$Se_{0.75h}$/LDH (■), 10-HC/Se_{3h}/LDH(●) and 10-HC/Se_{6h}/LDH(▲)

To understand the drug release mechanism of the drug-LDHs nanohybrid systems, three types of dissolution-diffusion kinetics models[4-6] are applied and are stated as follows:

(1) The Bhaskar equation expresses that the drug release could be controlled by the diffusion through the LDH particles, or by the diffusion through the solution layer surrounding the particle as shown in Eq. (7):

$$\ln(1-X_t) = -1.59 * (6/d_p)^{1.3} * D^{0.65} * t^{0.65} \tag{7}$$

(2) Modified Freundlich model suggests the experimental data on ion exchange and diffusion-controlled process fit with Eq. (8):

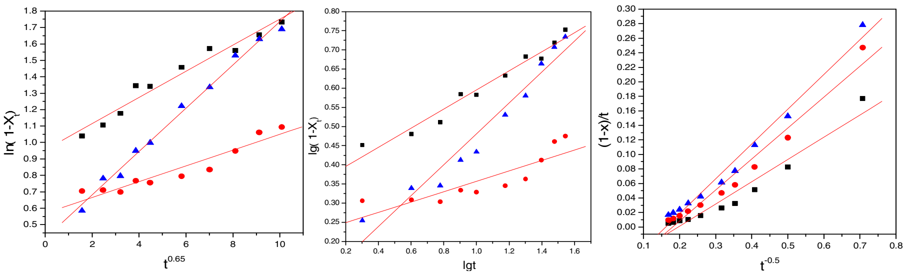
$$\lg(1-X_t) = K_m * \lg t \tag{8}$$

(3) Parabolic diffusion model describes the diffusion controlled phenomena of drug from clay nanohybrids and is written as Eq. (9):

$$(1-X_t)/t = K_d * t^{-0.5} + a \tag{9}$$

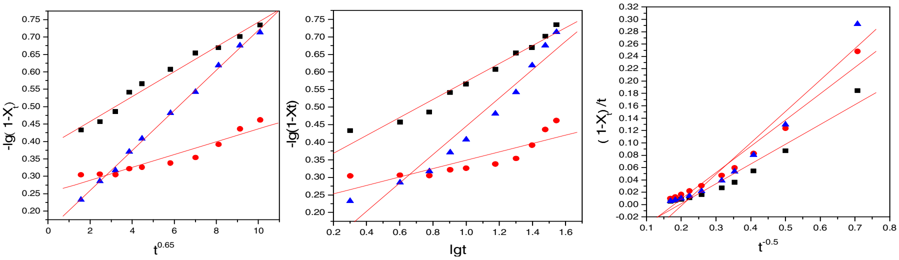
X_t, t, k in the above equations are the release percentage, release time and kinetics constant, respectively.

In order to determine which kinetics model the drug release mechanism of the drug-LDHs nanohybrid systems obey, the release profiles are fitted to the above equations and the corresponding linear correlation coefficient (r^2) is evaluated. Fig. 2, Fig. 3 and Table 1 are obtained by the experimental data before and after data reconciliation using the proposed algorithm.



(a) (b) (c)

Fig. 2. Release kinetics plots before data reconciliation based on (a) Bhaskar equation, (b) modified Freundlich model and (c) parabolic diffusion model



(a) (b) (c)

Fig. 3. Release kinetics plots after data reconciliation based on (a) Bhaskar equation, (b) modified Freundlich model and (c) parabolic diffusion model

Table 1. Corelation coefficients(r^2) obtained before and after data reconciliation from the three kinetics models

Kinetics equation	10-HC/Se$_{0.75h}$/LDH		10-HC/Se$_{3h}$/LDH		10-HC/Se$_{6h}$/LDH	
	r_1^2	r_2^2	r_1^2	r_2^2	r_1^2	r_2^2
Bhaskar equation	0.9796	0.9897	0.9522	0.9495	0.9955	0.9991
Modified Freundlich model	0.9825	0.9826	0.8619	0.8546	0.9754	0.9709
Parabolic diffusion model	0.9861	0.9740	0.9761	0.9828	0.9720	0.9699

