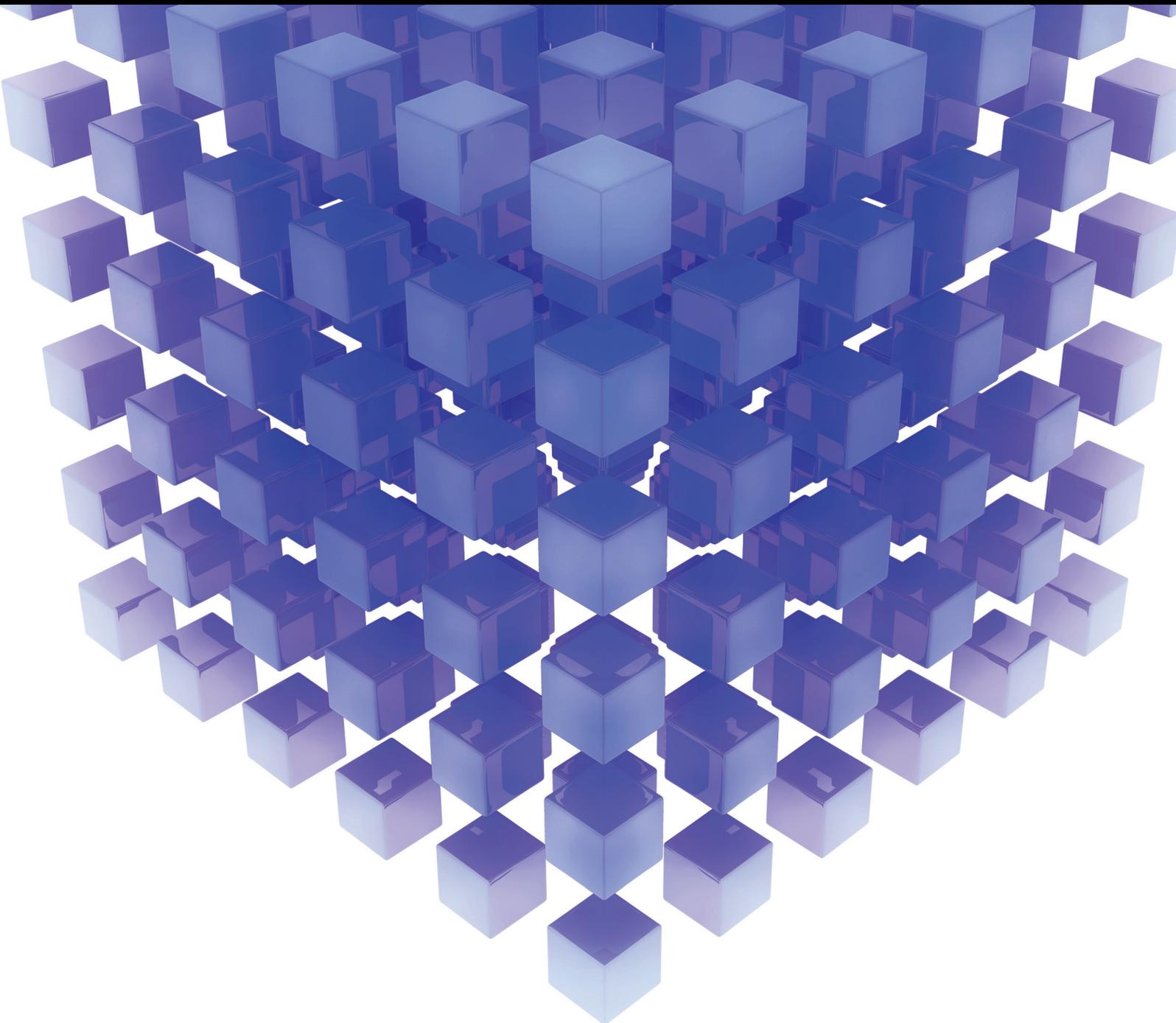


Mathematical Problems in Engineering

Swarm Intelligence in Engineering 2014

Guest Editors: Baozhen Yao, Fang Zong, Bin Yu, and Rui Mu





Swarm Intelligence in Engineering 2014

Mathematical Problems in Engineering

Swarm Intelligence in Engineering 2014

Guest Editors: Baozhen Yao, Fang Zong, Bin Yu, and Rui Mu



Copyright © 2015 Hindawi Publishing Corporation. All rights reserved.

This is a special issue published in “Mathematical Problems in Engineering.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Editorial Board

Mohamed Abd El Aziz, Egypt
Farid Abed-Meraim, France
Silvia Abrahão, Spain
Paolo Addresso, Italy
Claudia Adduce, Italy
Ramesh Agarwal, USA
Juan C. Agüero, Australia
Ricardo Aguilar-López, Mexico
Tarek Ahmed-Ali, France
Hamid Akbarzadeh, Canada
Muhammad N. Akram, Norway
Mohammad-Reza Alam, USA
Salvatore Alfonzetti, Italy
Francisco Alhama, Spain
Juan A. Almendral, Spain
Saiied Aminossadati, Australia
Lionel Amodeo, France
Igor Andrianov, Germany
Sebastian Anita, Romania
Renata Archetti, Italy
Felice Arena, Italy
Sabri Arik, Turkey
Fumihiko Ashida, Japan
Hassan Askari, Canada
Mohsen Asle Zaeem, USA
Francesco Aymerich, Italy
Seungik Baek, USA
Khaled Bahlali, France
Laurent Bako, France
Stefan Balint, Romania
Alfonso Banos, Spain
Roberto Baratti, Italy
Martino Bardi, Italy
Azeddine Beghdadi, France
Abdel-Hakim Bendada, Canada
Ivano Benedetti, Italy
Elena Benvenuti, Italy
Jamal Berakdar, Germany
Enrique Berjano, Spain
Jean-Charles Beugnot, France
Simone Bianco, Italy
David Bigaud, France
Jonathan N. Blakely, USA
Paul Bogdan, USA
Daniela Boso, Italy

Abdel-Ouahab Boudraa, France
Francesco Braghin, Italy
Michael J. Brennan, UK
Maurizio Brocchini, Italy
Julien Bruchon, France
Javier Bulduf, Spain
Tito Busani, USA
Pierfrancesco Cacciola, UK
Salvatore Caddemi, Italy
Jose E. Capilla, Spain
Ana Carpio, Spain
Miguel E. Cerrolaza, Spain
Mohammed Chadli, France
Gregory Chagnon, France
Ching-Ter Chang, Taiwan
Michael J. Chappell, UK
Kacem Chehdi, France
Xinkai Chen, Japan
Chunlin Chen, China
Francisco Chicano, Spain
Hung-Yuan Chung, Taiwan
Joaquim Ciurana, Spain
John D. Clayton, USA
Carlo Cosentino, Italy
Paolo Crippa, Italy
Erik Cuevas, Mexico
Peter Dabnichki, Australia
Luca D'Acerno, Italy
Weizhong Dai, USA
Purushothaman Damodaran, USA
Farhang Daneshmand, Canada
Fabio De Angelis, Italy
Stefano de Miranda, Italy
Filippo de Monte, Italy
Xavier Delorme, France
Luca Deseri, USA
Yannis Dimakopoulos, Greece
Zhengtao Ding, UK
Ralph B. Dinwiddie, USA
Mohamed Djemai, France
Alexandre B. Dolgui, France
George S. Dulikravich, USA
Bogdan Dumitrescu, Finland
Horst Ecker, Austria
Karen Egiazarian, Finland

Ahmed El Hajjaji, France
Fouad Erchiqui, Canada
Anders Eriksson, Sweden
Giovanni Falsone, Italy
Hua Fan, China
Yann Favennec, France
Roberto Fedele, Italy
Giuseppe Fedele, Italy
Jacques Ferland, Canada
Jose R. Fernandez, Spain
S. D. Flapper, Netherlands
Thierry Floquet, France
Eric Florentin, France
Francesco Franco, Italy
Tomonari Furukawa, USA
Mohamed Gadala, Canada
Matteo Gaeta, Italy
Zoran Gajic, USA
Ciprian G. Gal, USA
Ugo Galvanetto, Italy
Akemi Gálvez, Spain
Rita Gamberini, Italy
Maria Gandarias, Spain
Arman Ganji, Canada
Zhong-Ke Gao, China
Xin-Lin Gao, USA
Giovanni Garcea, Italy
Fernando Garca, Spain
Laura Gardini, Italy
A. Gasparetto, Italy
Vincenzo Gattulli, Italy
J. Geiser, Germany
Oleg V. Gendelman, Israel
Mergen H. Ghayesh, Australia
Anna M. Gil-Lafuente, Spain
Hector Gómez, Spain
Rama S. R. Gorla, USA
Oded Gottlieb, Israel
Antoine Grall, France
Jason Gu, Canada
Quang Phuc Ha, Australia
Ofer Hadar, Israel
Masoud Hajarian, Iran
F. Hamelin, France
Zhen-Lai Han, China

Thomas Hanne, Switzerland
Takashi Hasuike, Japan
Xiao-Qiao He, China
María I. Herreros, Spain
Vincent Hilaire, France
Eckhard Hitzer, Japan
Jaromir Horacek, Czech Republic
Muneo Hori, Japan
A. Horvth, Italy
Gordon Huang, Canada
Sajid Hussain, Canada
Asier Ibeas, Spain
Giacomo Innocenti, Italy
Emilio Insfran, Spain
Nazrul Islam, USA
Payman Jalali, Finland
Reza Jazar, Australia
Khalide Jbilou, France
Linni Jian, China
Zhongping Jiang, USA
Bin Jiang, China
Ningde Jin, China
Grand R. Joldes, Australia
Joaquim J. Judice, Portugal
Tadeusz Kaczorek, Poland
Tamas Kalmar-Nagy, Hungary
Tomasz Kapitaniak, Poland
Haranath Kar, India
K. Karamanos, Belgium
Chaudry Khalique, South Africa
Nam-Il Kim, Korea
Do Wan Kim, Korea
Oleg Kirillov, Germany
Manfred Krafczyk, Germany
Frederic Kratz, France
Jurgen Kurths, Germany
K. Kyamakya, Austria
Davide La Torre, Italy
Risto Lahdelma, Finland
Hak-Keung Lam, UK
Antonino Laudani, Italy
Aime' Lay-Ekuakille, Italy
Marek Lefik, Poland
Yaguo Lei, China
Thibault Lemaire, France
Stefano Lenci, Italy
Roman Lewandowski, Poland
Qing Q. Liang, Australia
Panos Liatsis, UK
Wanquan Liu, Australia
Yan-Jun Liu, China
Peide Liu, China
Peter Liu, Taiwan
Jean J. Loiseau, France
Paolo Lonetti, Italy
L. M. López-Ochoa, Spain
V. C. Loukopoulos, Greece
Valentin Lychagin, Norway
Fazal Mahomed, South Africa
Yassir T. Makkawi, UK
Noureddine Manamanni, France
Didier Maquin, France
Paolo Maria Mariano, Italy
Benoit Marx, France
G. A. Maugin, France
Driss Mehdi, France
Roderick Melnik, Canada
Pasquale Memmolo, Italy
Xiangyu Meng, Canada
Jose Merodio, Spain
Luciano Mescia, Italy
Laurent Mevel, France
Yuri V. Mikhlin, Ukraine
Aki Mikkola, Finland
Hiroyuki Mino, Japan
Pablo Mira, Spain
Vito Mocella, Italy
Roberto Montanini, Italy
Gisele Mophou, France
Rafael Morales, Spain
Aziz Moukrim, France
Emiliano Mucchi, Italy
Domenico Mundo, Italy
Jose J. Muñoz, Spain
Giuseppe Muscolino, Italy
Marco Mussetta, Italy
Hakim Naceur, France
Hassane Naji, France
Dong Ngoduy, UK
Tatsushi Nishi, Japan
Ben T. Nohara, Japan
Mohammed Nouari, France
Mustapha Nourelfath, Canada
Sotiris K. Ntouyas, Greece
Roger Ohayon, France
Mitsuhiro Okayasu, Japan
Eva Onaindia, Spain
J. Ortega-Garcia, Spain
A. Ortega-Moñux, Spain
Naohisa Otsuka, Japan
Erika Ottaviano, Italy
Alkiviadis Paipetis, Greece
Alessandro Palmeri, UK
Anna Pandolfi, Italy
Elena Panteley, France
Manuel Pastor, Spain
P. N. Pathirana, Australia
F. Pellicano, Italy
Haipeng Peng, China
Mingshu Peng, China
Zhike Peng, China
Marzio Pennisi, Italy
Matjaz Perc, Slovenia
Francesco Pesavento, Italy
Maria do R. Pinho, Portugal
Antonina Pirrotta, Italy
Vicent Pla, Spain
Javier Plaza, Spain
J. Ponsart, France
Mauro Pontani, Italy
Stanislav Potapenko, Canada
Sergio Preidikman, USA
C. Pretty, New Zealand
Carsten Proppe, Germany
Luca Pugi, Italy
Yuming Qin, China
Dane Quinn, USA
Jose Ragot, France
K. R. Rajagopal, USA
Gianluca Ranzi, Australia
Sivaguru Ravindran, USA
Alessandro Reali, Italy
Giuseppe Rega, Italy
Oscar Reinoso, Spain
Nidhal Rezg, France
Ricardo Riaza, Spain
Gerasimos Rigatos, Greece
José Rodellar, Spain
R. Rodriguez-Lopez, Spain
Ignacio Rojas, Spain
Carla Roque, Portugal
Aline Roumy, France
Debasish Roy, India
R. Ruiz Garcia, Spain

Antonio Ruiz-Cortes, Spain
Ivan D. Rukhlenko, Australia
Mazen Saad, France
Kishin Sadarangani, Spain
Mehrdad Saif, Canada
Miguel A. Salido, Spain
Roque J. Saltarén, Spain
F. J. Salvador, Spain
A. Salvini, Italy
Maura Sandri, Italy
Miguel A. F. Sanjuan, Spain
Juan F. San-Juan, Spain
Roberta Santoro, Italy
Ilmar F. Santos, Denmark
J. A. Sanz-Herrera, Spain
Nickolas S. Sapidis, Greece
E. J. Sapountzakis, Greece
Themistoklis P. Sapsis, USA
Andrey V. Savkin, Australia
Valery Sbitnev, Russia
Thomas Schuster, Germany
Mohammed Seaid, UK
Lotfi Senhadji, France
J. Serra-Sagrasta, Spain
Leonid Shaikhnet, Ukraine
Hassan M. Shanechi, USA
Sanjay K. Sharma, India
Bo Shen, Germany
Babak Shotorban, USA
Zhan Shu, UK
Dan Simon, USA
Luciano Simoni, Italy
C. H. Skiadas, Greece
Michael Small, Australia

Francesco Soldovieri, Italy
Raffaele Solimene, Italy
Ruben Specogna, Italy
Sri Sridharan, USA
Ivanka Stamova, USA
Yakov Strelniker, Israel
S. A. Suslov, Australia
Thomas Svensson, Sweden
Andrzej Swierniak, Poland
Yang Tang, Germany
Sergio Teggi, Italy
Roger Temam, USA
Alexander Timokha, Norway
Rafael Toledo, Spain
Gisella Tomasini, Italy
F. Tornabene, Italy
Antonio Tornambe, Italy
Fernando Torres, Spain
Fabio Tramontana, Italy
S. Tremblay, Canada
I. N. Trendafilova, UK
George Tsiatas, Greece
Antonios Tsourdos, UK
Vladimir Turetsky, Israel
Mustafa Tutar, Spain
E. Tzirtzilakis, Greece
Filippo Ubertini, Italy
Francesco Ubertini, Italy
Hassan Ugail, UK
Giuseppe Vairo, Italy
K. Vajravelu, USA
R. A. Valente, Portugal
Raoul van Loon, UK
Pandian Vasant, Malaysia

M. E. Vázquez-Méndez, Spain
Josep Vehi, Spain
Kalyana C. Veluvolu, Korea
F. J. Verbeek, Netherlands
Franck J. Vernerey, USA
Georgios Veronis, USA
Anna Vila, Spain
R.-J. Villanueva, Spain
U. E. Vincent, UK
Mirko Viroli, Italy
Michael Vynnycky, Sweden
Junwu Wang, China
Shuming Wang, Singapore
Yan-Wu Wang, China
Yongqi Wang, Germany
J. Witteveen, Netherlands
Yuqiang Wu, China
Dash Desheng Wu, Canada
Xuejun Xie, China
Guangming Xie, China
Gen Qi Xu, China
Hang Xu, China
Xinggong Yan, UK
Luis J. Yebra, Spain
Peng-Yeng Yin, Taiwan
Ibrahim Zeid, USA
Qingling Zhang, China
Huaguang Zhang, China
Jian Guo Zhou, UK
Quanxin Zhu, China
Mustapha Zidi, France
Alessandro Zona, Italy

Contents

Swarm Intelligence in Engineering 2014, Baozhen Yao, Fang Zong, Bin Yu, and Rui Mu
Volume 2015, Article ID 858901, 4 pages

Short-Term Wind Speed Forecasting Using Support Vector Regression Optimized by Cuckoo Optimization Algorithm, Jianzhou Wang, Qingping Zhou, Haiyan Jiang, and Ru Hou
Volume 2015, Article ID 619178, 13 pages

Swarm Intelligence-Based Smart Energy Allocation Strategy for Charging Stations of Plug-In Hybrid Electric Vehicles, Imran Rahman, Pandian M. Vasant, Balbir Singh Mahinder Singh, and M. Abdullah-Al-Wadud
Volume 2015, Article ID 620425, 10 pages

Multipeak Mean Based Optimized Histogram Modification Framework Using Swarm Intelligence for Image Contrast Enhancement, P. Babu, V. Rajamani, and K. Balasubramanian
Volume 2015, Article ID 265723, 14 pages

Optimal Sizing of a Photovoltaic-Hydrogen Power System for HALE Aircraft by means of Particle Swarm Optimization, Victor M. Sanchez, Romeli Barbosa, J. C. Cruz, F. Chan, and J. Hernandez
Volume 2015, Article ID 183701, 8 pages

Hybrid Biogeography Based Optimization for Constrained Numerical and Engineering Optimization, Zengqiang Mi, Yikun Xu, Yang Yu, Tong Zhao, Bo Zhao, and Liqing Liu
Volume 2015, Article ID 423642, 15 pages

An Adapted Firefly Algorithm for Product Development Project Scheduling with Fuzzy Activity Duration, Minmei Huang, Jijun Yuan, and Jing Xiao
Volume 2015, Article ID 973291, 11 pages

Taxiing Route Scheduling between Taxiway and Runway in Hub Airport, Yu Jiang, Xinxing Xu, Honghai Zhang, and Yuxiao Luo
Volume 2015, Article ID 925139, 14 pages

Study of On-Ramp PI Controller Based on Dural Group QPSO with Different Well Centers Algorithm, Tao Wu, Xi Chen, and Yusong Yan
Volume 2015, Article ID 814871, 10 pages

The Inertia Weight Updating Strategies in Particle Swarm Optimisation Based on the Beta Distribution, Petr Maca and Pavel Pech
Volume 2015, Article ID 790465, 9 pages

Artificial Intelligence Mechanisms on Interactive Modified Simplex Method with Desirability Function for Optimising Surface Lapping Process, Pongchanun Luangpaiboon and Sitthikorn Duangkaew
Volume 2014, Article ID 864586, 16 pages

Swarm Intelligence-Based Hybrid Models for Short-Term Power Load Prediction, Jianzhou Wang, Shiqiang Jin, Shanshan Qin, and Haiyan Jiang
Volume 2014, Article ID 712417, 17 pages

A Modified Artificial Bee Colony Algorithm Based on Search Space Division and Disruptive Selection Strategy, Zhen-an He, Caiwen Ma, Xianhong Wang, Lei Li, Ying Wang, Yuan Zhao, and Huinan Guo
Volume 2014, Article ID 432654, 14 pages

PSO-Based Robot Path Planning for Multisurvivor Rescue in Limited Survival Time, N. Geng, D. W. Gong, and Y. Zhang
Volume 2014, Article ID 187370, 10 pages

Location Prediction-Based Data Dissemination Using Swarm Intelligence in Opportunistic Cognitive Networks, Jie Li, Xingwei Wang, Jie Jia, Pengfei Wang, Yan Zhou, and Zhijie Zhao
Volume 2014, Article ID 453564, 15 pages

Quantum Behaved Particle Swarm Optimization Algorithm Based on Artificial Fish Swarm, Dong Yumin and Zhao Li
Volume 2014, Article ID 592682, 10 pages

Research of Ant Colony Optimized Adaptive Control Strategy for Hybrid Electric Vehicle, Linhui Li, Haiyang Huang, Jing Lian, Baozhen Yao, Yafu Zhou, Jing Chang, and Ning'an Zheng
Volume 2014, Article ID 239130, 10 pages

Damage Identification of Bridge Based on Modal Flexibility and Neural Network Improved by Particle Swarm Optimization, Hanbing Liu, Gang Song, Yubo Jiao, Peng Zhang, and Xianqiang Wang
Volume 2014, Article ID 640925, 8 pages

A Swarm Random Walk Based Method for the Standard Cell Placement Problem, Najwa Altwaijry and Mohamed El Bachir Menai
Volume 2014, Article ID 461062, 11 pages

Core Business Selection Based on Ant Colony Clustering Algorithm, Yu Lan, Yan Bo, and Yao Baozhen
Volume 2014, Article ID 136753, 6 pages