The drug release rate of LDH systems is generally controlled either by dissolution of LDH particles or by diffusion through the LDH particles. As can be seen from Fig. 2, Fig. 3 and Table 1, the Bhaskar equation can explain the release processes of 10-HC/Se_{6h}/LDH by providing a more reasonable r^2 value of 0.9955 while the release kinetics fits best with the parabolic diffusion model for both 10-HC/$Se_{0.75h}$/LDH (r^2=0.9861) and 10-HC/Se_{3h}/LDH (r^2=0.9761). After data reconciliation, the r^2 value of 10-HC/Se_{6h}/LDH and 10-HC/Se_{3h}/LDH become 0.9991 and 0.9828, respectively, indicating their release mechanisms fit the Bhaskar equation and the parabolic diffusion model better. The Bhaskar equation is proposed to describe the release process where the diffusion through the particle is the rate limiting step[4], while the parabolic diffusion model describes the intraparticle diffusion[13]. In both cases the diffusion take place together with the leaking of metal ions and the dissolution of brucite layers[14]. The corelation coefficients(r^2) obtained before and after data reconciliation from the three kinetics models in Table 1 suggest the three composites obey different release mechanism, which might be due to the difference in their structures resulting from different aging times[15]. The correlation coefficients changed after data reconciliation in Table 1 shows that the release kinetics of 10-HC/$Se_{0.75h}$/LDH satisfies Bhaskar equation(r^2=0.9897) rather than the parabolic diffusion model (r^2=0.9861), suggesting that the data reconciliation helps to diminish the experimental error in the raw data of 10-HC/$Se_{0.75h}$/LDH and makes a reasonable judgement on the release mechanism.

4 Conclusions

For the first time data reconciliation is used to make clear the 10-HC release kinetics mechanism of the LDH composites. The data reconciliation adopts a data filter algorithm based on first-order delay filter and time-delay principle to diminish random experimental errors, and makes the data satisfy the kinetics models for the 10-HC release mechanism more accurately. It provides a new assistant way in chemistry analysis to dispose experimental results.

Acknowledgments. This work are supported by National High-tech Research and Development Program of China(863 Program) (2009AA04Z14) and the National natural Science Foundation of China(61174040).

References

1. Ping, Y.H., Lee, H.C., Lee, J.Y., et al.: Oncol. Rep. 15, 1273–1279 (2006)
2. Cavani, F., Trifiro, A., Vaccari, C.: Today 11, 173–301 (1991)
3. Hou, W.G., Su, Y.L., Sun, D.J., et al.: Langmuir 17, 1885–1888 (2001)
4. Bhaskar, R., Murthy, S.R.S., Miglani, B.D., et al.: Int. J. Pharm. 28, 59–66 (1986)
5. Yang, J.H., Han, Y.S., Park, M., et al.: Chem. Mater. 19, 2679 (2007)
6. Kodama, T., Harada, Y., Ueda, M., et al.: Langmuir 17, 4881 (2001)
7. Nie, S.L., Wen, J.D., Li, Y.P., et al.: Optimization and Engineering (2001), doi:10. 1007/ s11081-011-9147-1

8. Cervantes, J., López, A., García, F., Trueba, A.: A Fast SVM Training Algorithm Based on a Decision Tree Data Filter. In: Batyrshin, I., Sidorov, G. (eds.) MICAI 2011, Part I. LNCS, vol. 7094, pp. 187–197. Springer, Heidelberg (2011)
9. Zhong, R.X., Yang, Z.: Asian J. of Control 8(1), 36–44 (2006)
10. Gao, L., Liu, X.M., Gu, X.S.: J. of Qingdao Technological University 31(3), 88–93 (2010) (in Chinese)
11. Tyner, K.M., Schiffman, S.R., Giannelis, E.P.: J. Control. Release 95, 501–514 (2004)
12. Lun, D., Yan, L., Hou, W.G., Liu, S.J.: J. Solid State Chemistry 183, 1811–1816 (2010)
13. Yang, J.H., Han, Y.S., Park, M., et al.: Chem. Mater. 19, 2679 (2007)
14. Panda, H.S., Srivastava, R., Bahadur, D.: J. Phys. Chem. 113, 15090–15100 (2009)
15. Pan, D.K., Zhang, H., Zhang, T., et al.: Chemical Engineering Science 65, 3762–3771 (2010)

Author Index