A Framing Link Based Tabu Search Algorithm for Large-Scale Multidepot Vehicle Routing Problems, Xuhao Zhang, Shiquan Zhong, Yiliu Liu, and Xuelian Wang
Volume 2014, Article ID 152494, 13 pages

Displacement Prediction of Tunnel Surrounding Rock: A Comparison of Support Vector Machine and Artificial Neural Network, Qingdong Wu, Bo Yan, Chao Zhang, Lu Wang, Guobao Ning, and B. Yu
Volume 2014, Article ID 351496, 6 pages

Short-Term Power Generation Energy Forecasting Model for Small Hydropower Stations Using GA-SVM, Gang Li, Yongjun Sun, Yong He, Xiufeng Li, and Qiyu Tu
Volume 2014, Article ID 381387, 9 pages

Real-Time Arterial Coordination Control Based on Dynamic Intersection Turning Fractions Estimation Using Genetic Algorithm, Pengpeng Jiao, Honglin Wang, and Tuo Sun
Volume 2014, Article ID 430497, 10 pages

Hybrid Model for Early Onset Prediction of Driver Fatigue with Observable Cues, Mingheng Zhang, Gang Longhui, Zhe Wang, Xiaoming Xu, Baozhen Yao, and Liping Zhou
Volume 2014, Article ID 385716, 9 pages

Research of Obstacle Recognition Technology in Cross-Country Environment for Unmanned Ground Vehicle, Zhao Yibing, Xu Hongbin, Guo Lie, Li Linhui, and Zhang Mingheng
Volume 2014, Article ID 531681, 13 pages

Road Network Vulnerability Analysis Based on Improved Ant Colony Algorithm, Yunpeng Wang, Yuqin Feng, Wenxiang Li, William Case Fulcher, and Li Zhang
Volume 2014, Article ID 356963, 9 pages

Pattern Synthesis of Planar Nonuniform Circular Antenna Arrays Using a Chaotic Adaptive Invasive Weed Optimization Algorithm, Huaning Wu, Chao Liu, and Xu Xie
Volume 2014, Article ID 575860, 13 pages

Optimal Sensor Placement for Latticed Shell Structure Based on an Improved Particle Swarm Optimization Algorithm, Xun Zhang, Juelong Li, Jianchun Xing, Ping Wang, Qiliang Yang, Ronghao Wang, and Can He
Volume 2014, Article ID 743904, 12 pages

Forecasting Dry Bulk Freight Index with Improved SVM, Qianqian Han, Bo Yan, Guobao Ning, and B. Yu
Volume 2014, Article ID 460684, 12 pages

Applications of PCA and SVM-PSO Based Real-Time Face Recognition System, Ming-Yuan Shieh, Juing-Shian Chiou, Yu-Chia Hu, and Kuo-Yang Wang
Volume 2014, Article ID 530251, 12 pages

Binary Structuring Elements Decomposition Based on an Improved Recursive Dilation-Union Model and RSAPSO Method, Yudong Zhang, Shuihua Wang, Yi Sun, Genlin Ji, Preetha Phillips, and Zhengchao Dong
Volume 2014, Article ID 272496, 12 pages

Decision of Multimodal Transportation Scheme Based on Swarm Intelligence, Kai Lei, Xiaoning Zhu, Jianfei Hou, and Wencheng Huang
Volume 2014, Article ID 932832, 10 pages

Comparison of Three Different Curves Used in Path Planning Problems Based on Particle Swarm Optimizer, J. J. Liang, H. Song, B. Y. Qu, and Z. F. Liu
Volume 2014, Article ID 623156, 15 pages

Editorial

Swarm Intelligence in Engineering 2014

Baozhen Yao,¹ Fang Zong,² Bin Yu,³ and Rui Mu⁴

¹*School of Automotive Engineering, Dalian University of Technology, Dalian 116024, China*

²*College of Transportation, Jilin University, Changchun 130022, China*

³*Transportation Management College, Dalian Maritime University, Dalian 116026, China*

⁴*Delft University of Technology, NL-2628 BX Delft, Netherlands*

Correspondence should be addressed to Bin Yu; ybzhyb@163.com

Received 19 November 2014; Accepted 19 November 2014

Copyright © 2015 Baozhen Yao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Swarm intelligence (SI) is an artificial intelligence technique based on the study of cooperation behaviors of simple individuals (e.g., ant colonies, bird flocking, animal herding, and bees gathering honey) in various decentralized systems. The population, which consists of simple individuals, can usually solve complex tasks in nature by individuals interacting locally with one another and with their environments. Although a simple individual's behavior is primarily characterized by autonomy, distributed functioning, and self-organizing capacities, local interactions among the individuals often lead to a global optimal solution. Therefore, swarm intelligence is a promising way to develop powerful solution methods for optimization problems in engineering.

However, for large and complex engineering problems, SI algorithms often consume too much computation time due to the stochastic features of their searching approaches. For this reason, there is a potential requirement to develop efficient algorithms that are able to find solutions under limited resources, time, and money in real-world applications.

This special issue is aimed at highlighting the most significant recent developments on the topics of SI and to apply SI algorithms in real-life scenarios. Papers selected for this special issue present new findings and insights into this field. A broad range of topics are discussed, especially in the following areas: benchmarking and evaluation of new SI algorithms, convergence proof for SI algorithms, comparative theoretical and empirical studies on SI algorithms, and SI algorithms for real-world application.

Some works focus on the application of SI in the area of traffic and transportation. For example, in the paper entitled “Road network vulnerability analysis based on improved ant

colony algorithm,” Y. Wang et al. developed a novel approach based on an improved ant colony algorithm (IACA) to evaluate the vulnerability as well as identify the critical infrastructures of a real-world road network. They proved the high computational efficiency of the IACA-based approach when applied in large-scale networks and presented the dependence of road network vulnerability on the ration between traffic demand and road capacity.

In the paper titled “Hybrid model for early onset prediction of driver fatigue with observable cues,” M. Zhang et al. presented a hybrid model for the early onset prediction of driver fatigue. The hybrid model consists of three submodels which separately solve the three stages of the prediction problem. The support vector machine (SVM) is incorporated in the hybrid model for predicting the early onset driver fatigue state, while the genetic algorithm is used for optimizing parameters in the SVM. The approach has a good performance in the numerical test for the early onset prediction of driver fatigue.

In the paper titled “Real-time arterial coordination control based on dynamic intersection turning fractions estimation using genetic algorithm,” P. Jiao et al. established a real-time arterial coordination control model based on dynamic intersection turning fractions estimation. They developed an improved genetic algorithm to solve the dynamic intersection turning fractions estimation model, the solution of which was the inputs of the real-time arterial coordination control model. To solve the real-time arterial coordination control model, the authors proposed another improved genetic algorithm. Simulation experiment based on actual data proved the superiority of their approach.

In the paper entitled “*Study of on-ramp PI controller based on dural group QPSO with different well centers algorithm*,” T. Wu et al. proposed a new quantum behaved particle swarm optimization (QPSO) algorithm, which is the dual-group QPSO with different well centers (DWC-QOSO) algorithm. This novel algorithm could slow down the disappearance of population diversity and thus improve the global searching ability. The successful application of the DWC-QOSO algorithm in the on-ramp traffic control simulation indicated its powerful potential in the study of on-ramp PI controller.

In the paper entitled “*Decision of multimodal transportation scheme based on swarm intelligence*,” K. Lei et al. considered three elements of transportation costs, time, and risks when establishing a multimodal transportation scheme decision model. To solve the model, they proposed an improved SI algorithm combining particle swarm optimization (PSO) and ant colony (AC) algorithm. The numerical test showed that their algorithm inherited the advantages of both PSO and AC and had good ability to solve the decision problem of multimodal transportation scheme.

Also, the SI is discussed in the area of computer science. In the paper entitled “*Multipeak mean based optimized histogram modification framework using swarm intelligence for image contrast enhancement*,” P. Babu et al. proposed a PSO based approach to enhance the contrast and preserve essential details for any given image. They divided the histogram of an image into two subhistograms. The two subhistograms can be modified by optimal enhancement parameters which are found by the particle swarm optimization algorithm.

In the paper entitled “*Binary structuring elements decomposition based on an improved recursive dilation-union model and RSAPSO method*,” Y. Zhang et al. developed an improved dilation-union model for decomposing the structuring elements of any given shape, and they used the restarted simulated annealing particle swarm optimization as the algorithm of their model. Experiments showed that the proposed method outperformed most existing methods.

In the paper entitled “*Applications of PCA and SVM-PSO based real-time face recognition system*,” M.-Y. Shieh et al. used the support vector machine (SVM) which could be an optimal classifier of image recognition system to improve the validity of real-time face recognition systems. The particle swarm optimization is used by the authors to implement a feature selection, results of which are then input to the SVM for calculating the fitness value of classification.

There are also some works focusing on the application of SI in the operation research and management science. In the paper entitled “*Core business selection based on ant colony clustering algorithm*,” Y. Lan et al. proposed a method based on the ant colony clustering algorithm to identify the core business of an enterprise. By applying the method, the authors successfully found the core business of Tianjin Port from its ten main businesses.

In the paper entitled “*A framing link based tabu search algorithm for large-scale multidepot vehicle routing problems*,” X. Zhang et al. presented a framing link based Tabu search algorithm for solving a large-scale multidepot vehicle routing problem (LSMDVRP). Framing links are extracted from continuous great optimization of current solutions and updated

iteratively in the whole algorithm. The proposed algorithm was tested with 18 LSMDVRP instances and the results indicated that it had high computation speed and reliability.

In the paper entitled “*Forecasting dry bulk freight index with improved SVM*,” Q. Han et al. attempted to forecast the short-term trend of dry bulk freight index with an improved SVM model which incorporated the wavelet transform. As the dry bulk freight index is usually influenced by random events in the freight market, the wavelet transform is then combined to the SVM to denoise the index data series. The genetic algorithm is used to optimize parameters of the improved SVM. The prediction method displayed high accuracy in the experiment based on real data.

In the paper entitled “*Artificial intelligence mechanisms on interactive modified simplex method with desirability function for optimising surface lapping process*,” the authors applied the harmony search and firefly algorithms to optimize the influential parameters of surface lapping process of disk clamps.

In the paper entitled “*An adapted firefly algorithm for product development project scheduling with fuzzy activity duration*,” M. Huang et al. developed an adapted fuzzy firefly algorithm to solve a mathematical model which was built for scheduling new product development project with uncertain activity duration. The effectiveness and efficiency of the proposed algorithm were validated with benchmark experiments as well as a real-world example.

Besides those mentioned above, some authors present the application of SI in the electrical and electronic engineering. In the paper entitled “*Short-term power generation energy forecasting model for small hydropower stations using GA-SVM*,” G. Li et al. used a GA-based SVM model to predict short-term power generation energy of small hydropower and the numerical test proved the better performance of the GA-SVM forecasting model compared with the ARMA model.

In the paper entitled “*Short-term wind speed forecasting using support vector regression optimized by cuckoo optimization algorithm*,” J. Wang et al. proposed a hybrid forecasting model combining recurrence plot (RP) with support vector regression (SVR) to predict short-term wind speed series. In the hybrid model, the RP is used to analyze the wind speed series and select input variables for the SVR which is employed for short-term prediction, while the parameters of the SVR are optimized by three SI algorithms, respectively, say the GA, the PSO, and the cuckoo optimization algorithm (COA). The experimental results indicated that the forecasting model based on COA-SVR outperformed the other two models, especially in the context of jump samplings and multistep prediction.

In the paper entitled “*Swarm intelligence-based smart energy allocation strategy for charging stations of plug-in hybrid electric vehicles*,” I. Rahman et al. used the gravitational search algorithm (GSA) and the PSO to optimize the charging allocation strategy for extensive participation of plug-in hybrid electric vehicles, respectively, and analyzed the advantages as well as disadvantages of the two SI-based methods in the numerical experiment.

In the paper entitled “*Swarm intelligence-based hybrid models for short-term power load prediction*,” J. Wang et al. combined the singular spectrum analysis (SSA) with the seasonal autoregressive integrated moving average (SARIMA) and the SVR, respectively, and developed two improved forecasting models, that is, the SSA-SARIMA and the SSA-SVR, to implement short-term power load prediction. In order to optimize the parameters of the two hybrid models, the cuckoo search is employed. The experimental results showed that the application of CS could further improve the prediction accuracies of both the two models.

In the paper entitled “*A swarm random walk based method for the standard cell placement problem*,” the authors presented a new SI-based method, the swarm random walk method, to optimize the standard cell placement on large-size chip. According to the resulting placement in the benchmark experiment, the proposed method could be very competitive for solving the standard cell placement problem.

In the paper entitled “*Pattern synthesis of planar nonuniform circular antenna arrays using a chaotic adaptive invasive weed optimization algorithm*,” to optimize the nonuniform circular antenna arrays, H. Wu et al. revised the invasive weed optimization (IWO) by combining the chaotic search method into the IWO with adaptive dispersion and obtained a novel IWO called chaotic adaptive invasive weed optimization (CAIWO) which has better convergence speed as well as global searching ability.

There are several papers that apply SI in the civil engineering. In the paper entitled “*Displacement prediction of tunnel surrounding rock: a comparison of support vector machine and artificial neural network*,” Q. Wu et al. used two methods, that is, the SVM and artificial neural network (ANN), to separately predict tunnel surrounding rock displacement, and they found that the SVM-based method was more robust and accurate than the ANN-based method for the displacement prediction of tunnel surrounding rock.

In the paper entitled “*Damage identification of bridge based on modal flexibility and neural network improved by particle swarm optimization*,” H. Liu et al. developed a two-stage method to identify damage of bridge. In the first stage, modal flexibility indices are employed to localize damage and then an ANN whose parameters are optimized by PSO is used in the second stage to identify damage severity. The numerical simulation proved the feasibility and superiority of the proposed method.

In the paper entitled “*Optimal sensor placement for latticed shell structure based on an improved particle swarm optimization algorithm*,” X. Zhang et al. proposed a new improved particle swarm optimization (IPSO) algorithm to optimize the sensor placement in large-scale structures for the structural health monitoring. In the case study, they proved the feasibility of the IPSO-based method, as well as its superiority to other PSO algorithms in terms of convergence speed and precision.

Communication is another realm where SI can be applied. In the paper entitled “*Optimal sizing of a photovoltaic-hydrogen power system for HALE aircraft by means of particle swarm optimization*,” V. Sanchez et al. attempted to design a power supply system for high altitude long endurance

(HALE) aircrafts which could provide a wide range of telecommunication services. The main problem they met resided in the size optimization for the system, and they used the PSO to solve the optimal sizing problem. Case study showed that the power supply systems calculated by PSO could work efficiently and steadily for long time flights.

In the paper entitled “*Location prediction-based data dissemination using swarm intelligence in opportunistic cognitive networks*,” J. Li et al. applied the ant colony optimization to address the data dissemination problem in the opportunistic cognitive networks. Real-world simulation indicated that the routing scheme for conveying message optimized by the ant colony algorithm had better performances.

There are also two papers discussing the SI in automotive engineering. In the paper entitled “*Research of ant colony optimized adaptive control strategy for hybrid electric vehicle*,” L. Li et al. develop an ant colony optimization based method to dynamically determine energy management control strategy of hybrid electric vehicles (HEVs) in different driving cycles. Based on a certain type of driving cycle identified by a fuzzy driving cycle recognition algorithm, the ant colony optimization algorithm is used to optimize the control parameters in the corresponding context. The optimization method is validated in the simulation experiments.

In the paper entitled “*Research of obstacle recognition technology in cross-country environment for unmanned ground vehicle*,” Z. Yibing et al. presented a multistep approach incorporating filtering algorithm and multifeature fusion algorithm based on Bayes classification theory to solve the obstacle recognition problem of unmanned ground vehicles in cross-country environment. The robustness and accuracy of the algorithm are well proved in the simulation test.

Some works use the SI in robotics. In the paper entitled “*PSO-based robot path planning for multisurvivor rescue in limited survival time*,” N. Geng et al. studied the optimal rescue path planning of a robot when using it in urgent and dangerous circumstances. To solve the problem, the authors developed a novel improved PSO incorporating particle updating, insertion and inversion operators, and a rapid local search method. The simulation results presented the feasibility and efficiency of the algorithm in finding optimal paths.

The robot path planning problem is also attempted by J. J. Liang et al. In their paper entitled “*Comparison of three different curves used in path planning problems based on particle swarm optimizer*,” they employed the dynamic multiswarm particle swarm optimizer to search the optimal parameters of three commonly used curves which were usually applied in robot path planning problems.

The following four papers mainly focus on enhancing the representative SI algorithms. In the paper entitled “*Hybrid biogeography based optimization for constrained numerical and engineering optimization*,” Z. Mi et al. proposed a new hybrid biogeography based optimization (HBBO) algorithm which contained chaotic search and a new mutation operator combining differential evolution mutation operator with simulated binary crossover of genetic algorithm. To achieve global optimal, the differential evolution mutation operator is still used in the HBBO to provide an update on half

population. The HBBO was tested with 12 benchmark and four engineering optimization problems, and results showed that the HBBO outperformed other evolutionary algorithms, especially for constrained optimization problems.

In the paper entitled “*A modified artificial bee colony algorithm based on search space division and disruptive selection strategy*,” Z. He et al. improved the artificial bee colony (ABC) algorithm from three aspects. They proposed a new initialization method to generate high quality initial solutions, a disruptive selection strategy to enhance population diversity, and a novel definition of the scout bee phase.

In the paper entitled “*Quantum behaved particle swarm optimization algorithm based on artificial fish swarm*,” the authors introduced adaptive parameters and swarm and follow activities to the existing quantum behaved particle swarm optimization (QPSO) algorithm and obtained a new QPSO called quantum particle swarm optimization algorithm based on artificial fish swarm, which was validated to have better optimization ability than QPSO.

In the paper entitled “*The inertia weight updating strategies in particle swarm optimisation based on the beta distribution*,” the authors compared the performances of different PSOs under different selected random updating strategies of inertia weight. By using 28 benchmark functions, they found that the multiswarm PSO combining two updating strategies of inertia weight was the best.

Acknowledgments

These articles present rich and valuable advancements that SI technologies have made for solving problems in engineering. We would like to thank all the authors for their excellent work and contributions to this special issue. We would also like to express our gratitude to all the reviewers for their fundamental work and patience in assisting us.

*Baozhen Yao
Fang Zong
Bin Yu
Rui Mu*

Research Article

Short-Term Wind Speed Forecasting Using Support Vector Regression Optimized by Cuckoo Optimization Algorithm

Jianzhou Wang,¹ Qingping Zhou,² Haiyan Jiang,² and Ru Hou²

¹ School of Statistics, Dongbei University of Finance and Economics, Dalian 116025, China

² School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, China

Correspondence should be addressed to Qingping Zhou; zhouqp12@lzu.edu.cn

Received 6 June 2014; Revised 5 September 2014; Accepted 16 September 2014

Academic Editor: Fang Zong

Copyright © 2015 Jianzhou Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper develops an effectively intelligent model to forecast short-term wind speed series. A hybrid forecasting technique is proposed based on recurrence plot (RP) and optimized support vector regression (SVR). Wind caused by the interaction of meteorological systems makes itself extremely unsteady and difficult to forecast. To understand the wind system, the wind speed series is analyzed using RP. Then, the SVR model is employed to forecast wind speed, in which the input variables are selected by RP, and two crucial parameters, including the penalties factor and gamma of the kernel function RBF, are optimized by various optimization algorithms. Those optimized algorithms are genetic algorithm (GA), particle swarm optimization algorithm (PSO), and cuckoo optimization algorithm (COA). Finally, the optimized SVR models, including COA-SVR, PSO-SVR, and GA-SVR, are evaluated based on some criteria and a hypothesis test. The experimental results show that (1) analysis of RP reveals that wind speed has short-term predictability on a short-term time scale, (2) the performance of the COA-SVR model is superior to that of the PSO-SVR and GA-SVR methods, especially for the jumping samplings, and (3) the COA-SVR method is statistically robust in multi-step-ahead prediction and can be applied to practical wind farm applications.

1. Introduction

Energy is an indispensable ingredient of economic growth, but rapid global economic growth has contributed to the ever-increasing demand for energy. However, traditional energy sources, such as coal, oil, and natural gas, not only are facing severe depletion, but also produce various forms of air pollution, which have adverse human health effects [1]. Thus, there is an urgent need to develop renewable resources that aid governments in reducing their reliance on fossil fuels, simultaneously decreasing carbon emissions and improving air quality. Hence, renewable energy sources such as wind power, hydropower, solar energy, geothermal energy, and biofuels have gained more and more attention in recent years [2]. Among these renewable resources, wind power has the characteristics of a mature technique and is relatively cost competitive. Wind power has naturally become the fastest growing energy resource around the world. According to the Global Wind Energy Council (GWEC), the global cumulative installed capacity reached 282.5 GW in 2012 [3]. Moreover,

approximately 10 million MW of wind energy is continuously installed every year. By 2020, 12 percent of the world's power could feasibly come from wind energy. The Chinese wind market shared approximately 27 percent of the global wind energy in 2012. The wind-generated electricity in China reached 100.4 billion kWh in 2012, accounting for 2% of the total electricity consumption.

However, high wind power penetration requires addressing many issues, such as balancing power production and consumption, interconnection standards, and power system stability and reliability [4]. The power grid dispatchers do face a significant challenge due to the increased integration of large scale wind power into existing power systems. To address these challenges, dispatchers must forecast wind power production for wind turbine control, preload sharing, and power system management, as well as for determining the power reserves [5]. The relationship between wind speed and wind power is basically cubic; therefore, a small error in the wind speed forecast will lead to a larger error in wind

power production. Thus, the accurate forecast of short-term wind speed prediction plays a crucial role in wind power forecasting [6].

A number of methods have been developed to forecast wind speed. These methods are usually classified into four categories: (a) physical modeling methods, (b) time series models, (c) artificial intelligence models, and (d) hybrid models. Physical models, such as numerical weather prediction, require various weather data, such as temperature, pressure, and topography information, and solve complex mathematical equations. However, physical modeling approaches are not effective for very short-term predictions. In the time series forecasting methods, mathematical equations are used to predict wind speed based on historical data. The commonly used time series models are the autoregressive moving average model (ARMA), autoregressive integrated moving average model (ARIMA), fractional ARIMA, and seasonal ARIMA [7–9]. These time series methods perform well in situations when the data set has low frequency in nature, such as weekly/yearly patterns, but have difficulties with high-frequency changes and rapid variations of the original data.

With the capability to track complex nonlinearity systems, artificial intelligence techniques have received considerable attention in the wind energy field. An artificial intelligence technique is a generalized term that encompasses artificial neural networks (ANNs) [10, 11], fuzzy logic methods [12, 13], and support vector machines (SVM) [6, 14–21]. Mohandes et al. [16] used the SVM model to forecast the mean daily wind speed and compared the results with the multilayer perceptron (MLP) neural networks. They concluded that the SVM model outperformed the MLP forecasting method. Similar conclusions are found elsewhere [18, 19]. Hybrid forecasting methods, which involve the mixture of several methods or algorithms, have also been proven to obtain better performance. Zhou et al. [21] presented a hybrid method for wind speed prediction based on wavelet analysis and support vector machine. The experimental results show that this method is capable of improving forecasting precision and generalization performance. Salcedo-Sanz et al. [17] successfully used SVM to forecast wind speed at a Spanish wind farm, in which the hyperparameters are tuned by two evolutionary algorithms including the evolutionary programming algorithm (EP) and particle swarm optimization approach (PSO). Because an abundance of redundant information in the wind speed series would decrease the forecasting validity and precision, Cheng and Guo [14] proposed a hybrid forecasting method based on SVM and information granulation. The original data were refined with fuzzy information granulation, and then the refined data are forecasted by the SVM model. Liu et al. [15] presented a hybrid model based on wavelet transform (WT), genetic algorithm (GA), and SVM. The parameters in SVM were chosen by GA. A case study demonstrates that the hybrid method outperforms the comparison models. A hybrid approach [22] based on the ensemble empirical mode decomposition (EEMD) and SVM was proposed to forecast the mean hourly wind speed, which proved to be effective in improving the prediction accuracy.

Wind speed forecasting based on weather data encounters some difficulties. The accurate measurement of weather data is commonly expensive and unavailable. On the other hand, inaccurate measurements would introduce information to the forecasting model [6, 22, 23]. As a dynamic system, wind speed has a relationship with its past values at any time [24]. Therefore, achieving a low wind speed forecasting error through a relatively simple forecasting model with the historical wind speed data is desired.

Based on the above discussion, this paper proposed a hybrid wind speed forecasting model based on RP and optimized SVM, using its historical data. RP is used to analyze the wind system and to extract the useful information from the wind speed data; the optimized SVM model is then operated to obtain accurate hourly wind speeds. An actual case study is tested in the Shandong province in China.

This paper is organized as follows. Section 2 describes the framework of the study. Section 3 demonstrates the fundamentals of the methods, including the recurrence plot, support vector regression model, and some optimized algorithms. Section 4 presents some chosen performance criteria. Section 5 presents an experimental study to evaluate the proposed models, and a discussion and some conclusions are drawn in Section 6.

2. Framework of the Hybrid Model

The framework of the hybrid model is described in Figure 1. The detailed processes are given as follows.

- (1) Apply the RP to analyze the predictability of the original wind speed data, which forms the prerequisite conditions for accurate forecasting. Moreover, RP can also extract the optimal information from the original data that is used as the input variables of the optimized SVR models.
- (2) The SVR models, in which parameters are optimized by GA, PSO, and COA, are established for multi-step-ahead wind speed prediction.
- (3) Compare the forecasts and the actual wind speed. Meanwhile, a two-tailed hypothesis test is performed to evaluate the stability of the forecasting methods.

3. Methodology

In this section, the fundamentals of the recurrence plot (RP) and support vector regression (SVR) are introduced. To obtain the best performance of the SVR method, three optimized algorithms, including genetic algorithm (GA), particle swarm optimization (PSO), and cuckoo optimization algorithm (COA), are used to select the optimal parameters of SVR.

3.1. Recurrence Plot. Recurrence plot (RP) can identify whether the wind speed series are generated by a nonlinear deterministic process and select the input set of SVR. Recurrence is a basic property of the dynamic system. This property can be visualized and analyzed using the powerful tool of recurrence plot (RP). Eckmann et al. [25] first introduced

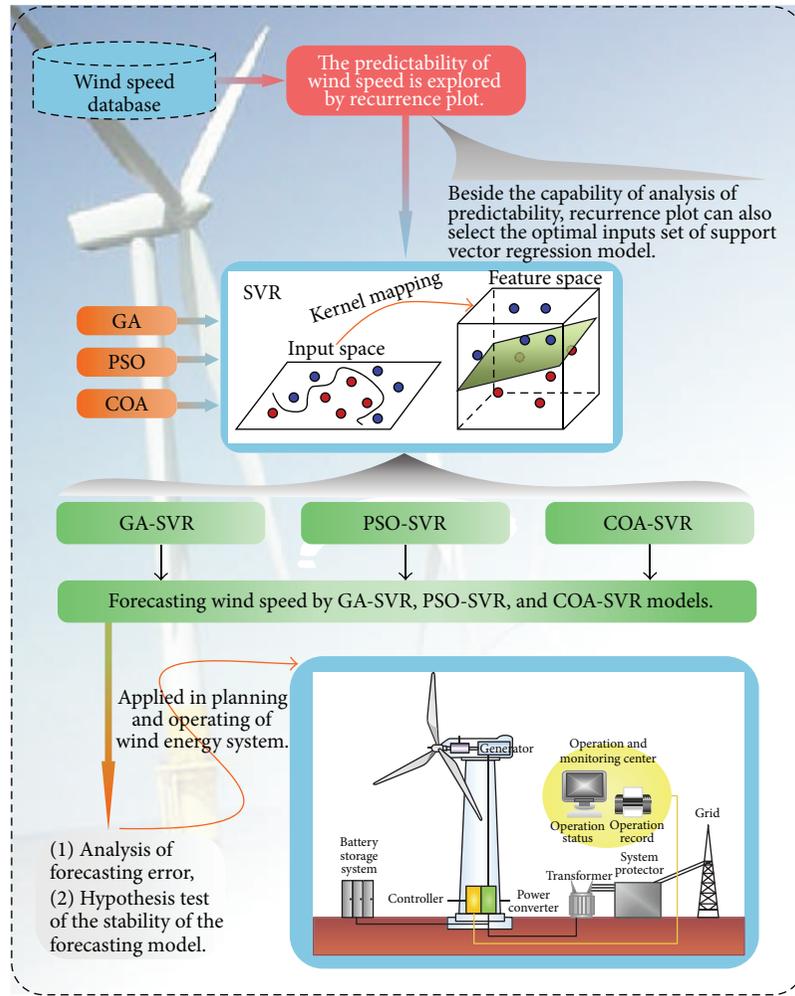


FIGURE 1: Framework of the hybrid method.

RP and used it to detect nonlinearities and chaotic dynamics in experimental signals in physics. The RP technique has been used recently to identify structural changes and hidden patterns in data or detect similarities in patterns across the time series in many fields, such as economics [26], physiology [27], and energy markets [27]. In this paper, the RP technique will be applied to analyze the predictability of the wind time series.

In the process of producing an RP, the original one-dimensional data are first mapped to a higher dimensional reconstructed phase space. Letting $\{x_i\}_1^N$ be the wind speed time series, the reconstructed vector in the phase space can be represented as

$$\mathbf{X}_i = (x_i, x_{i+1}, \dots, x_{i+(m-1)\tau}), \quad i = 1, 2, \dots, N - (m-1)\tau, \quad (1)$$

where m is the embedding dimension and τ is the embedding delay.

Recurrence is defined as the degree of closeness between two different elements in the phase space. RP visualizes the

behavior of recurrence via a graphical representation of the distance matrix:

$$R_{i,j} = \Theta(\varepsilon - \|\mathbf{X}_i - \mathbf{X}_j\|), \quad i, j = 1, 2, \dots, N - (m-1)\tau, \quad (2)$$

where ε is a predefined threshold, $\|\cdot\|$ is the usual Euclidean norm, and $\Theta(\cdot)$ is the Heaviside function. RP is the graphic representative of $R_{i,j}$. RP can identify three types of dynamic systems, as shown in Figure 2. (1) Periodic systems are characterized by diagonal parallel lines with the same periodic distance. Data with this RP indicates that the data are periodic or quasiperiodic. (2) Chaotic systems are characterized by diametric lines with the irregular distances which have been cut. Data with this RP reveals that the data often has abrupt changes. (3) Stochastic systems are characterized by many individual dots that indicate their distribution is quite irregular. An example of such an RP is that of a random time series.

3.2. Support Vector Regression Model. The support vector machine (SVM), proposed by Vapnik and his coworkers [28], has been successfully used for pattern recognition,

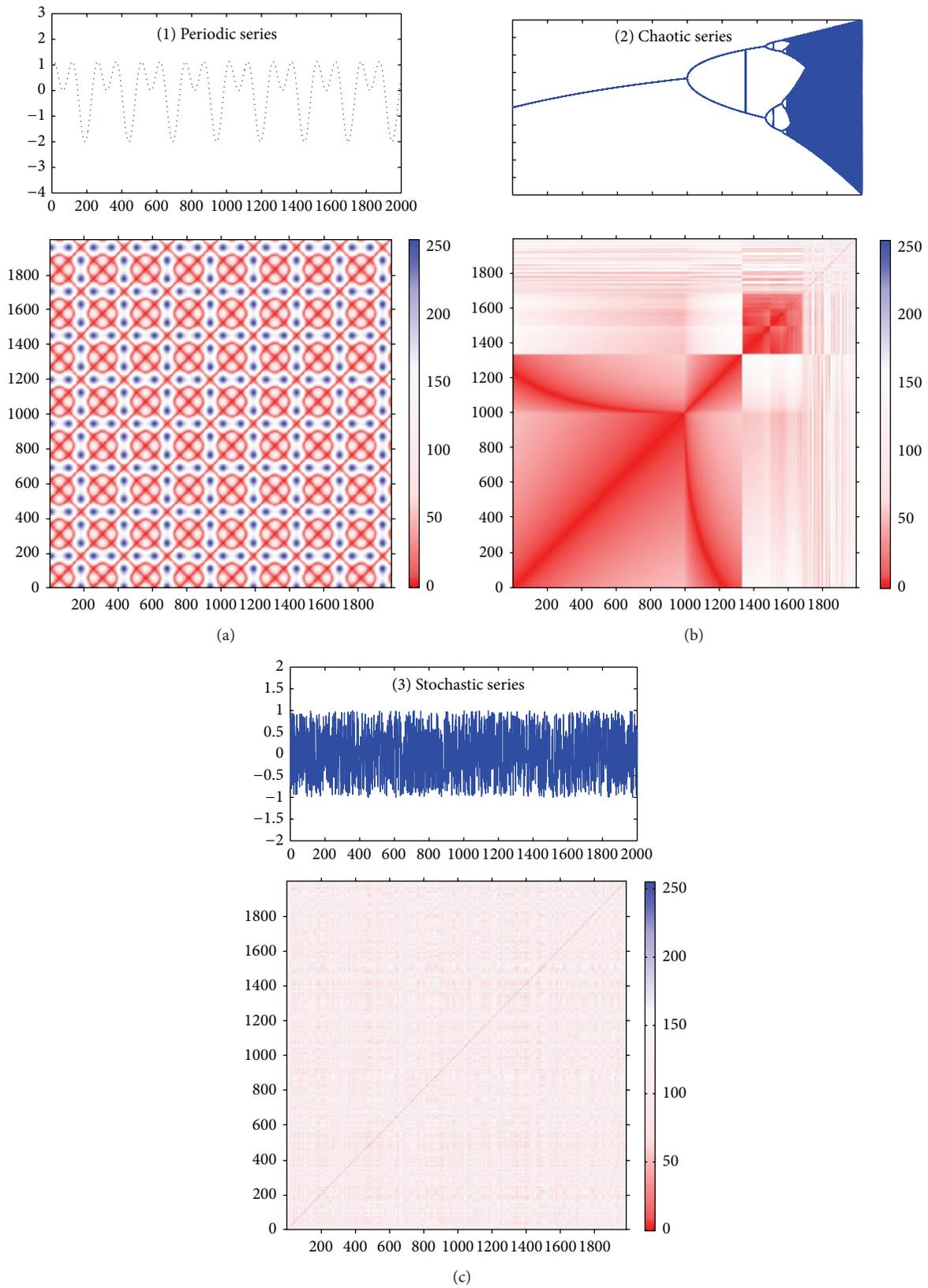


FIGURE 2: Recurrence plots of periodic series, chaotic series, and stochastic series.

classification, and regression. SVM is based on the statistical learning theory, the Vapnik-Chervonenkis dimension theory, and the structural risk minimization (SRM) principle. Support vector regression (SVR), also developed by Vapnik and coworkers [29], is an extension of SVM. Compared with SVM, SVR minimizes the generalized error bound instead of minimizing the observed training error [30].

To gain the optimal generalization ability, the basic principle of SVR is to map the original data into a high-dimensional feature space through nonlinear mapping [31]. Supposing the training data are $\{(x_1, y_1), \dots, (x_n, y_n)\}$, the regression formula can be expressed as

$$f(x) = \sum_{i=1}^D \omega_i \Phi_i(x) + b, \quad \Phi_i: R^n \rightarrow F, \quad \omega_i \in F, \quad b \in R. \quad (3)$$

Here $\{\omega_i\}_{i=1}^D$ are weights estimated from the training data, b is the threshold value, and $\{\Phi_i\}_{i=1}^D$ are nonlinear mapping functions which map the sample datasets to high-dimensional feature space F . Based on the SRM principle, the weights $\{\omega_i\}_{i=1}^D$ can be obtained from the sample data by minimizing the following quadratic programming problem:

$$\min \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*), \quad (4)$$

subject to

$$|y_i - \omega \cdot \Phi(x_i) - b| \leq \varepsilon + \xi_i, \quad \xi_i, \xi_i^* \geq 0, \quad (5)$$

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

where constant C , also called the penalties factor, is greater than zero and determines the tradeoff by minimizing the training error and minimizing the model complexity. ξ_i and ξ_i^* are the slack variables. $\varepsilon(\cdot)$ is the ε -intensive loss function and is defined as follows:

$$\varepsilon(y_i) = \begin{cases} 0, & |f(x_i) - y_i| < \varepsilon, \\ |f(x_i) - y_i| - \varepsilon, & \text{otherwise.} \end{cases} \quad (6)$$

By solving the optimization problem, the estimation function can be obtained as follows:

$$f(x, a, \alpha^*) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(x_i, x_j) + b, \quad (7)$$

subject to

$$\sum_{i=1}^n (\alpha_i - \alpha_i^*) = 0, \quad 0 \leq \alpha_i, \quad \alpha_i^* \leq C, \quad (8)$$

where $K(x_i, x_j)$ is a kernel function that represents the inner product in the D -dimensional feature space. SVR is characterized by the use of "kernel trick" to apply linear classification techniques to nonlinear classification problems. Consider

$$K(x_i, x_j) = \sum_{i=1}^D \Phi(x_i) \cdot \Phi(x_j). \quad (9)$$

The typical kernel functions include Gaussian Radial Basis Functions (RBF), polynomial kernel functions, Gaussian kernel functions, and sigmoid kernel functions. In this paper, the RBF kernel function is used due to its computational simplicity and its robustness in dealing with nonlinear data. The RBF is defined mathematically as

$$K(x_i, x_j) = e^{(-\|x_i - x_j\|^2 / \delta^2)}, \quad (10)$$

where δ is the width of RBF.

3.3. Optimized Algorithms. Evolutionary algorithms and heuristic optimization algorithms have played a significant part in addressing practical mathematics and engineering problems in recent years [6, 8]. Some optimization algorithms used in this study are described as follows.

3.3.1. Genetic Algorithm. The genetic algorithm (GA) proposed by Holland [32] is inspired by the living natural evolution procedure. GA is a global optimization method that mimics the biological evolution mechanisms, including survival of the fittest, crossover, and mutation. It performs well in complex optimization problems given its simplicity and robustness, and it has been successfully applied in various forecasting fields. To effectively search for the optimal solution, attention should be focused on the selection of population size, crossover rate, and mutation rate. More details of GA can be found elsewhere [33]. The basic optimization procedure is described as follows.

Step 1. Randomly initialize a group of chromosomes. The chromosomes correspond to the solutions in an optimization problem.

Step 2. Evaluate the fitness of the candidate chromosomes. The chromosomes with smaller fitness are more likely to survive.

Step 3. Produce a new generation according to the proportionate. The chromosomes with higher fitness are more likely to be selected.

Step 4. Perform a crossover or mutation operation to generate new candidate chromosomes.

Step 5. Return the solution if the optimal solution has been achieved or return to Step 2 until a certain number of iterations has been reached.

3.3.2. Particle Swarm Optimization Algorithm. PSO, first proposed by Kennedy and Eberhart [34], is an efficient evolutionary computation technique. In the PSO algorithm, a swarm of particles keep updating around a search space, and an optimal solution will emerge at the end. Suppose the search space is m -dimensional space and the number of particles is n ; then the i th particle can be defined as an m -dimensional vector $X_i = (x_{i1}, x_{i2}, \dots, x_{im})$ ($i = 1, 2, \dots, n$) and the update velocity of this particle is $V_i = (v_{i1}, v_{i2}, \dots, v_{im})$ ($i = 1, 2, \dots, n$). The mechanism of the algorithm is described as follows.

Step 1. Randomly initialize a population of n particles with positions and velocities. The positions represent the candidate solutions in the optimized problems.

Step 2. Evaluate the fitness value for each particle.

Step 3. Calculate the position of the best fitness value for each particle from its historical movement. The best position of the i th particle is denoted as $P_i = (p_{i1}, p_{i2}, \dots, p_{im})$.

Step 4. Evaluate the position of the best fitness value for all particles. The best position for all particles is expressed as $P_g = (P_{g1}, P_{g2}, \dots, P_{gn})$.

Step 5. Update the speed and the position of particles using the following:

$$v_i^{k+1} = \omega * v_i^k + c_1 * r_1 * (p_i^k - x_i^k) + c_2 * r_2 * (p_g - x_i^k), \quad (11)$$

$$x_i^{k+1} = x_i^k + v_i^{k+1}, \quad (12)$$

where ω is the inertia weight, r_1 and r_2 are two uniformly distributed random variables in the interval from 0 to 1, and c_1 and c_2 are the personal learning and global learning coefficients, respectively. The first term on the right side of (11) makes the particles move around the search space; the second term makes the particles stay close to their personal best positions, and the last term, known as the social collaboration, has the advantage of quickly approaching the global optimal value.

Step 6. If the terminated condition is not satisfied, return to Step 2. The terminated condition is usually a predefined iteration value or fitness value.

3.3.3. Cuckoo Optimization Algorithm. The cuckoo optimization algorithm (COA) is a metaheuristic optimization method first proposed by Yang and Deb in 2009 [35]. The basic idea of this new algorithm was based on the obligate brood parasitic behavior of the cuckoo and its method in egg laying and breeding. In nature, many animals search for food in a quasirandom manner. In fact, an animal's path of searching for food is effectively a random walk because the next move is based on the current location and the transition probability to the next location. Choosing the direction depends on a probability which can be modeled mathematically. A recent study presents that *Drosophila melanogaster*, or fruit flies, wander in their territory using a series of straight flight paths punctuated by sudden 90° turns, leading to a Levy flight-style intermittent scale-free search pattern.

The basic COA is defined by the endeavor to survive among cuckoos. In the survival competition, the cuckoo makes two main operations, including a random search based on the probability of finding an alien egg of a host bird and a direct search based on Levy flights. With the combination of the two operations, COA is more effective than other optimization methods for nonconvex and complex optimization problems [36]. In the COA method, each nest represents a candidate solution and a number of nests are used for finding

the appropriate solution of the optimization problem. The main steps of the COA method are described as follows.

Step 1. Initialize the population. Suppose the number of host nests is N_p and the optimization problem has N parameters. Then, a population of N_p nests is denoted as $X_i = [P_{i1}, P_{i2}, \dots, P_{iN}]$ ($i = 1, 2, \dots, N_p$).

Step 2. Update the new solution via Levy flights. The best path of the Levy flights is estimated by Mantegna's algorithm [37]. The new solution of each nest is calculated using the following:

$$X_i^{\text{new}} = X\text{best}_i + \alpha * \text{rand}_1 * \Delta X_i^{\text{new}},$$

$$\Delta X_i^{\text{new}} = \nu * \frac{\sigma_x(\beta)}{\sigma_y(\beta)} * (X\text{best}_i - G\text{best}), \quad (13)$$

where $X\text{best}_i$ is the best solution for the individual, $G\text{best}$ is the best solution for the total population, $\alpha > 0$ is the iteration step size, and rand_1 is a uniformly distributed random number in the interval from 0 to 1. The variable ν is defined as $\text{rand}_x/\text{rand}_y$, rand_x , and rand_y , having normal distributions with the standard deviations $\sigma_x(\beta)$ and $\sigma_y(\beta)$ given as follows:

$$\sigma_x(\beta) = \left[\Gamma(1 + \beta) * \frac{\sin(\pi\beta/2)}{\Gamma((1 + \beta)/2)} * \beta * 2^{((\beta-1)/2)} \right]^{1/\beta},$$

$$\sigma_y(\beta) = 1, \quad (14)$$

where β is the distribution factor between 0.3 and 1.99 and $\Gamma(\cdot)$ is the gamma function.

Step 3. Discover the alien egg and randomize the nests. Suppose the probability of a host bird to discover an alien egg in its nest is p_a . The action of this discovery also creates a new solution, which is defined as the following:

$$X_i^{\text{dis}} = X\text{best}_i + K * \Delta X_i^{\text{dis}},$$

$$\Delta X_i^{\text{dis}} = \text{rand}_2 * [\text{rand}_{p_1}(X\text{best}_i) - \text{rand}_{p_2}(X\text{best}_i)], \quad (15)$$

where rand_2 is a uniformly distributed random number and ranges from 0 to 1. The variables $\text{rand}_{p_1}(X\text{best}_i)$ and $\text{rand}_{p_2}(X\text{best}_i)$ are the random perturbations of nest positions in $X\text{best}_i$. Parameter K is determined based on p_a . Consider

$$K = \begin{cases} 1, & \text{if } \text{rand}_2 < p_a, \\ 0, & \text{otherwise.} \end{cases} \quad (16)$$

Step 4. Stop the algorithm when the maximum number of iterations is reached.

4. Model Performance Evaluation

To evaluate the performance of the optimized SVR models, MAE, RMSE, MAPE, R^2 , and IA (see (17) through (21)) are used to measure the forecasting accuracy. MAE and RMSE

are employed to measure the absolute error of the actual data and the forecasts. Including the quadratic term in the RMSE makes it more sensitive to excessive values than MAE. Moreover, the relative error is also evaluated using MAPE. R^2 and IA measure the overall similarity between the actual and the forecasted wind speed. They both range from 0 to 1. For a perfect forecasting model, MAE, RMSE, and MAPE take values of 0, and the values of R^2 and IA are equal to 1. Let A_t and P_t be the actual and the forecasting values at time-point t , respectively; \bar{A} and \bar{P} denote the mean of the actual and the forecasting values, respectively. Then, these performance metrics are defined as follows.

(i) Mean absolute error (MAE):

$$\text{MAE} = \frac{1}{N} \sum_{t=1}^N |A_t - P_t|. \quad (17)$$

(ii) Root mean square error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{t=1}^N (A_t - P_t)^2}. \quad (18)$$

(iii) Mean absolute percentage error (MAPE):

$$\text{MAPE} = \frac{1}{N} \sum_{t=1}^N \frac{|A_t - P_t|}{|A_t|} \times 100\%. \quad (19)$$

(iv) Coefficient of determination (R^2):

$$R^2 = \frac{[\sum_{t=1}^N (P_t - \bar{P})(A_t - \bar{A})]^2}{\sum_{t=1}^N (P_t - \bar{P})^2 \sum_{t=1}^N (A_t - \bar{A})^2}. \quad (20)$$

(v) Index of agreement (IA):

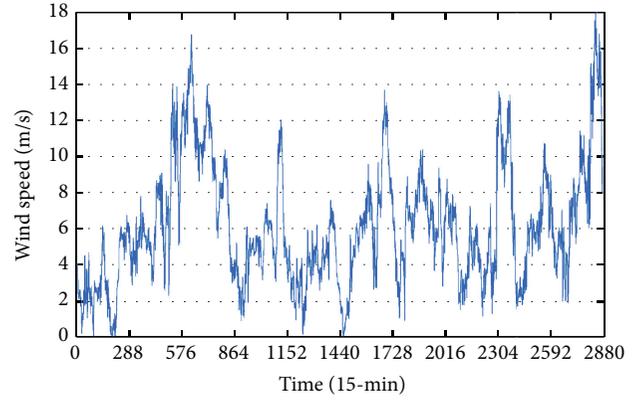
$$\text{IA} = 1 - \frac{\sum_{t=1}^N (P_t - A_t)^2}{\sum_{t=1}^N (|P_t - \bar{A}| + |A_t - \bar{A}|)^2}. \quad (21)$$

5. Case Studies

The hybrid methods are applied to wind speed prediction at the Shandong wind farm in China. The prediction horizon is 15 min (one-step-ahead), 30 min (two-step-ahead), 45 min (three-step-ahead), and 1 h ahead (four-step-ahead). In this paper, the following forecasting approaches are performed:

- (i) forecasting based on SVR and COA, that is, the COA-SVR method,
- (ii) forecasting based on SVR and PSO, that is, the PSO-SVR method,
- (iii) forecasting based on SVR and GA, that is, the GA-SVR method.

5.1. Available Data. The wind speed data used in this paper are collected from a Shandong wind farm in China. The studied time range covers September 23rd to October 22nd,



The highest wind speed is 17.93 m/s.
The lowest wind speed is 0 m/s.
The mean wind speed is 6.02 m/s.
The standard deviation value is 3.29 m/s,

FIGURE 3: Original wind speed time series.

2012. The data are sampled in 15 min time intervals, so there are 96 data points for one day. The total dataset includes 2880 wind speed samples; the first 2592 samples are used to build the models, and the remaining 288 samples will be used to validate the forecasting performance. Figure 3 shows the variation trend of the wind speed data. Shandong province had a total installed capacity of 6980.5 MW in 2013, ranking fourth in China. Given the considerable wind power potential around the coastal areas and islands as well as those wind farms located in electricity demand centers [38], Shandong wind farms have received more and more attention in China. Thus, accurate wind speed forecasting is crucial for planning and operating a power system for the city's sustainable development.

5.2. Predictability of Wind Speed. Figure 2 shows severe fluctuations in the wind speed data, while no hallmark of periodicity is demonstrated. However, previous researchers have concluded that the fluctuation comes from the stochastic or chaotic nature of a nonlinear system [39, 40]. To study the predictability of wind speed, RP is performed on the original wind speed data. Based on the derived results in Section 5.3, a correlation analysis is performed to choose proper input sets.

The embedding dimension and the embedding delay of the wind speed series must be acquired first to reconstruct the phase space. The false nearest neighbor method and the mutual information method are used to calculate these two parameters. The embedding dimension and the embedding delay are 3 and 2, respectively. The RP of wind speed time series is shown in Figure 4 [41]. It can be concluded that the short-term erratic distribution of recurrence points is representative of the strong stochastic nature of the wind speed series with mimic predictability. White ribbons in the RP correspond to transitive processes in the system. The dynamic transition process is representative of both the seasonality and no stationary of the wind speed time series.

5.3. Selection of SVR Inputs. It is concluded above that the wind speed has mimic predictability. Therefore, the mimic

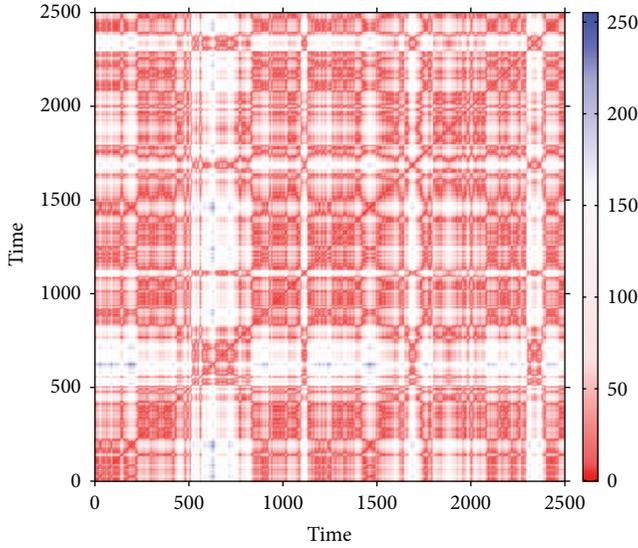


FIGURE 4: Recurrence plot of the original wind speed.

predictability of the wind speed series should be taken into account in selecting the inputs for the SVR model. Moreover, because the input set of the forecasting model is the lagged wind speed, it should be as close as possible to the forecasted time to compensate for the non-stationarity of the system. From Section 5.2, it is known that the embedding dimension (m) and the embedding delay (τ) are 3 and 2, respectively. According to (1), the element of reconstructed phase space is four-dimensional. The input set is 4, and thus, the lagged wind speed series is the element in the reconstructed phase space:

$$\mathbf{X}_i = (x_i, x_{i+1}, \dots, x_{i+4}), \quad i = 1, 2, \dots, N - 4. \quad (22)$$

5.4. Selection of SVR Parameters. As mentioned previously, the performance of SVR modeling depends on its parameters. The optimization algorithms including COA, PSO, and GA are developed to tune crucial parameters of SVM via minimizing training errors and validation errors. In this paper, we choose RBF as the kernel function. The crucial parameters that need to be optimized are the penalties factor (C), which determines the tradeoff cost between minimizing the training error and minimizing the SVM model complexity, and gamma of the kernel function RBF (g), which defines the nonlinear mapping from the input space to the high-dimensional feature space. The corresponding parameters of the optimized algorithms are shown in Table 1.

The process to search the SVR parameters is described as the following.

Step 1. Set the range of the penalties factor (C) and gamma of the kernel function RBF (g). In this paper, C ranges from 0.1 to 100, and g ranges from 0.1 to 10.

Step 2. Initialize the parameters of each optimization algorithm according to Table 1. Note that the dimension of chromosomes of GA, particles of PSO, and population of COA are all equal to 2.

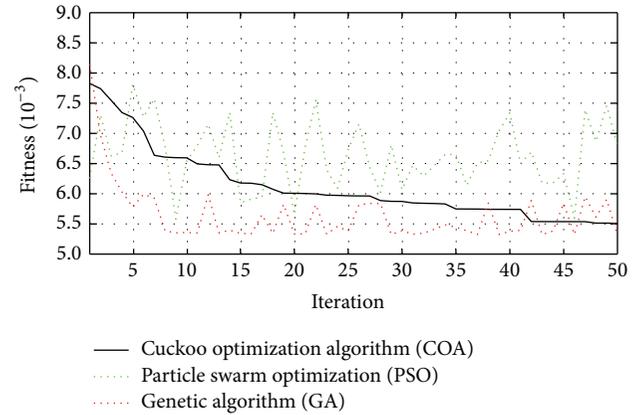


FIGURE 5: Fitness curves of one-step-ahead predictions by the optimization algorithms.

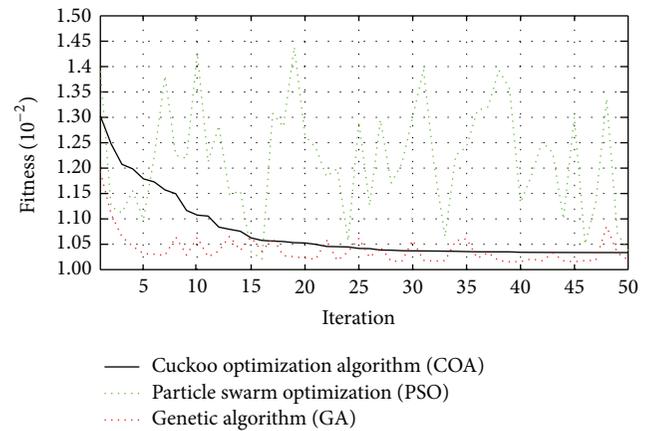


FIGURE 6: Fitness curves of two-step-ahead predictions by the optimization algorithms.

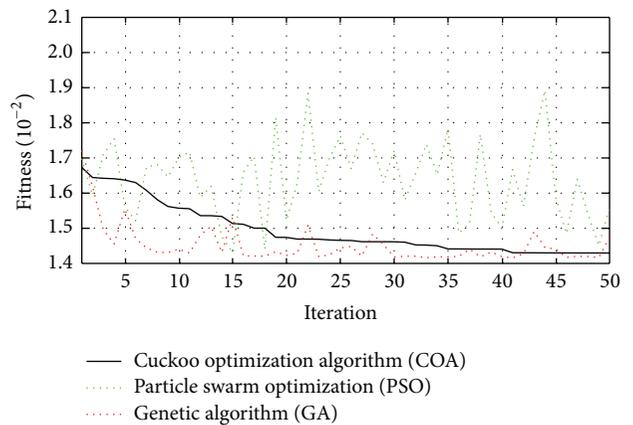


FIGURE 7: Fitness curves of three-step-ahead predictions by the optimization algorithms.

Step 3. Perform the optimization processes according to Section 3.3.

Figures 5, 6, 7, and 8 show the fitness curves for one-step- to four-step-ahead predictions of all three algorithms. It can be seen that COA has a stationary process of finding

TABLE 1: Parameters of the employed optimization algorithms.

COA		PSO		GA	
Maximum iteration	50	Maximum iteration	50	Maximum iteration	50
Number of nests (N_p)	20	Swarm size	20	Population size	20
Discovery probability (P_a)	0.25	Personal learning coefficient (c_1)	1.4945	Crossover rate	0.9
		Global learning coefficient (c_2)	1.4945	Mutation rate	0.1
		Inertia weight (ω)	0.6		

TABLE 2: Best C and g values of multi-step-ahead predictions tuned by optimized algorithm.

Forecasting horizons	COA		PSO		GA	
	C	g	C	g	C	g
One-step-ahead	9.986	0.135	1.351	1.762	1.911	0.434
Two-step-ahead	7.629	0.087	0.100	7.824	0.232	8.194
Three-step-ahead	9.873	0.140	0.100	10.000	0.243	2.490
Four-step-ahead	8.660	0.084	0.100	1.518	0.569	1.081

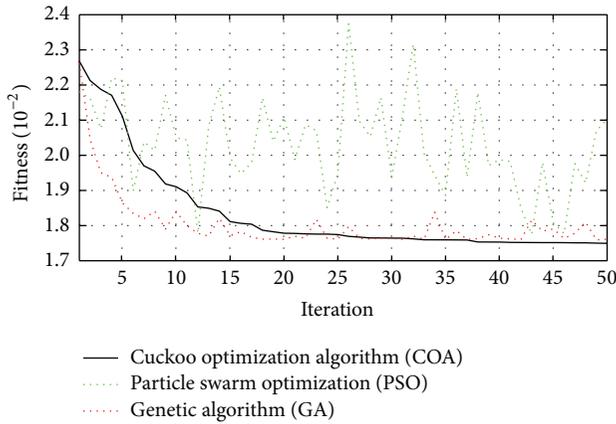


FIGURE 8: Fitness curves of four-step-ahead predictions by the optimization algorithms.

the minimal fitness, while GA and PSO show fairly high fluctuations of the fitness curve. COA is more likely to reach the minimal fitness and to maintain that status, while GA detects the minimal fitness and quickly loses it. PSO fails to find the best fitness, and overfitting may be one reason for the failure [42]. The restricted parameters of these optimization algorithms form another possible reason. To quickly obtain the forecasting results in the very short-term wind speed prediction, the parameters of these optimization algorithms are restricted. For example, if the global learning coefficient (c_2) of PSO is small or large, the running time increases. The experiment illustrates the superiority of COA over both PSO and GA.

The best parameters (in Table 2) for multi-step-ahead predictions, penalties factor (C), and gamma of kernel function RBF (g) are selected out via the three optimized algorithms.

5.5. Comparison of the Forecasting Results. The SVR models with the optimal parameters are then used to forecast the

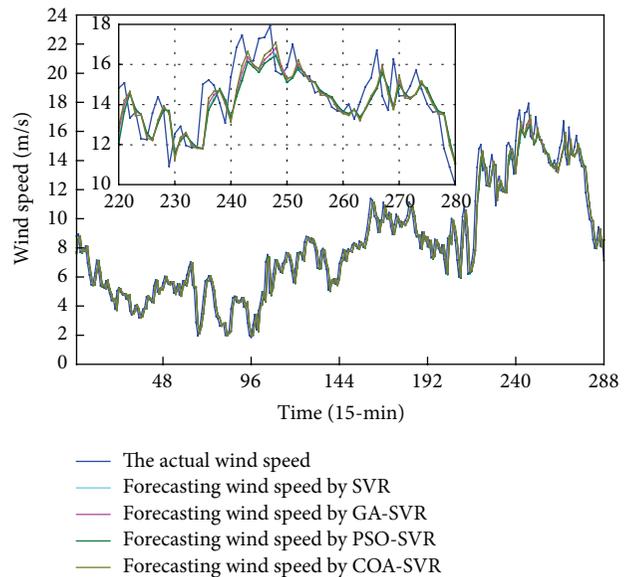


FIGURE 9: One-step-ahead forecasting results.

wind speed data. A randomly chosen SVR model with C of 2 and g of 1, denoted as the SVR method, is run for comparison with the optimized SVR models based on five criteria including MAE, MAPE, RMSE, R^2 , and IA, as shown in Table 3. The forecasting results of one-step-, two-step-, three-step-, and four-step-ahead forecasts are shown in Figures 9, 10, 11, and 12.

The following can be seen from Figures 9–12 and Table 3.

- (1) The hybrid methods have satisfactory performances in the wind speed predictions because the forecasting curves are very close to the actual wind speed curve. The subplot in the north-west corner of each figure shows that the forecasting curve of COA-SVR is closer to the actual wind speed data, especially for the two- to four-step-ahead predictions.

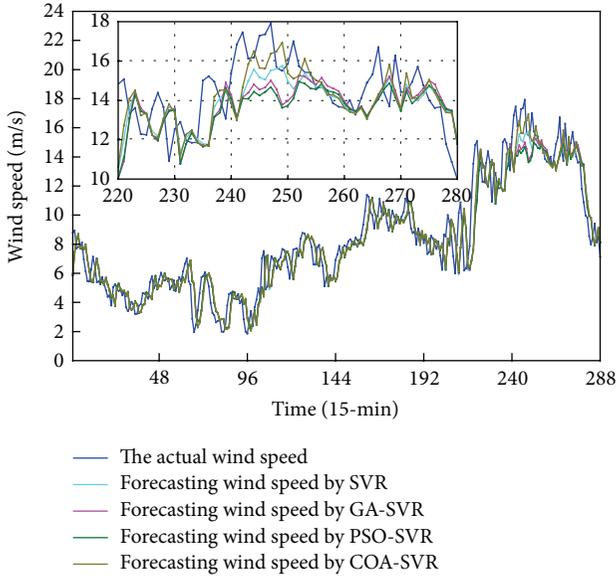


FIGURE 10: Two-step-ahead forecasting results.

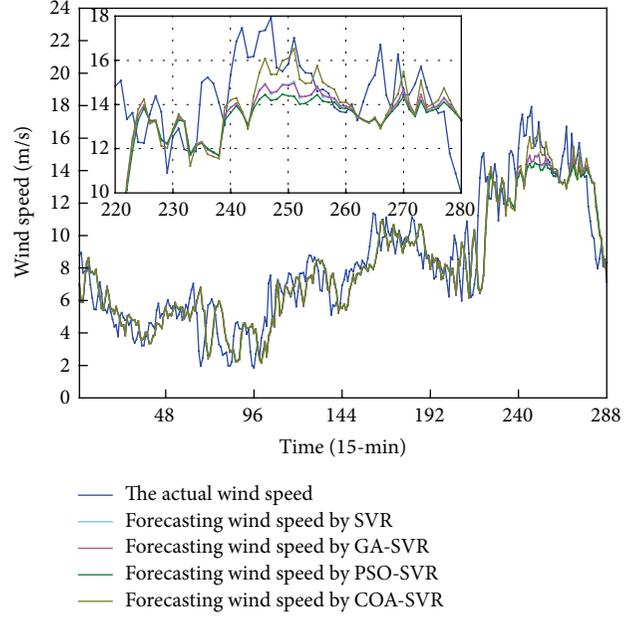


FIGURE 12: Four-step-ahead forecasting results.

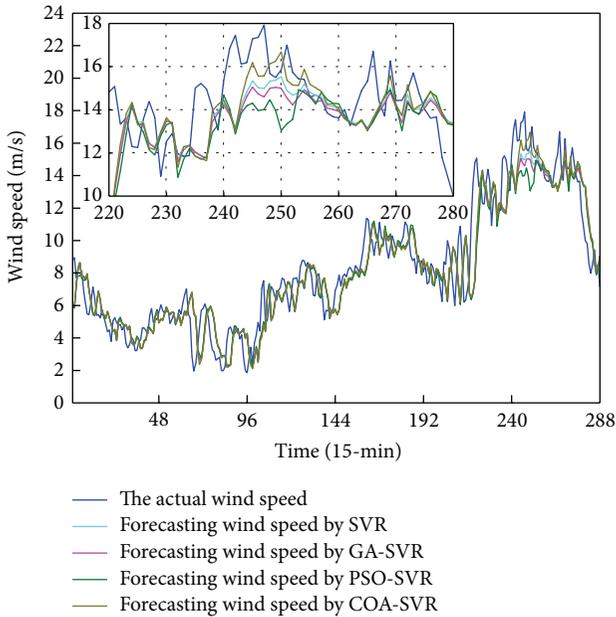


FIGURE 11: Three-step-ahead forecasting results.

- (2) The forecasting approaches are effective at forecasting the nonstationary wind speed series, and all forecasting models display satisfactory performance. Taking the four-step-ahead prediction as an example, the MAE errors of the general SVR, GA-SVR, PSO-SVR, and COA-SVR are 1.2602, 1.2616, 1.2761, and 1.2386, respectively.
- (3) The performance of the optimized SVR models is much better than that of the SVR model in which C and g take values of 2 and 1, respectively. The COA-SVR model is the best among the optimized SVR models.

- (4) When comparing the performance of the COA-SVR model with that of the SVR model, the optimized algorithm part has improved the performance of the general SVR model. The improved percentage of the RMSE from one-step to four-step is 1.98%, 1.33%, 0.80%, and 0.98%, respectively. The improved percentage of the MAPE for one-step, three-step, and four-step is 0.29%, 0.37%, and 0.63%, respectively. The improved percentage of R^2 from one-step to four-step is 0.2%, 0.24%, 0.30%, and 0.4%, respectively.
- (5) When comparing the performance of the COA-SVR model with that of the GA-SVR model, the COA part has improved the performance more than the GA part. The improved percentage of the RMSE from one-step to four-step is 0.71%, 4.53%, 1.99%, and 1.25%, respectively. The improved percentage of the MAPE from one-step to four-step is 0.16%, 0.42%, 1.20%, and 0.79%, respectively. The improved percentage of R^2 from one-step to four-step is 0.08%, 1.07%, 0.7%, and 0.53%, respectively.
- (6) When comparing the performance of the COA-SVR model with that of the PSO-SVR model, the COA part has improved the performance more than the PSO part. The improved percentage of the RMSE from one-step to four-step is 2.82%, 5.42%, 5.36%, and 2.48%, respectively. The improved percentage of the MAPE from one-step to four-step is 0.54%, 0.71%, 2.60%, and 1.12%, respectively. The improved percentage of R^2 from one-step to four-step is 0.31%, 1.17%, 1.94%, and 0.94%, respectively.

5.6. Hypothesis Test. The aim of using the hypothesis test is to answer whether the COA-SVR model is statistically robust. If the COA-SVR method is statistically robust, we

TABLE 3: Performance evaluation for one-step- to four-step-ahead predictions.

Predictions	Methods	MAE (m/s)	RMSE (m/s)	MAPE (%)	R ² (—)	IA (—)
One-step-ahead prediction	SVR	0.6911	0.9300	9.6650	0.9424	0.9846
	GA-SVR	0.6860	0.9184	9.6524	0.9435	0.9851
	PSO-SVR	0.6951	0.9376	9.6896	0.9414	0.9844
	COA-SVR	0.6836	0.9119	9.6371	0.9443	0.9853
Two-step-ahead prediction	SVR	1.0101	1.3598	14.6636	0.8779	0.9657
	GA-SVR	1.0418	1.4028	14.7376	0.8706	0.9630
	PSO-SVR	1.0472	1.4148	14.7800	0.8697	0.9618
	COA-SVR	1.0051	1.3420	14.6764	0.8800	0.9673
Three-step-ahead prediction	SVR	1.1608	1.5941	17.3654	0.8318	0.9521
	GA-SVR	1.1814	1.6129	17.5094	0.8285	0.9502
	PSO-SVR	1.2290	1.6662	17.7517	0.8181	0.9460
	COA-SVR	1.1515	1.5815	17.3018	0.8343	0.9539
Four-step-ahead prediction	SVR	1.2602	1.7247	18.9360	0.8043	0.9423
	GA-SVR	1.2616	1.7293	18.9662	0.8032	0.9421
	PSO-SVR	1.2761	1.7502	19.0296	0.7999	0.9398
	COA-SVR	1.2386	1.7079	18.8183	0.8075	0.9451

TABLE 4: Two-tailed hypothesis test results.

Compare groups	Levene's test for equality of variances	<i>t</i> -test for quality of means				α
		<i>F</i>	Sig.	<i>t</i>	Sig. (two-tailed)	
μ_0 and μ_1	Equal variances assumed	0.188	0.665	0.252	0.801	0.05
	Equal variances not assumed			0.252	0.801	0.05
μ_0 and μ_2	Equal variances assumed	0.483	0.487	0.462	0.645	0.05
	Equal variances not assumed			0.462	0.645	0.05
μ_0 and μ_3	Equal variances assumed	0.638	0.425	0.585	0.559	0.05
	Equal variances not assumed			0.585	0.559	0.05
μ_0 and μ_4	Equal variances assumed	1.092	0.297	0.728	0.467	0.05
	Equal variances not assumed			0.728	0.467	0.05

can confidently guarantee that the hybrid method has good generalization and can be used in other areas. A two-tailed hypothesis test is taken to the forecasting and actual data [9]. Based on the forecasting results above, COA-SVR has the best performance among the optimized methods for multi-step-ahead wind speed prediction. Thus, the COA-SVR predictions and the actual wind speed are selected for hypothesis test.

The hypothesis test is expressed as the following:

$$\begin{aligned} H_0: \mu_0 &= \mu_1 = \mu_2 = \mu_3 = \mu_4, \\ H_1: \mu_i &\neq \mu_j, \quad i, j = 0, 1, 2, 3 \quad i \neq j. \end{aligned} \quad (23)$$

Here, μ_0 is the mean of the actual wind speed. The variables $\mu_1, \mu_2, \mu_3,$ and μ_4 are the average of the COS-SVR forecasting results from the one-step- to four-step-ahead predictions. The two-tailed hypothesis test is taken with $\alpha = 0.05$, and it is run based on SPSS software (in Table 4). It can be seen from Table 4, with 95% confidence, that the average values of the COA-SVR predictions from one-step to four-step are equal to the average of the actual wind speed series.

The hypothesis test indicates that the COA-SVR model has advantages of both great precision and good generalization.

6. Discussion and Conclusions

Accurate short-term wind speed prediction is crucial for improving the stability of grid-connected wind power generality units and avoiding disadvantageous impacts on the electric networks. This paper proposed a hybrid model based on recurrence plot and optimized support vector regression model for wind speed prediction.

Wind is caused by the complex interaction of meteorological conditions, which makes wind a complex dynamic process and extremely difficult to forecast. Analyzing the predictability of wind speed is a primary step to build a proper model.

Recurrence plot (RP), which performs excellently in studying dynamic systems, is thus chosen to unravel the available wind speed. The results of RP reveal that wind speed data here has short-term predictability over the short-term time scale.

With the capability of modeling complex nonlinearity systems, support vector regression (SVR) models are employed to forecast the wind speed data at the Shandong wind farm in China. The forecast horizons are 15 min (one-step-ahead), 30 min (two-step-ahead), 45 min (three-step-ahead), and 1 h ahead (four-step-ahead). The forecasting accuracy of SVR

greatly depends on its parameters (C and g) and its input signals. Therefore, proper selection of inputs and parameters is essential for satisfactory SVR performance. The elements of the reconstructed phase space of the RP with the biggest predictability are naturally chosen as the input sets. Once the input sets are fixed, the optimum parameters of SVR are selected by some optimized algorithms, including genetic algorithm (GA), particle swarm optimization (PSO), and cuckoo optimization algorithm (COA).

Finally, the optimized approaches are then evaluated according to some chosen performance criteria. The forecasting results show that the available wind data has short-term predictability over the short-term time scale; the performance of optimized SVR models is better than that of the SVR model without parameter selection. COA-SVR is superior among the optimized SVR models, especially in dealing with the jumping samples. Furthermore, a two-tailed hypothesis is performed for checking the robustness of the COA-SVR model. With 95% confidence, the COA-SVR method is statistically robust in multi-step-ahead predictions.

Highlights

- (i) Recurrence plot (RP) is a new and powerful tool to explore wind system.
- (ii) Short-term predictability of wind speed on a short-term time scale is proved by RP.
- (iii) The optimized SVR techniques are highly satisfactory.
- (iv) The COA-SVR model is statistically robust, especially in addressing jumping samples.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

The work was supported by the National Natural Science Foundation of China (Grant no. 71171102).

References

- [1] N. Bigdeli, K. Afshar, A. S. Gazafroudi, and M. Y. Ramandi, "A comparative study of optimal hybrid methods for wind power prediction in wind farm of Alberta, Canada," *Renewable & Sustainable Energy Reviews*, vol. 27, pp. 20–29, 2013.
- [2] G. Boyle, *Renewable Energy*, Oxford University Press, 2004.
- [3] Global Wind Energy Council (GWEC), "The global status of wind power in 2012," <http://www.gwec.net/wp-content/uploads/2013/07/The-Global-Status-of-Wind-Power-in-2012.pdf>.
- [4] W. Zhang, Z. Su, H. Zhang, Y. Zhao, and Z. Zhao, "Hybrid wind speed forecasting model study based on SSA and intelligent optimized algorithm," *Abstract and Applied Analysis*, vol. 2014, Article ID 693205, 14 pages, 2014.
- [5] J. Wu, J. Wang, and D. Chi, "Wind energy potential assessment for the site of Inner Mongolia in China," *Renewable and Sustainable Energy Reviews*, vol. 21, pp. 215–228, 2013.
- [6] M. Monfared, H. Rastegar, and H. M. Kojabadi, "A new strategy for wind speed forecasting using artificial intelligent methods," *Renewable Energy*, vol. 34, pp. 845–848, 2009.
- [7] R. G. Kavasseri and K. Seetharaman, "Day-ahead wind speed forecasting using f-ARIMA models," *Renewable Energy*, vol. 34, no. 5, pp. 1388–1393, 2009.
- [8] Y. Dong, Z. Guo, J. Wang, and H. Lu, "The forecasting procedure for long-term wind speed in the Zhangye area," *Mathematical Problems in Engineering*, vol. 2010, Article ID 684742, 17 pages, 2010.
- [9] Z. H. Guo, J. Zhao, W. Y. Zhang, and J. Z. Wang, "A corrected hybrid approach for wind speed prediction in Hexi Corridor of China," *Energy*, vol. 36, no. 3, pp. 1668–1679, 2011.
- [10] Z.-H. Guo, J. Wu, H.-Y. Lu, and J.-Z. Wang, "A case study on a hybrid wind speed forecasting method using BP neural network," *Knowledge-Based Systems*, vol. 24, no. 7, pp. 1048–1056, 2011.
- [11] C. Ren, N. An, J. Wang, L. Li, B. Hu, and D. Shang, "Optimal parameters selection for BP neural network based on particle swarm optimization: a case study of wind speed forecasting," *Knowledge-Based Systems*, vol. 56, pp. 226–239, 2014.
- [12] H. Q. Minh, N. Frédéric, E. Najib, and H. Abdelaziz, "Power management of a variable speed wind turbine for stand-alone system using fuzzy logic," in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ '11)*, pp. 1404–1410, June 2011.
- [13] M. G. Simões, B. K. Bose, and R. J. Spiegel, "Fuzzy logic based intelligent control of a variable speed cage machine wind generation system," *IEEE Transactions on Power Electronics*, vol. 12, no. 1, pp. 87–95, 1997.
- [14] X. Cheng and P. Guo, "Short-term wind speed prediction based on support vector machine of fuzzy information granulation," in *Proceedings of the 25th Chinese Control and Decision Conference (CCDC '13)*, pp. 1918–1923, May 2013.
- [15] D. Liu, D. X. Niu, H. Wang, and L. L. Fan, "Short-term wind speed forecasting using wavelet transform and support vector machines optimized by genetic algorithm," *Renewable Energy*, vol. 62, pp. 592–597, 2014.
- [16] M. A. Mohandes, T. O. Halawani, S. Rehman, and A. A. Hussain, "Support vector machines for wind speed prediction," *Renewable Energy*, vol. 29, no. 6, pp. 939–947, 2004.
- [17] S. Salcedo-Sanz, E. G. Ortiz-García, Á. M. Pérez-Bellido, A. Portilla-Figueras, and L. Prieto, "Short term wind speed prediction based on evolutionary support vector regression algorithms," *Expert Systems with Applications*, vol. 38, no. 4, pp. 4052–4057, 2011.
- [18] S. Salcedo Sanz, E. G. Ortiz-García, Á. M. Pérez-Bellido et al., "Performance comparison of multilayer perceptrons and support vector machines in a short-term wind speed prediction problem," *Neural Network World*, vol. 19, no. 1, pp. 37–51, 2009.
- [19] Y. Wang, D. L. Wu, C. X. Guo, Q. H. Wu, W. Z. Qian, and J. Yang, "Short-term wind speed prediction using support vector regression," *Proceedings of the IEEE Power and Energy Society General Meeting*, 2010.
- [20] P. Zhao, J. Xia, Y. Dai, and J. He, "Wind speed prediction using support vector regression," in *Proceedings of the 5th IEEE Conference on Industrial Electronics and Applications (ICIEA '10)*, pp. 882–886, Taichung, Taiwan, June 2010.
- [21] S. L. Zhou, M. Q. Mao, and L. Chang, "Forecasting of wind speed based on wavelet analysis and support vector machine," in *Proceedings of the 2nd International Symposium on Power*

- Electronics for Distributed Generation Systems (PEDG '10)*, pp. 864–867, June 2010.
- [22] Y. Quan, S. Wang, M. Gu, and J. Kuang, “Field measurement of wind speeds and wind-induced responses atop the shanghai world financial center under normal climate conditions,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 902643, 14 pages, 2013.
- [23] E. Cadenas and W. Rivera, “Short term wind speed forecasting in La Venta, Oaxaca, México, using artificial neural networks,” *Renewable Energy*, vol. 34, no. 1, pp. 274–278, 2009.
- [24] N. Amjady, F. Keynia, and H. Zareipour, “A new hybrid iterative method for short-term wind speed forecasting,” *European Transactions on Electrical Power*, vol. 21, no. 1, pp. 581–595, 2011.
- [25] J.-P. Eckmann, S. O. Kamphorst, D. Ruelle, and S. Ciliberto, “Liapunov exponents from time series,” *Physical Review A*, vol. 34, no. 6, pp. 4971–4979, 1986.
- [26] N. Bigdeli, M. Jafarzadeh, and K. Afshar, “Characterization of Iran stock market indices using recurrence plots,” in *Proceedings of the International Conference on Management and Service Science (MASS '11)*, pp. 1–5, Wuhan, China, August 2011.
- [27] K. C. Andrade, R. Wehrle, V. I. Spoomaker, P. G. Sämann, and M. Czisch, “Statistical evaluation of recurrence quantification analysis applied on single trial evoked potential studies,” *Clinical Neurophysiology*, vol. 123, no. 8, pp. 1523–1535, 2012.
- [28] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [29] V. Vapnik, S. E. Golowich, and A. Smola, “Support vector method for function approximation, regression estimation, and signal processing,” in *Proceedings of the 10th Annual Conference on Neural Information Processing Systems (NIPS '96)*, pp. 281–287, December 1996.
- [30] R. Stoean, M. Preuss, D. Dumitrescu, and C. Stoean, “Evolutionary support vector regression machines,” in *Proceedings of the 8th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC '06)*, pp. 330–335, Timișoara, Romania, September 2006.
- [31] J. M. Hu, J. Z. Wang, and G. W. Zeng, “A hybrid forecasting approach applied to wind speed time series,” *Renewable Energy*, vol. 60, pp. 185–194, 2013.
- [32] H. J. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, Mass, USA, 1992.
- [33] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, vol. 412, Addison-Wesley, Reading Menlo Park, Calif, USA, 1989.
- [34] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [35] X.-S. Yang and S. Deb, “Cuckoo search via Lévy flights,” in *Proceedings of the World Congress on Nature and Biologically Inspired Computing (NABIC '09)*, pp. 210–214, Coimbatore, India, December 2009.
- [36] P. Jiang, Q. Zhou, H. Jiang, and Y. Dong, “An optimized forecasting approach based on grey theory and Cuckoo search algorithm: a case study for electricity consumption in New South Wales,” *Abstract and Applied Analysis*, vol. 2014, Article ID 183095, 13 pages, 2014.
- [37] S. Painter, “Stochastic interpolation of aquifer properties using fractional Levy motion,” *Water Resources Research*, vol. 32, no. 5, pp. 1323–1332, 1996.
- [38] <http://windpower.ofweek.com/2014-03/ART-330002-8420-28792372.html>.
- [39] N. Bigdeli and M. Haeri, “Time-series analysis of TCP/RED computer networks, an empirical study,” *Chaos, Solitons & Fractals*, vol. 39, no. 2, pp. 784–800, 2009.
- [40] J. Timmer, S. Häussler, M. Lauk, and C.-H. Lücking, “Pathological tremors: deterministic chaos or nonlinear stochastic oscillators?” *Chaos*, vol. 10, no. 1, pp. 278–288, 2000.
- [41] “CRP toolbox for Matlab provided by TOCSY,” 2014, <http://tocsy.agnld.uni-potsdam.de>.
- [42] N. I. Ghali, N. El-Dessouki, A. Mervat, and L. Bakrawi, “Exponential particle swarm optimization approach for improving data clustering,” *International Journal of Electrical, Computer, and Systems Engineering*, vol. 3, pp. 208–212, 2009.

Research Article

Swarm Intelligence-Based Smart Energy Allocation Strategy for Charging Stations of Plug-In Hybrid Electric Vehicles

Imran Rahman,¹ Pandian M. Vasant,¹
Balbir Singh Mahinder Singh,¹ and M. Abdullah-Al-Wadud²

¹Department of Fundamental and Applied Sciences, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia

²Department of Software Engineering, College of Computer and Information Sciences, King Saud University, P.O. BOX 2454, Riyadh 11451, Saudi Arabia

Correspondence should be addressed to Pandian M. Vasant; pvasant@gmail.com

Received 15 May 2014; Accepted 18 August 2014

Academic Editor: Baozhen Yao

Copyright © 2015 Imran Rahman et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recent researches towards the use of green technologies to reduce pollution and higher penetration of renewable energy sources in the transportation sector have been gaining popularity. In this wake, extensive participation of plug-in hybrid electric vehicles (PHEVs) requires adequate charging allocation strategy using a combination of smart grid systems and smart charging infrastructures. Daytime charging stations will be needed for daily usage of PHEVs due to the limited all-electric range. Intelligent energy management is an important issue which has already drawn much attention of researchers. Most of these works require formulation of mathematical models with extensive use of computational intelligence-based optimization techniques to solve many technical problems. In this paper, gravitational search algorithm (GSA) has been applied and compared with another member of swarm family, particle swarm optimization (PSO), considering constraints such as energy price, remaining battery capacity, and remaining charging time. Simulation results obtained for maximizing the highly nonlinear objective function evaluate the performance of both techniques in terms of best fitness.

1. Introduction

The vehicular network recently accounts for around 25% of CO₂ emissions and over 55% of oil consumption around the world [1]. Carbon dioxide (CO₂) is the primary greenhouse gas emitted through human activities like combustion of fossil fuels (coal, natural gas, and oil) for energy and transportation. Several researchers have proved that a great amount of reductions in greenhouse gas emissions and the increasing dependence on oil could be accomplished by electrification of transport sector [2]. Indeed, the adoption of hybrid electric vehicles (HEVs) has brought significant market success over the past decade. Vehicles can be classified into three groups: internal combustion engine vehicles (ICEVs), hybrid electric vehicles (HEVs), and all-electric vehicles (AEVs) [3]. Plug-in hybrid electric vehicles (PHEVs) which are very recently introduced promise to boost up the overall fuel efficiency

by holding a higher capacity battery system, which can be directly charged from traditional power grid system that helps the vehicles to operate continuously in “all-electric range” (AER). All-electric vehicle or AEV is a vehicle using electric power as the only source to move the vehicle [4]. Plug-in hybrid electric vehicles with a connection to the smart grid can possess all of these strategies. Hence, the widely extended adoption of PHEVs might play a significant role in the alternative energy integration into traditional grid systems [5]. There is a need of efficient mechanisms and algorithms for smart grid technologies in order to solve highly heterogeneous problems like energy management, cost reduction, efficient charging infrastructure, and so forth with different objectives and system constraints [6].

According to a statistics of Electric Power Research Institute (EPRI), about 62% of the entire United States (US) vehicle will comprise PHEVs within the year 2050 [7].

Moreover, there is an increasing demand to implement this technology on the electric grid system. Large numbers of PHEVs have the capability to threaten the stability of the power system. For example, in order to avoid interruption when several thousand PHEVs are introduced into the system over a short period of time, the load on the power grid will need to be managed very carefully. One of the main targets is to facilitate the proper interaction between the power grid and the PHEV. For the maximization of customer satisfaction and minimization of burdens on the grid, a complicated control mechanism will need to be addressed in order to govern multiple battery loads from a number of PHEVs appropriately [8]. The total demand pattern will also have an important impact on the electricity industry due to differences in the needs of the PHEVs parked in the deck at certain time [9]. Proper management can ensure strain minimization of the grid and enhance the transmission and generation of electric power supply. The control of PHEV charging depending on the locations can be classified into two groups: household charging and public charging. The proposed optimization focuses on the public charging station for plug-in vehicles because most of PHEV charging is expected to take place in public charging locations [10].

Wide penetration of PHEVs in the market depends on a well efficient charging infrastructure. The power demand from this new load will put extra stress on the traditional power grid [11]. As a result, a good number of PHEV charging infrastructures with appropriate facilities are essential to be built for recharging electric vehicles; for this some strategies have been proposed by the researchers [12, 13]. Charging stations are needed to be built at workplaces, markets/shopping malls, and home. In [14], authors proposed the necessity of building new smart charging station with effective communication among utilities along with substation control infrastructure in view of grid stability and proper energy utilization. Furthermore, assortment of charging stations with respect to charging characteristics of different PHEVs traffic mobility characteristics, sizeable energy storage, cost minimization, quality of services (QoS), and optimal power of intelligent charging station are underway [15]. Thus, evolution of reliable, efficient, robust, and economical charging infrastructure is underway. In this wake, numerous techniques and methods have been proposed for deployment of charging station for PHEVs [16, 17].

One of the important constraints for accurate charging is state of charge (SoC). Charging algorithm can accurately be managed by the precise state of charge estimation [18]. An approximate graph of a typical lithium-ion cell voltage versus SoC is shown in Figure 1. The figure indicates that the slope of the curve below 20% and above 90% is high enough to result in a detectable voltage difference to be relied on by charge balancing control and measurement circuits [19]. There is a need of in-depth study on maximization of average SoC in order to facilitate intelligent energy allocation for PHEVs in a charging station. Gravitational search algorithm (GSA) is one of the newest heuristic algorithms introduced by Rashedi et al. [20]. GSA algorithm is also a member of swarm intelligence family which is inspired by the well-known law of

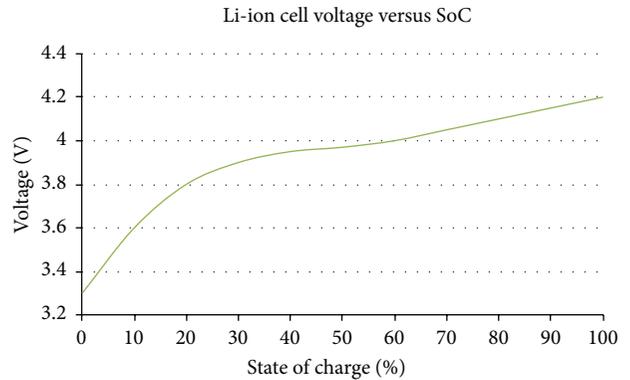


FIGURE 1: Li-ion cell voltage versus state of charge [24].

gravity and interactions between the masses and implements Newtonian gravity and the laws of motion [21–23].

GSA-based optimization has already been used by the researchers for postoutage bus voltage magnitude calculations, economic dispatch with valve-point effects, optimal sizing and suitable placement for distributed generation (DG) in distribution system, optimization of synthesis gas production [25], rectangular patch antenna [26], orthogonal array based performance improvement [27], solving thermal unit commitment (UC) problem, and finding out optimal solution for optimal power flow (OPF) problem in a power system [28]. Specifically, we are investigating the use of the gravitational search algorithm (GSA) method for developing real-time and large-scale optimizations for allocating power.

The remainder of this paper is organized as follows: next section will describe the specific problem that we are trying to solve. We will provide the optimization objective and constraints and mathematical formulation of our algorithm, review the GSA method, and describe how the algorithm works for our optimization problems. The simulation results and analysis are then presented. Finally, conclusions and future directions are drawn.

2. Problem Formulation

The idea behind smart charging is to charge the vehicle when it is most beneficial such as when the electricity price and total power demand remain lowest or there is excess capacity of generated power [24].

Suppose there is a charging station with the capacity of total power P . Total N numbers of PHEVs need to be charged within 24 hours of time interval. The proposed system should allow PHEVs to leave the charging station before their expected leaving time for making the system more effective. It is worth to mention that each PHEV is regarded to be plugged in to the charging station once. The main aim is to allocate power intelligently for each PHEV coming to the charging station. The state of charge is the main parameter which needs to be maximized in order to allocate power effectively. For this, the objective function considered in this paper is the maximization of average SoC and thus allocates energy for PHEVs at the next time step.

The constraints considered are charging time, present SoC, and price of the energy.

The objective function is defined as

$$\begin{aligned} \max J(k) &= \sum_i w_i(k) \text{SoC}_i(k+1), \\ w_i(k) &= f(C_{r,i}(k), T_{r,i}(k), D_i(k)), \\ C_{r,i}(k) &= (1 - \text{SoC}_i(k)) * C_i, \end{aligned} \quad (1)$$

where $C_{r,i}(k)$ is the battery capacity (remaining) needed to be filled for i number of PHEV at time step k ; C_i is the battery capacity (rated) of the i number of PHEV; remaining time for charging a particular PHEV at time step k is expressed as $T_{r,i}(k)$; the price difference between the real-time energy price and the price that a specific customer at the i number of PHEV charger is willing to pay at time step k is presented by $D_i(k)$; $w_i(k)$ is the charging weighting term of the i number of PHEV at time step k (a function of charging time, present SoC, and price of the energy); $\text{SoC}_i(k+1)$ is the state of charge of the i number of PHEV at time step $k+1$.

Here, the weighting term indicates a bonus proportional to the attributes of a specific PHEV. For example, if a PHEV has a lower initial SoC and less charging time (remaining), but the driver is eager to pay a higher price, the system will provide more power to this particular PHEV battery charger:

$$w_i(k) \propto \left[\text{Cap}_{r,i}(k) + D_i(k) + \frac{1}{T_{r,i}(k)} \right]. \quad (2)$$

The charging current is also assumed to be constant over Δt :

$$\begin{aligned} [\text{SoC}_i(k+1) - \text{SoC}_i(k)] \cdot \text{Cap}_i &= Q_i = I_i(k) \Delta t, \\ \text{SoC}_i(k+1) &= \text{SoC}_i(k) + \frac{I_i(k) \Delta t}{\text{Cap}_i}, \end{aligned} \quad (3)$$

where the sample time Δt is defined by the charging station operators and $I_i(k)$ is the charging current over Δt .

The battery model is regarded as a capacitor circuit, where C_i is the capacitance of battery (Farad). The model is defined as

$$C_i \cdot \frac{dV_i}{dt} = I_i. \quad (4)$$

Therefore, over a small time interval, one can assume the change of voltage to be linear:

$$\begin{aligned} C_i \cdot \frac{[V_i(k+1) - V_i(k)]}{\Delta t} &= I_i, \\ V_i(k+1) - V_i(k) &= \frac{I_i \Delta t}{C_i}. \end{aligned} \quad (5)$$

As the decision variable used here is the allocated power to the PHEVs, by replacing $I_i(k)$ with $P_i(k)$ the objective function finally becomes

$$\begin{aligned} J(k) &= \sum w_i \cdot \left[\text{SoC}_i(k) + (2P_i(k) \Delta t) \right. \\ &\quad \times \left(0.5 \cdot C_i \cdot \left[\sqrt{\frac{2P_i(k) \Delta t}{C_i} + V_i^2(k)} \right. \right. \\ &\quad \left. \left. + V_i(k) \right] \right)^{-1} \left. \right]. \end{aligned} \quad (6)$$

2.1. System Constraints. Possible real-world constraints could include the charging rate (i.e., slow, medium, and fast), the time that the PHEV is connected to the grid, the desired departure SOC, the maximum electricity price that a user is willing to pay, and certain battery requirements. Furthermore, the available communication bandwidth could limit sampling time, which would have effects on the processing ability of the vehicle.

Power obtained from the utility (P_{utility}) and the maximum power ($P_{i,\text{max}}$) absorbed by a specific PHEV are the primary energy constraints being considered in this paper. The power demand of a PHEV/PEV cannot exceed the rated power output of the battery charger:

$$\begin{aligned} \sum_i P_i(k) &\leq P_{\text{utility}}(k) \times \eta, \\ 0 &\leq P_i(k) \leq P_{i,\text{max}}(k). \end{aligned} \quad (7)$$

The overall charging efficiency of a particular charging infrastructure is described by η . From the system point of view, charging efficiency is supposed to be constant at any given time step. Maximum battery SoC limit for the i number of PHEV is $\text{SoC}_{i,\text{max}}$. When SoC_i reaches the values close to $\text{SoC}_{i,\text{max}}$, the i number of battery charger shifts to a standby mode. The state of charge ramp rate is confined within limits by the constraint $\Delta \text{SoC}_{\text{max}}$. To accommodate the system dynamics, the energy scheduling is updated when (i) system utility data is updated; (ii) a new PHEV is plugged in, and (iii) time period Δt has periodically passed. Table 1 shows all the objective function parameters that were tuned for performing the optimization.

Energy allocation to PHEV charging station is subjected to various constraints as mentioned in the problem formulation section. Different constraints make the entire search space limited to a particular suitable region. So, a powerful optimization algorithm should be implemented in order to achieve high quality solutions with a stable convergence rate.

3. Gravitational Search Algorithm

GSA is an optimization method which has been introduced by Rashedi et al. in the year of 2009 [20]. In GSA, the

TABLE 1: Parameter settings of the objective function.

Parameter	Values
Fixed parameters	Maximum power, $P_{i,\max} = 6.7$ kWh
	Charging station efficiency, $\eta = 0.9$
	Total charging time, $\Delta t = 20$ minutes (1200 seconds)
	Power allocation to each PHEV: 30 W
Variables	$0.2 \leq \text{state of charge (SoC)} \leq 0.8$
	Waiting time ≤ 30 minutes (1800 seconds)
	$16 \text{ kWh} \leq \text{battery capacity } (C_i) \leq 40 \text{ kWh}$
Constraints	$\sum_i P_i(k) \leq P_{\text{utility}}(k) \times \eta$
	$0 \leq P_i(k) \leq P_{i,\max}(k)$
	$0 \leq \text{SoC}_i(k) \leq \text{SoC}_{i,\max}$
	$0 \leq \text{SoC}_i(k+1) - \text{SoC}_i(k) \leq \Delta \text{SoC}_{\max}$

specifications of each mass (or agent) are four in total, which are inertial mass, position, active gravitational mass and passive gravitational mass. The position of the mass presents a solution of a particular problem and masses (gravitational and inertial) are obtained by using a fitness function. GSA can be considered as a collection of agents (candidate solutions), whose masses are proportional to their value of fitness function. During generations, all masses attract each other by the gravity forces between them. A heavier mass has the bigger attraction force. Therefore the heavier masses which are probably close to the global optimum attract the other masses proportional to their distances.

3.1. Law of Gravity. The law states that particles attract each other and the force of gravitation between two particles is directly proportional to the product of their masses and inversely proportional to the distance between them.

3.2. Law of Motion. The law states that the present velocity of any mass is the summation of the fraction of its previous velocity and the velocity variance. Variation in the velocity or acceleration of any mass is equal to the force divided by inertia mass.

The gravitational force is expressed as follows:

$$F_{ij}^d(t) = G(t) \frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon} (x_j^d(t) - x_i^d(t)), \quad (8)$$

where M_{aj} is the active gravitational mass related to agent j , M_{pi} is the passive gravitational mass related to agent i , $G(t)$ is gravitational constant at time t , ε is a small constant, and $R_{ij}(t)$ is the Euclidian distance between two agents i and j . The $G(t)$ is calculated as follows:

$$G(t) = G_0 \times \exp\left(\frac{-\alpha \times \text{iter}}{\max \text{ iter}}\right), \quad (9)$$

where α and G_0 are descending coefficient and primary value, respectively, and current iteration and maximum number of

iterations are expressed as iter and max iter. In a problem space with the dimension d , the overall force acting on agent i is estimated as the following equation:

$$F_i^d(t) = \sum_{j=1, j \neq i}^N \text{rand}_j F_{ij}^d(t), \quad (10)$$

where rand_j is a random number with interval $[0, 1]$. From law of motion we know that an agent's acceleration is directly proportional to the resultant force and inverse of its mass, so the acceleration of all agents should be calculated as follows:

$$\text{ac}_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)}, \quad (11)$$

where t is a specific time and M_{ii} is the mass of the object i . The velocity and position of agents are calculated as follows:

$$\text{vel}_i^d(t+1) = \text{rand}_i \times \text{vel}_i^d(t) + \text{ac}_i^d(t), \quad (12)$$

$$x_i^d(t+1) = x_i^d(t) + \text{vel}_i^d(t+1), \quad (13)$$

where rand_i is a random number with interval $[0, 1]$.

Gravitational and inertia masses are simply calculated by the fitness evaluation. A heavier mass means a more efficient agent. This means that better agents have higher attractions and walk more slowly. Assuming the equality of the gravitational and inertia mass, the values of masses are calculated using the map of fitness. We update the gravitational and inertia masses by the following equations:

$$M_{ai} = M_{pi} = M_{ii} = \bar{M}_i, \quad i = 1, 2, \dots, N. \quad (14)$$

In gravitational search algorithm, all agents are initialized first with random values. Each of the agents is a candidate solution. After initialization, velocities for all agents are defined using (12). Moreover, the gravitational constant, overall forces, and accelerations are determined by (9), (10), and (11), respectively. The positions of agents are calculated using (13). At the end, GSA will be terminated by meeting the stopping criterion of maximum 100 iterations. The parameter settings for GSA are demonstrated in Table 2. Moreover, GSA flowchart is shown in Figure 2.

4. Simulation Results and Analysis

The GSA algorithm was applied to find out global best fitness of the objective function (Algorithm 1). All the calculations were run on an Intel (R) Core i5-3470 M CPU@ 3.20 GHz, 4.00 GB RAM, Microsoft 32 bit Windows 7 OS, and MATLAB® R2013a.

Many optimization algorithms involve local search techniques which can get stuck on local maxima. Most search techniques strive to find a global maximum in the presence of local maxima [29]. One of the most important characteristics of GSA is its significant performance during exploration process. The capability of an algorithm to extend the problem in search gap is known as exploration while the ability of an algorithm to recognize optimal solution near a favorable one is exploitation [30, 31].

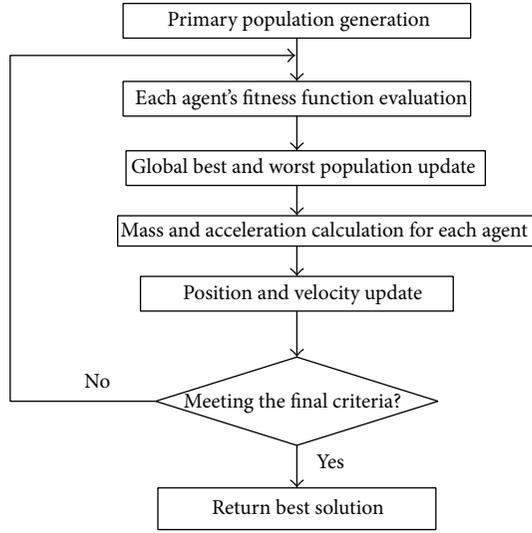


FIGURE 2: The GSA flowchart.

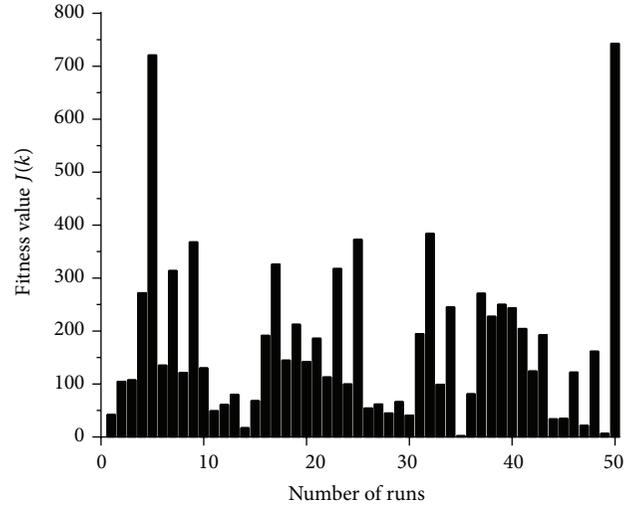


FIGURE 5: Fitness value versus number of runs (300 PHEVs).

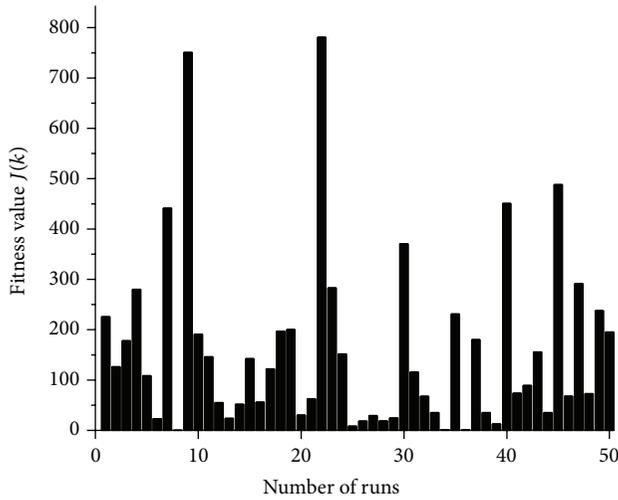


FIGURE 3: Fitness value versus number of runs (50 PHEVs).

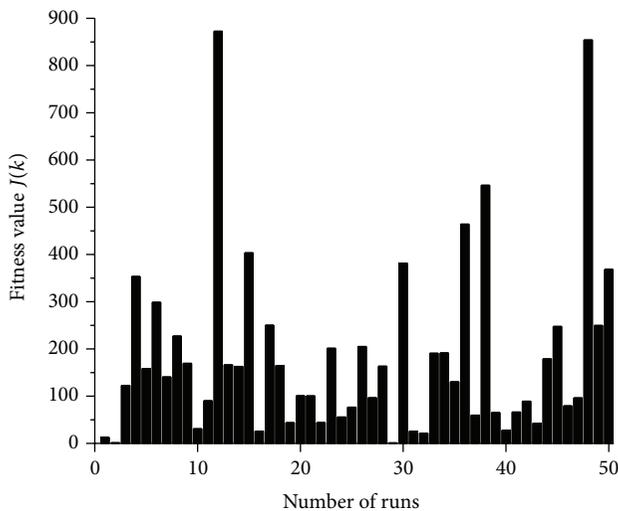


FIGURE 4: Fitness value versus number of runs (100 PHEVs).

TABLE 2: GSA parameter settings.

Parameters	Values
Primary parameter, G_0	100
Number of mass agents, n	100
Constant parameter, α	20
Constant parameter, ϵ	.01
Power of "R"	1
Maximum iteration	100
Number of runs	50

Figures 3, 4, 5, 6, and 7 show the simulation results for 50,100, 300,500, and 1000 plug-in hybrid electric vehicles (PHEVs), respectively, for finding the maximum fitness value of objective function J . In order to evaluate the performance and show the efficiency and superiority of the proposed algorithm, we ran each scenario 50 times in total.

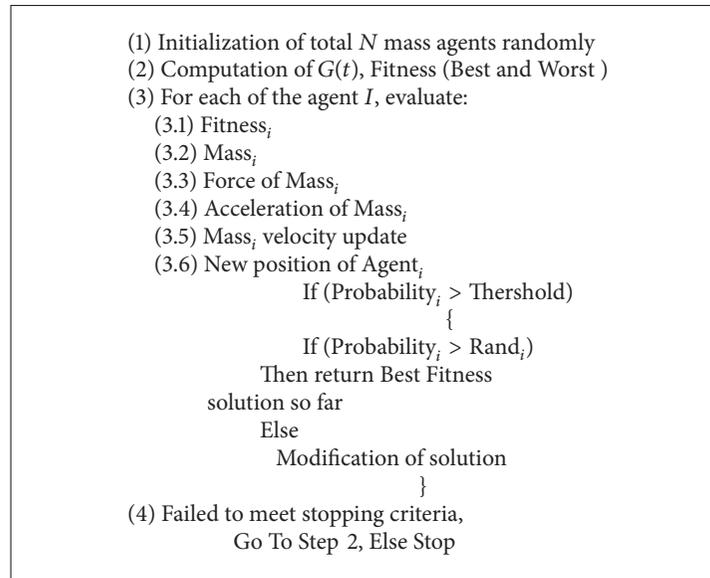
For Figure 3 (50 PHEVs), the maximum best fitness and minimum best fitness were 781.1267 and 0.2191, respectively.

The average best fitness is 158.8289. Figure 4 depicts the maximum fitness value for 100 PHEVs. In this case, the maximum best fitness and minimum best fitness were 579.3955 and 3.2523. The average best fitness is decreased into 139.7536.

For Figure 5 (300 PHEVs), the maximum best fitness and minimum best fitness were 743.1251 and 2.3279, respectively. The average best fitness is 172.4296.

Figure 6 depicts the maximum fitness value for 500 PHEVs. In this case, the maximum best fitness and minimum best fitness were 836.2707 and 0.9818. The average best fitness is decreased into 152.36437.

Figure 7 shows the maximum fitness value for 1000 PHEVs. In this case, the maximum best fitness and minimum best fitness were 968.7652 and 7.2747. The average best fitness is decreased into 161.52349.



ALGORITHM 1

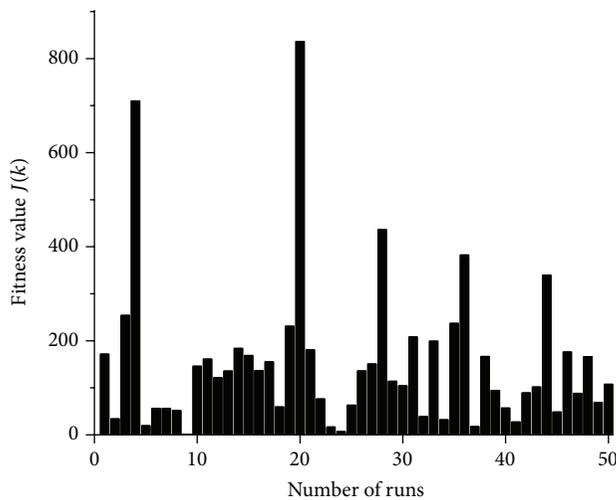


FIGURE 6: Fitness value versus number of runs (500 PHEVs).

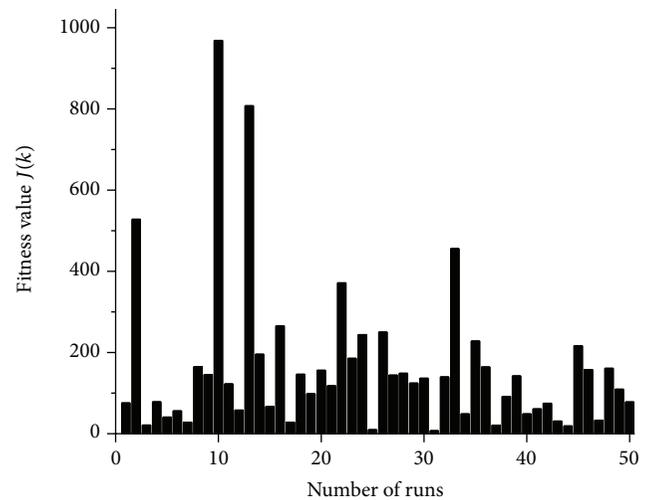


FIGURE 7: Fitness value versus number of runs (for 1000 PHEVs).

Finally, Table 3 summarizes the result. From that it can be concluded that average best fitness remains almost in similar pattern for five (05) different scenarios.

4.1. Performance Evaluation of GSA

4.1.1. Convergence Analysis. It can be apparently seen that although the algorithm has been set to run for maximum 100 iterations, the convergence happened in about 20 iterations. The result derived in this paper reveals that each object of the standard GSA converges to a stable point. Here, the assumption was that the gravitational and inertia masses are the same. However, for some applications different values for them can be used. A heavier inertia mass provides a slower motion of agents in the search space and hence a more

precise search [20]. On the contrary, a heavier gravitational mass causes a higher attraction of agents. This allows a faster convergence. The analysis results confirm the convergence characteristics of GSA according to the given parameters ranges of the algorithm. Figures 8, 9, 10, 11, and 12 show the convergence behavior of GSA. The best fitness function shows convergences after the same iterations (35 iterations) for both 50 and 100 numbers of PHEVs while for 500 and 1000 numbers of PHEVs, it shows early convergence (before 20 iterations).

4.1.2. Robustness. The similar numeric patterns of average best fitness show the robustness of GSA method. This method resists change without adapting its initial stable configuration

TABLE 3: Fitness evaluation of GSA.

Fitness function $J(k)$	For 50 PHEVs	For 100 PHEVs	For 300 PHEVs	For 500 PHEVs	For 1000 PHEVs
Maximum best fitness	781.1267	579.3955	743.1251	836.2707	968.7652
Average best fitness	158.8289	139.7536	172.4296	152.36437	161.52349
Minimum best fitness	0.2191	3.2523	2.3279	0.9818	7.2747

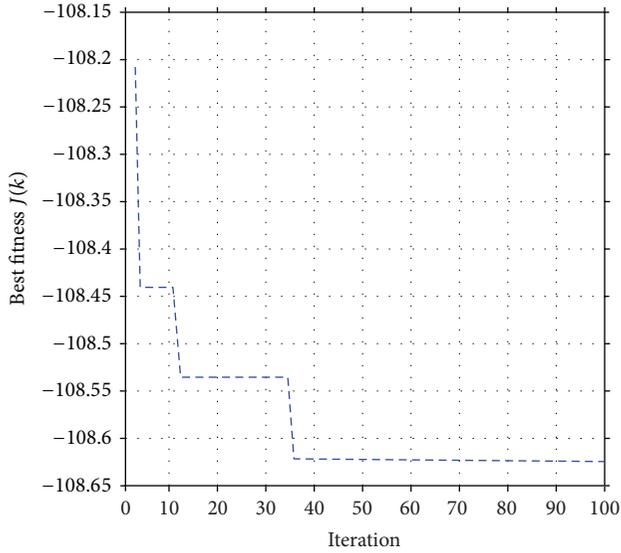


FIGURE 8: Best fitness versus iteration (50 PHEVs).

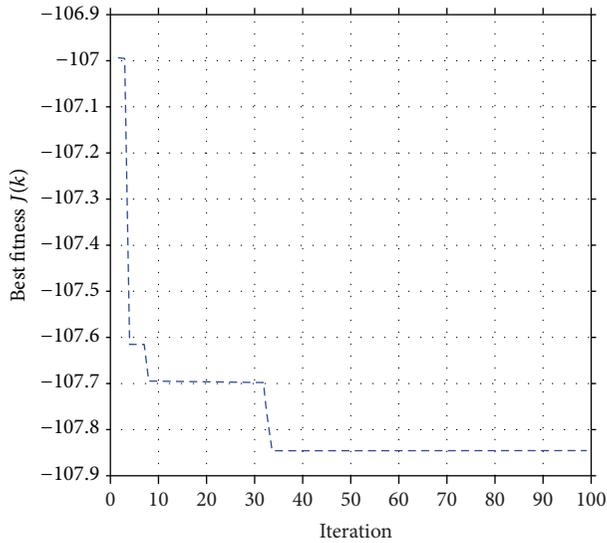


FIGURE 9: Best fitness versus iteration (100 PHEVs).

for different cases (number of PHEVs) which proves GSA as a robust algorithm.

4.1.3. *Diversity.* Here, the average best fitness gives different values with the increment of PHEVs population. The rate of convergence of mass agents in GSA is good through the

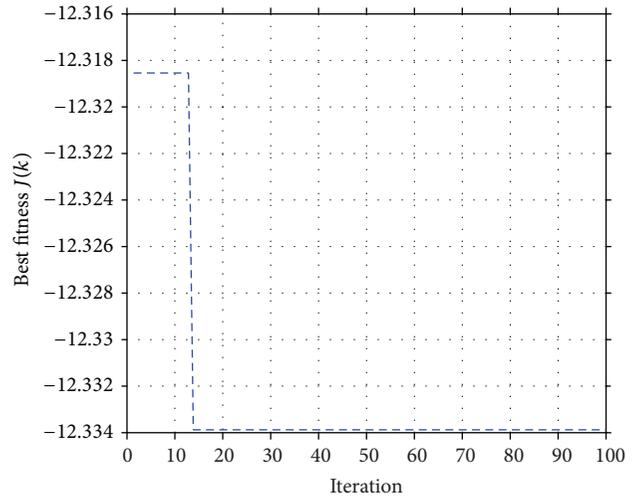


FIGURE 10: Best fitness versus iteration (300 PHEVs).

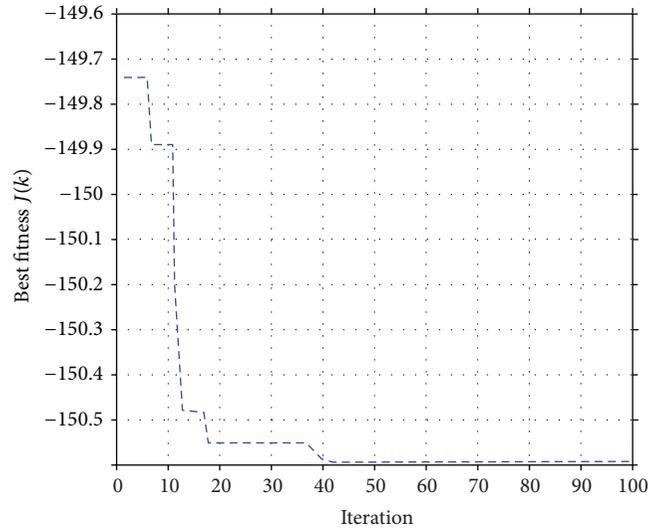


FIGURE 11: Best fitness versus iteration (500 PHEVs).

fast information flowing among mass agents, so its diversity decreases very quickly in the successive iterations and leads to a suboptimal solution.

4.1.4. *Computational Cost.* Here, we measured the computational cost of the algorithm in terms of total running time. Table 6 shows the computational time of GSA for five different scenarios. Here, average CPU time is measured in seconds.

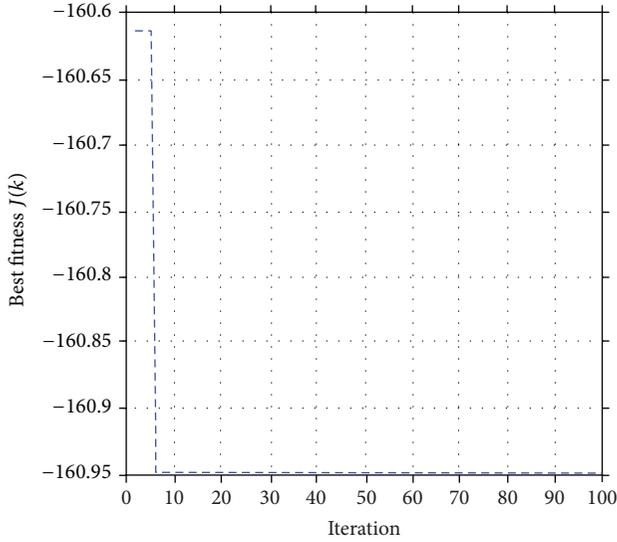


FIGURE 12: Best fitness versus iteration (1000 PHEVs).

As GSA needs a good number of parameter tuning, the computational cost increases with the increment of the total number of PHEVs.

4.1.5. Quality of Solution. When an algorithm finds an optimal solution to a given problem, one of the important factors is speed and rate of convergence to the optimal solution. For heuristics, the additional consideration of how close the heuristic solution comes to optimal is generally the primary concern of the researcher [32]. In GSA, the faster convergence and better exploitation rate ensure good quality solution, which is best fitness function.

For this optimization, the initial state of charge was expressed as a random number which is continuous and uniform between 0.2 and 0.6. The sample time was set around 1200 seconds (20 minutes). The remaining charge time was defined as continuous random number between 0 and 6 hours. The price according to customer's choice for paying the bill for electricity was expressed as a continuous random number which is in between \$1 and \$2.

The capacity of the battery was assumed to be identical for all vehicles.

4.2. Comparison between GSA and PSO. Particle swarm optimization (PSO) with the parameter settings stated in Table 4 was also performed for the same objective function and compared with the performance of gravitational search algorithm in terms of average best fitness. The swarm size and maximum iterations were set exactly the same as those of GSA algorithm for the comparison purpose. The values of parameters $c1$, $c2$, and w were set as standard values, 1.4, 1.4, and 0.9, respectively.

From Figure 13 it is clear that gravitational search algorithm outperformed particle swarm optimization in terms of average best fitness. Starting from 50 numbers of PHEVs up to 1000 PHEVs, GSA shows better fitness value than PSO.

TABLE 4: PSO parameter settings.

Parameters	Values
Size of the swarm	100
Maximum number of steps	100
PSO parameter, $c1$	1.4
PSO parameter, $c2$	1.4
PSO inertia (w)	0.9
Maximum iteration	100
Number of runs	50

TABLE 5: Summarizes the comparisons of GSA with PSO algorithm in terms of average best fitness.

Average best fitness for	PSO	GSA
50 PHEVs	142.839	158.8289
100 PHEVs	171.102	182.3097
300 PHEVs	169.312	172.4296
500 PHEVs	150.869	152.36437
1000 PHEVs	156.802	161.52349

TABLE 6: Computational time for PSO and GSA.

Computational Time (sec.)	PSO	GSA
50 PHEVs	1.650	2.721
100 PHEVs	1.686	4.439
500 PHEVs	1.990	18.165
1000 PHEVs	2.398	36.275

Table 7 illustrates the advantages and disadvantages of both GSA and PSO for solving different optimization problems.

It has been proven that gravitational search algorithm has good ability to search for the global optimum, but it suffers from slow searching speed in the last iterations [35]. Moreover, the inertia mass is against the motion and slows the mass movement. Agents with heavy inertia mass move slowly and hence search the space more locally. So, it can be considered as an adaptive learning rate [34]. GSA is a memory-less algorithm. However, it works competently like the algorithms with memory. Our simulation results show the good convergence rate of the GSA.

5. Conclusion and Recommendations

In this paper, gravitational search algorithm- (GSA-) based optimization was performed in order to optimally allocate power to each of the PHEVs entering into the charging station. A sophisticated controller will need to be designed in order to allocate power to PHEVs appropriately. For this wake, the applied algorithm in this paper is a step towards real-life implementation of such controller for PHEV charging infrastructures.

Here, five (05) different numbers of PHEVs were considered for MATLAB[®] simulation and then obtained results were compared with PSO in terms of average best fitness. The

TABLE 7: Advantages and disadvantages of PSO and GSA.

Optimization method	Advantages	Disadvantages
PSO	Less parameters tuning Easy constraint Good for multiobjective optimization [33]	Low quality solution Needs memory to update velocity Slow convergence rate
GSA	High quality solution Good convergence rate Local exploitation capability [34]	Needs more computational time More parameters tuning

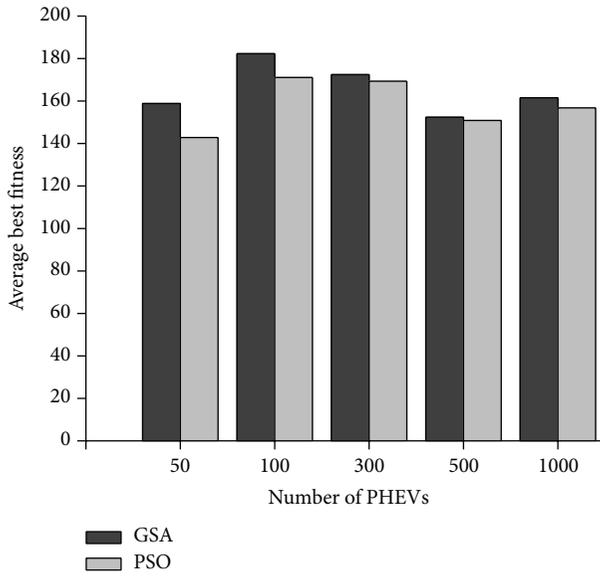


FIGURE 13: Average best fitness versus number of PHEVs.

success of the electrification of transportation sector solely depends on charging infrastructure. Only proper charging control and infrastructure management can assure the larger penetration of PHEVs. The researchers should try to develop efficient control mechanism for charging infrastructure in order to facilitate upcoming PHEVs penetration in highways. In future, more vehicles should be considered for intelligent power allocation strategy and improved versions of GSA and hybrid swarm intelligence based methods should be applied to ensure low computational cost.

Nomenclature and Acronyms

- PHEVs: Plug-in hybrid electric vehicles
- EPRI: Electric Power Research Institute
- V2G: Vehicle to grid
- SoC: State of charge
- ICEVs: Internal combustion engine vehicles
- AEVs: All-electric vehicles
- HEVs: Hybrid electric vehicles
- AER: All-electric range
- $I_i(k)$: Charging current over Δt

- $C_{r,i}(k)$: Remaining battery capacity required to be filled for i th PHEV at time step k
- C_i : Rated battery capacity of the i th PHEV (Farad)
- $T_{r,i}(k)$: Remaining time for charging the i th PHEV at time step k
- $D_i(k)$: Price difference
- $w_i(k)$: Charging weighting term of the i th PHEV at time step
- $\text{SoC}_i(k + 1)$: State of charge of the i th PHEV at time step $k + 1$
- $\text{SoC}_{i,\max}$: User-defined maximum battery SoC limit for the i th PHEV
- P_{utility} : Power available from the utility
- $P_{i,\max}$: Maximum power that can be absorbed by a specific PHEV
- η : Overall charging efficiency of the charging station.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors would like to thank Universiti Teknologi Petronas for supporting the research under Graduate Assistance Scheme. This research paper is financially sponsored by the Centre of Graduate study with the support of the Department of Fundamental & Applied Sciences, Universiti Teknologi Petronas.

References

- [1] On certain integrals International Energy Agency (IEA), Transport, Energy and CO2-Moving Towards Sustainability, 2009.
- [2] D. Holtz-Eakin and T. M. Selden, "Stoking the fires? CO₂ emissions and economic growth," *Journal of Public Economics*, vol. 57, no. 1, pp. 85–101, 1995.
- [3] S. F. Tie and C. W. Tan, "A review of energy sources and energy management system in electric vehicles," *Renewable and Sustainable Energy Reviews*, vol. 20, pp. 82–102, 2013.
- [4] "Environmental assessment of plug-in hybrid electric vehicles, Volume 1: Nationwide greenhouse gas emissions," Tech. Rep.

- 1015325, Electric Power Research Institute (EPRI), Palo Alto, Calif, USA, 2007.
- [5] H. Lund and W. Kempton, "Integration of renewable energy into the transport and electricity sectors through V2G," *Energy Policy*, vol. 36, no. 9, pp. 3578–3587, 2008.
 - [6] A. R. Hota, M. Juvvanapudi, and P. Bajpai, "Issues and solution approaches in PHEV integration to the smart grid," *Renewable and Sustainable Energy Reviews*, vol. 30, pp. 217–229, 2014.
 - [7] J. Soares, T. Sousa, H. Morais, Z. Vale, B. Canizes, and A. Silva, "Application-Specific Modified Particle Swarm Optimization for energy resource scheduling considering vehicle-to-grid," *Applied Soft Computing Journal*, 2013.
 - [8] W. Su and M.-Y. Chow, "Computational intelligence-based energy management for a large-scale PHEV/PEV enabled municipal parking deck," *Applied Energy*, vol. 96, pp. 171–182, 2012.
 - [9] W. Su and M. Y. Chow, "Performance evaluation of a PHEV parking station using particle swarm optimization," in *Proceedings of the IEEE PES General Meeting*, pp. 1–6, July 2011.
 - [10] W. Su and M.-Y. Chow, "Performance evaluation of an EDA-based large-scale plug-in hybrid electric vehicle charging algorithm," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 308–315, 2012.
 - [11] K. Morrow, D. Karner, and J. Francfort, "Plug-in hybrid electric vehicle charging infrastructure review," Tech. Rep., The Idaho National Laboratory, 2008.
 - [12] "Investigation into the scope for the transport sector to switch to electric vehicles and plug- in hybrid vehicles," Tech. Rep., Department for Transport, London, UK, 2008.
 - [13] D. Mayfield, "Site Design for Electric Vehicle Charging Stations, ver. 1.0," Sustainable Transportation Strategies, July 2012.
 - [14] G. Boyle, *Renewable Electricity and the Grid: The Challenge of Variability*, Earthscan Publications, London, UK, 2007.
 - [15] A. Hess, F. Malandrino, M. B. Reinhardt, C. Casetti, K. A. Hummel, and J. M. Barceló-Ordinas, "Optimal deployment of charging stations for electric vehicular networks," in *Proceedings of the 1st ACM Conference on Urban Networking (UrbaNe '12)*, pp. 1–6, New York, NY, USA, December 2012.
 - [16] Z. Li, Z. Sahinoglu, Z. Tao, and K. H. Teo, "Electric vehicles network with nomadic portable charging stations," in *Proceeding of the IEEE 72nd Vehicular Technology Conference Fall (VTC '10)*, pp. 1–5, Ottawa, Canada, September 2010.
 - [17] T. Ghanbarzadeh, S. Goleijani, and M. P. Moghaddam, "Reliability constrained unit commitment with electric vehicle to grid using hybrid particle swarm optimization and ant colony optimization," in *Proceeding of the IEEE PES General Meeting: The Electrification of Transportation and the Grid of the Future*, pp. 1–7, San Diego, Calif, USA, July 2011.
 - [18] A. Shafiei and S. S. Williamson, "Plug-in hybrid electric vehicle charging: current issues and future challenges," in *Proceedings of the IEEE Vehicle Power and Propulsion Conference (VPPC '10)*, pp. 1–8, September 2010.
 - [19] J. Yan, C. Li, G. Xu, and Y. Xu, "A novel on-line self-learning state-of-charge estimation of battery management system for hybrid electric vehicle," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 1161–1166, June 2009.
 - [20] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.
 - [21] D. Martens, B. Baesens, and T. Fawcett, "Editorial survey: swarm intelligence for data mining," *Machine Learning*, vol. 82, no. 1, pp. 1–42, 2011.
 - [22] J. Krause, J. Cordeiro, R. S. Parpinelli, and H. S. Lopes, "A survey of swarm algorithms applied to discrete optimization problems," in *Swarm Intelligence and Bio-inspired Computation: Theory and Applications*, pp. 169–191, Elsevier Science & Technology Books, 2013.
 - [23] O. Okobiah, S. P. Mohanty, and E. Kougiianos, "Geostatistics inspired fast layout optimization of nanoscale CMOS phase locked loop," in *Proceedings of the 14th International Symposium on Quality Electronic Design (ISQED '13)*, pp. 546–551, Santa Clara, Calif, USA, March 2013.
 - [24] W.-Y. Chang, "The state of charge estimating methods for battery: a review," *ISRN Applied Mathematics*, vol. 2013, Article ID 953792, 7 pages, 2013.
 - [25] T. Ganesan, I. Elamvazuthi, K. Z. Ku Shaari, and P. Vasant, "Swarm intelligence and gravitational search algorithm for multi-objective optimization of synthesis gas production," *Applied Energy*, vol. 103, pp. 368–374, 2013.
 - [26] O. T. Altinoz and A. E. Yilmaz, "Calculation of optimized parameters of rectangular patch antenna using gravitational search algorithm," in *Proceedings of the International Symposium on INnovations in Intelligent SysTems and Applications (INISTA '11)*, pp. 349–353, IEEE, June 2011.
 - [27] Ö. T. Altinöz, A. E. Yilmaz, and G. W. Weber, "Orthogonal array based performance improvement in the gravitational search algorithm," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 21, no. 1, pp. 174–185, 2013.
 - [28] N. M. Sabri, M. Puteh, and M. R. Mahmood, "A review of gravitational search algorithm," *International Journal of Advances in Soft Computing and Its Applications*, vol. 5, no. 3, pp. 1–39, 2013.
 - [29] R. E. Haskell, G. Castelino, and B. Mirshab, "Efficient algorithm for locating the global maximum of an arbitrary univariate function," *Journal of Forth Application and Research*, vol. 5, no. 3, pp. 357–364, 1989.
 - [30] M. Udgir, H. M. Dubey, and M. Pandit, "Gravitational search algorithm: a novel optimization approach for economic load dispatch," in *Proceedings of the Annual International Conference on Emerging Research Areas and International Conference on Microelectronics, Communications and Renewable Energy (AICERA/ICMiCR '13)*, pp. 1–6, 2013.
 - [31] A. A. Bazmi and G. Zahedi, "Sustainable energy systems: role of optimization modeling techniques in power generation and supply—a review," *Renewable and Sustainable Energy Reviews*, vol. 15, pp. 3480–3500, 2011.
 - [32] R. S. Barr, B. L. Golden, J. P. Kelly, M. G. C. Resende, and W. R. Stewart Jr., "Designing and reporting on computational experiments with heuristic methods," *Journal of Heuristics*, vol. 1, no. 1, pp. 9–32, 1995.
 - [33] Y.-T. Kao and E. Zahara, "A hybrid genetic algorithm and particle swarm optimization for multimodal functions," *Applied Soft Computing*, vol. 8, pp. 849–857, 2008.
 - [34] I. Rahman, P. M. Vasant, B. S. M. Singh, and M. Abdullah-Al-Wadud, "Intelligent energy allocation strategy for PHEV charging station using gravitational search algorithm," *AIP Conference Proceedings*, vol. 1621, pp. 52–59, 2014.
 - [35] S. Mirjalili, S. Z. Mohd Hashim, and H. Moradian Sardroudi, "Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm," *Applied Mathematics and Computation*, vol. 218, no. 22, pp. 11125–11137, 2012.

Research Article

Multipeak Mean Based Optimized Histogram Modification Framework Using Swarm Intelligence for Image Contrast Enhancement

P. Babu,¹ V. Rajamani,² and K. Balasubramanian¹

¹ Department of MCA, PSNA College of Engineering and Technology, Dindigul 624622, India

² Vel Tech Multitech Dr. Rangarajan Dr. Sakunthala Engg. College, Vel Tech, Avadi, Chennai 600001, India

Correspondence should be addressed to P. Babu; pbabu@psnacet.edu.in

Received 6 June 2014; Revised 28 July 2014; Accepted 11 August 2014

Academic Editor: Fang Zong

Copyright © 2015 P. Babu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A novel approach, Multipeak mean based optimized histogram modification framework (MMOHM) is introduced for the purpose of enhancing the contrast as well as preserving essential details for any given gray scale and colour images. The basic idea of this technique is the calculation of multiple peaks (local maxima) from the original histogram. The mean value of multiple peaks is computed and the input image's histogram is segmented into two subhistograms based on this multipeak mean (*mmean*) value. Then, a bicriteria optimization problem is formulated and the subhistograms are modified by selecting optimal contrast enhancement parameters. While formulating the enhancement parameters, particle swarm optimization is employed to find optimal values of them. Finally, the union of the modified subhistograms produces a contrast enhanced and details preserved output image. This mechanism enhances the contrast of the input image better than the existing contemporary HE methods. The performance of the proposed method is well supported by the contrast enhancement quantitative metrics such as discrete entropy, natural image quality evaluator, and absolute mean brightness error.

1. Introduction

Contrast enhancement plays an important role in the improvement of visual quality for computer vision, pattern recognition, and the processing of digital images. Poor contrast in digital video or images can result from many circumstances, including lack of operator expertise and inadequacy of the image capture device [1]. Contrast enhancement of an image is achieved through redistribution of intensity values. The resultant contrast enhanced image provides feature extraction in computer vision system. Histogram equalization (HE) is one of the commonly used algorithms for contrast enhancement due to its simplicity and effectiveness. It remaps the gray levels based on the probability distribution of the input gray levels. It flattens and stretches the dynamic range of the images histogram which results in overall contrast improvement [2].

HE methods can be categorized into two methods: improved global HE methods (GHE) and adaptive HE (AHE)

methods. The GHE methods improve image quality by extending the dynamic range of intensity using the histogram of the whole image. Since GHE is based on the intensity distribution of the whole image, it causes washed out effect and changes average intensity to middle level [3]. In AHE methods, the equalization is based on the histogram and statistics obtained from neighbourhood around each pixel. These methods can usually provide stronger enhancement effects than global methods. They divide the original image into several nonoverlapped subblocks and perform HE on individual subblocks. The resultant image is produced by merging the subblocks using bilinear interpolation method. However, due to their high computational load, AHE methods are not well suited for real time video applications [4].

The objective of an image enhancement technique is to bring out hidden image details or to increase the contrast of an image with the low dynamic range. Such a technique produces an output image that subjectively looks better than the

original image by increasing the gray-level differences among objects and background. Numerous enhancement techniques have been introduced and can be divided into three groups: (1) techniques that decompose an image into high and low frequency signals for manipulation, (2) transform based techniques, and (3) histogram modification techniques. Among the three groups, the third group received the most attention due to their straightforward and intuitive implementation qualities [5]. Some research works have focused on improving HE based contrast enhancement such as brightness preserving bihistogram equalization (BBHE), equal area dualistic subimage histogram equalization (DSIHE), recursive mean separate histogram equalization (RMSHE), weighted thresholded histogram equalization (WTHE), range limited bihistogram equalization (RLBHE), efficient contrast enhancement using adaptive gamma correction with weighting distribution (AGCWD), and contrast enhancement based on layered difference representation of 2D histograms (LDR). Efficient histogram modification using bilateral Bezier curve for the contrast enhancement (BBC) has been proposed in the past years. The aforementioned techniques may create problems when the histogram has spikes or when a natural looking enhanced image is required. The detailed literature of these techniques was given in next section.

This paper uses the particle swarm optimization (PSO) and multipeak mean (*mmean*) value for the segmentation of histogram and hence trying to improve the contrast enhancement with the help of optimal threshold value as the enhancement parameters. The technique, multipeak mean based optimized histogram modification framework using particle swarm optimization (MMOHM) is proposed for the purpose of enhancing the contrast as well as preserving essential details for any given input image. Multiple peaks (local maxima) are identified from the input histogram. Then, the mean value of multipeaks is computed and the input image's histogram is segmented into two subhistograms based on this multipeak mean value. Then, a bicriteria optimization problem is formulated to satisfy aforementioned requirements and the subhistograms are modified by selecting the optimal contrast enhancement parameters. Finally, the union of the modified subhistograms produces a contrast enhanced and details preserved output image. While formulating the optimization problem, PSO is employed to find the optimal values of contrast enhancement parameters.

The traditional HE and several HE based methods are analyzed in Section 2. Section 3 presents the proposed MMOHM method along with the statistical measurements to measure the image quality. The results are discussed in Section 4. The conclusion is given in Section 5.

2. Review of Histogram Equalization Methods

Consider the input image $X = \{X(i, j)\}$, where $X(i, j)$ denotes the gray-level of a pixel at (i, j) . The total number of pixels in the image is N and the image intensity is digitized into L gray levels that are $[X_0, X_1, \dots, X_{L-1}]$. If n_k represents the number of times that the level X_k appears in the input image X , then

the probability density function (PDF) $P(X_k)$ for the level X_k is defined as

$$P(X_k) = \frac{n_k}{N}, \quad k = 0, 1, \dots, L-1. \quad (1)$$

Based on (1), the cumulative density function (CDF) is defined as

$$c(X) = \sum_{j=0}^k P(X_j). \quad (2)$$

HE maps the input image into the entire dynamic range, (X_0, X_{L-1}) , by using the CDF as a transform function. The transform function $f(X)$ based on the CDF is defined as

$$f(X) = X_0 + (X_{L-1} - X_0)c(X). \quad (3)$$

HE technique is rarely used because it flattens the histogram of an image which results in bringing significant change in brightness and causes undesirable artifacts. Kim has proposed a method known as brightness preserving bihistogram equalization (BBHE) to preserve the brightness [6]. BBHE decomposes the input histogram into two subhistograms based on its input mean X_m . It is clearly proved that BBHE can preserve the original brightness to a certain extent. Wang et al. proposed a method called equal area dualistic subimage histogram equalization (DSIHE) which is an extension of BBHE [7]. DSIHE differs from BBHE only in the segmentation process. The input image is segmented into two subimages based on median rather than mean. This method is well suited for some images but fails to preserve original brightness for most of the images. Chen and Ramli proposed a method called minimum mean brightness error bihistogram equalization (MMBEBHE) which is an extension of BBHE [8]. It performs the separation process based on the threshold level and it preserves the original brightness of the image. Chen and Ramli proposed a method called recursive mean separate histogram equalization (RMSHE) in which histogram of the given image is partitioned recursively [9]. Unlike BBHE which decomposes the input histogram only once, RMSHE decomposes it recursively up to a recursion level r and thereby generating 2^r subhistograms. The resultant subhistograms are then equalized individually to get the contrast enhanced image. Sim et al. proposed a similar method called recursive subimage histogram equalization (RSIHE) [10]. It shares similar recursive framework with RMSHE except that for histogram segmentation, RSIHE uses the median of subhistograms instead of the mean of subhistograms in RMSHE. But, the fact is that as the recursion level increases, the computational complexity also increases and finding an optimal recursion level is a difficult task for all such methods. A detailed study of various bilevel and multilevel partitioning methods are analyzed in [11]. A fast and effective method for video and image contrast enhancement known as weighted thresholded histogram equalization (WTHE) was presented in [4]. WTHE provides a good tradeoff between the two features, adaptively to different images and ease of control, which is difficult to achieve through GHE based enhancement methods. In recursively separated and

weighted HE (RSWHE) and weight clustering HE (WCHE), different weighing principles are applied successfully [12, 13]. But these methods fail to preserve the spatial relationship among the pixels and their surroundings. Ibrahim and Pik Kong proposed subregions histogram equalization (SRHE) for sharpening the images [14]. Zuo et al. developed the range limited bihistogram equalization (RLBHE) for image contrast enhancement in which the input image's histogram is divided into two independent subhistograms through a threshold that minimizes the intraclass variance [15]. Then the range of the equalized image is calculated to yield minimum absolute mean brightness error between the original and equalized image. Sundaram et al. proposed a method called histogram modified local contrast enhancement for mammogram images in which the contrast enhancement of the mammogram image can be achieved by histogram modification and local contrast enhancement [16]. BBHE is combined with local enhancement to preserve the brightness and improve performance of detail preservation. It combines spatial edge information with gray information to enhance the local details [17]. In thresholded and optimized histogram equalisation (TOHE), histogram modification was carried out based on otsu's optimality principle. Then, weighing constraints are applied; mean errors are calculated and the error values were added to the original probability density values for contrast enhancement [18]. Lee and Kim proposed a novel contrast enhancement technique based on layered difference representation (LDR) of 2D histograms [19]. They attempt to enhance the contrast by amplifying the gray-level differences between the adjacent pixels. A constrained optimization problem is formulated based on the observation that the gray-level differences, occurring more frequently in the input image, should be more emphasized in the output image. The optimization problem is solved to derive the transformation function at each layer and then combine the transformation functions at all layers to map input gray levels to output gray levels. Huang et al. proposed an automatic histogram transformation technique called efficient contrast enhancement using adaptive gamma correction with weighting distribution (AGCWD) that improves the brightness of dimmed images via gamma correction and probability distribution of luminance pixels [1]. To enhance the video, AGCWD uses temporal information regarding the differences between the discrete entropy values of each frame to reduce the computational complexity. Efficient histogram modification using bilateral Bezier curve for the contrast enhancement (BBC) is proposed to enhance the quality of the input image and reduce the processing time [20]. The control points of the mapping curve are automatically calculated by Bezier curve which performs in dark and bright regions separately. Using the fast and accurate histogram modification allows this method to transform the intensity for both image and video. Ghita et al. introduced a new variational approach for HE which involves the application of the total variation minimization with a L^1 fidelity (TV- L^1) model to achieve cartoon-texture decomposition [21]. The texture information is also employed along with the intensity information to emphasize the contribution of local textural features in the contrast enhancement process. This is achieved

by implementing a nonlinear histogram warping strategy that will maximize the information content in the transformed image. Celik and Tjahjadi proposed an algorithm for contrast enhancement which is free from parameter setting [5]. In this method, the pixel values of an input image are modeled using Gaussian mixture model. The intersection points of the Gaussian components are used in partitioning the dynamic range of the image into input gray-level intervals. The gray levels in each input interval are transformed according to the dominant Gaussian component and the CDF of the interval to obtain the contrast equalized image.

3. Multipeak Mean Based Optimized Histogram Modification (MMOHM)

HE is the simple and straightforward method for image contrast enhancement. Several HE based techniques were proposed in the past years and a detailed discussion on it is given in the literature. Most of the contemporary HE based techniques have the common drawback of mean shift, which results in brightness degradation, lack of details preservation, and overenhancement. In addition, computational complexity and controllable contrast enhancement become an important issue when the goal is to design a contrast enhancement algorithm for gray scale and color images. The main objective of this paper is to obtain a visually pleasing brightness preserved enhancement method, which incorporates a provision to have a control over the level of contrast enhancement and works well for all types of images. The proposed MMOHM method is an effective technique to combat with the aforementioned pitfalls. MMOHM produces contrast enhanced as well as details preserved output image with the help of the following steps.

3.1. Identification of Peaks and Mean Computation. The first step of the proposed method finds the local maximum points of the histogram by tracing the histogram of the given input image. Generally, peak in the histogram specifies the highest occurrence of some specific gray valued pixel. The histograms of consumer electronics images in general consist of many peaks and, hence, it is desirable to enhance the image around its peaks, which is of prime importance in order to have controlled contrast enhancement and brightness preservation. A point on the histogram is a peak value (local maximum) if its amplitude is more than its neighbors. In order to obtain the peak point, the signs of the difference between two successive probabilities in the histogram are calculated. Let r denote a random variable representing discrete gray levels in the range $[0, L - 1]$ and let $P(r_i)$ denote the peak value corresponding to the i th value of r . Then, the $mmean$ value is calculated as

$$mmean = \frac{\sum_{i=0}^{NP} P(r_i)}{NP}, \quad (4)$$

where NP represents the total number of peaks in the input image histogram. While doing this, if the histogram value is found lower than the $mmean$ value, then it is made to reach the $mmean$ value by increasing it or if the histogram value is higher than the $mmean$ value, it is made to decrease its

value to reach the $mmean$ value. Based on $mmean$ value, the input histogram $H(X)$ is segmented into two subhistograms, namely, $H_L(X)$ and $H_U(X)$, where the lower subhistogram $H_L(X)$ is associated with gray levels ranging from minimum to $mmean$, that is, $\{X_0, X_1, \dots, mmean\}$ and the upper subhistogram $H_U(X)$ is associated with $mmean$ gray-level to maximum gray-level, that is, $\{mmean + 1, \dots, X_{L-1}\}$. This type of histogram partitioning helps to avoid some portions of the histogram from being dominated by other portions. The motivation of computing $mmean$ value is to improve the mean image brightness preserving capability.

3.2. Lower Subhistogram Modification. HE often produces overenhanced, unnatural looking in the output image which leads to loss of information in the original image. Another problem with HE is its large backward difference values of mapping functions. When the input histogram distribution is already uniform, the mapping obtained from cumulative distribution is $C(X_k) = X_k$, which identically maps input to output. In order to find the level of contrast enhancement, the input histogram H_{iL} can be altered so that the modified histogram H_{mL} is closer to uniform histogram U_{iL} of the lower subimage. It also aims to make the difference between the histograms of modified and input image ($H_{mL} - H_{iL}$) small, resulting in the increased potentiality of image contrast enhancement. This is a bicriteria optimization problem since the optimization enhances the contrast of the input image while preserving the details of the original image. So, the output image would be more relevant to the input image. The optimization problem for lower subimage can be defined as

$$\min \|H_{mL} - H_{iL}\| + \varphi_1 \|H_{mL} - u_{iL}\|. \quad (5)$$

The modified histogram for lower subimage can be obtained by finding an analytical solution for (5) as follows:

$$(H_{mL} - H_{iL}) + \varphi_1 (H_{mL} - u_{iL}) = 0 \quad \text{That is,} \quad (6)$$

$$H_{mL} - H_{iL} + \varphi_1 H_{mL} - \varphi_1 u_{iL} = 0.$$

Equation (6) can be rewritten as

$$H_{mL} (1 + \varphi_1) = H_{iL} + \varphi_1 u_{iL},$$

$$H_{mL} = \frac{H_{iL} + \varphi_1 u_{iL}}{1 + \varphi_1} = \left(\frac{1}{1 + \varphi_1} \right) H_{iL} + \left(\frac{\varphi_1}{1 + \varphi_1} \right) u_{iL}, \quad (7)$$

where H_{mL} , H_{iL} , u_{iL} , and φ_1 are the modified histogram, input histogram, uniform histogram, and the contrast enhancement parameter for lower subhistogram, respectively.

3.3. Upper Subhistogram Modification. The main objective of this method is to find a modified histogram for upper subhistogram that is closer to uniform histogram and to make the difference between histograms of modified image and original image ($H_{mU} - H_{iU}$) small, which results in increasing the potentiality of image contrast enhancement. Then, the bicriteria optimization problem for upper subimage can be written as

$$\min \|H_{mU} - H_{iU}\| + \varphi_2 \|H_{mU} - u_{iU}\|. \quad (8)$$

The modified histogram for upper subimage can be obtained by finding an analytical solution for (8) as follows:

$$(H_{mU} - H_{iU}) + \varphi_2 H_{mU} - \varphi_2 u_{iU} = 0. \quad (9)$$

Equation (9) can be rewritten as

$$H_{mU} = \frac{H_{iU} + \varphi_2 u_{iU}}{1 + \varphi_2} = \left(\frac{1}{1 + \varphi_2} \right) H_{iU} + \left(\frac{\varphi_2}{1 + \varphi_2} \right) u_{iU}, \quad (10)$$

where H_{mU} , H_{iU} , u_{iU} , and φ_2 are the modified histogram, input histogram, uniform histogram, and the contrast enhancement parameter for upper subhistogram, respectively.

The value of the contrast enhancement parameters, φ_1 and φ_2 , is in the range from 0.0 to 1.0 practically in order to avoid overenhancement. For very low value of these parameters, the mapping function gets saturated leading to overenhancement in the output image. When the value of enhancement parameters is zero, this method tends to behave like a traditional HE. When it is nearer to 1.0, the mapping gradually reaches a maximum value, which preserves the naturalness of the image with increased image quality. When these values are greater than 1.0, the mapping function closely reaches identity mapping which means there is no difference between the original image and output image, resulting in no contrast enhancement. The various changes in the levels of contrast enhancement can be achieved by changing the values of these parameters. The optimal value of φ_1 and φ_2 for both subhistograms can be obtained by using PSO. The MMOHM procedure is given in Algorithm 1.

The flowchart representation of MMOHM process is given in Figure 1

3.4. Optimizing the Contrast Enhancement Parameters Using PSO. In the proposed technique, the parameters identified to be optimized are φ_1 and φ_2 , as these parameters play an important role in controlling the degree of enhancement. PSO is used to find the optimal value of these parameters, since PSO takes lesser time to converge into better optima than the other evolutionary soft computing techniques [18]. PSO is one of the evolutionary computational techniques based on movement and intelligence of swarms looking for most fertile feeding location. A "swarm" is a disorganized collection (population) of moving individuals that tend to cluster together while each individual seems to be moving in a random direction. It uses a number of agents (particles) that constitute swarm moving around in search space looking for the best solution. Each particle is treated as a point in a two-dimensional space which adjusts its "flying" according to its own flying experience as well as the flying experience of other particles. Each particle keeps track of its coordinates in the problem space which are associated with best solution (fitness) that has been achieved so far. This value is called p_{best} . Another best value that is tracked by the PSO is the maximum value obtained so far by any particle among the neighboring particle. This value is called g_{best} . The PSO concept consists of changing the velocity of each particle

Input: An image with a total number of N pixels in the gray-level range $[X_0, X_{L-1}]$, φ_1, φ_2 .
 Output: The contrast enhanced, details preserved image
BEGIN
Step 1. Segment the input image into two sub-images (lower and upper sub-histogram of the object) based on multi-peak mean value.
Step 2. Generate the input histograms H_{iL} and H_{iU} for lower and upper sub-images separately.
Step 3. Find an uniform histogram u_{iL} for lower sub-image using (1), (2) and (3).
Step 4. Obtain an optimal value of the contrast enhancement parameters φ_1 and φ_2 for lower and upper sub-images using optimization_PSO procedure.
Step 5. Compute the modified histogram H_{mL} using the analytical solution of (7)
Step 6. For upper sub-image image, obtain an uniform histogram u_{iU} using (1), (2) and (3).
Step 7. Compute the modified histogram for upper sub-image H_{mU} using (10)
Step 8. Merge the two modified sub-histograms into a single histogram and display the final contrast enhanced and details preserved output image H_o is given as.
 $H_o = H_{mL} \cup H_{mU}$
END

ALGORITHM 1: MMOHM procedure.

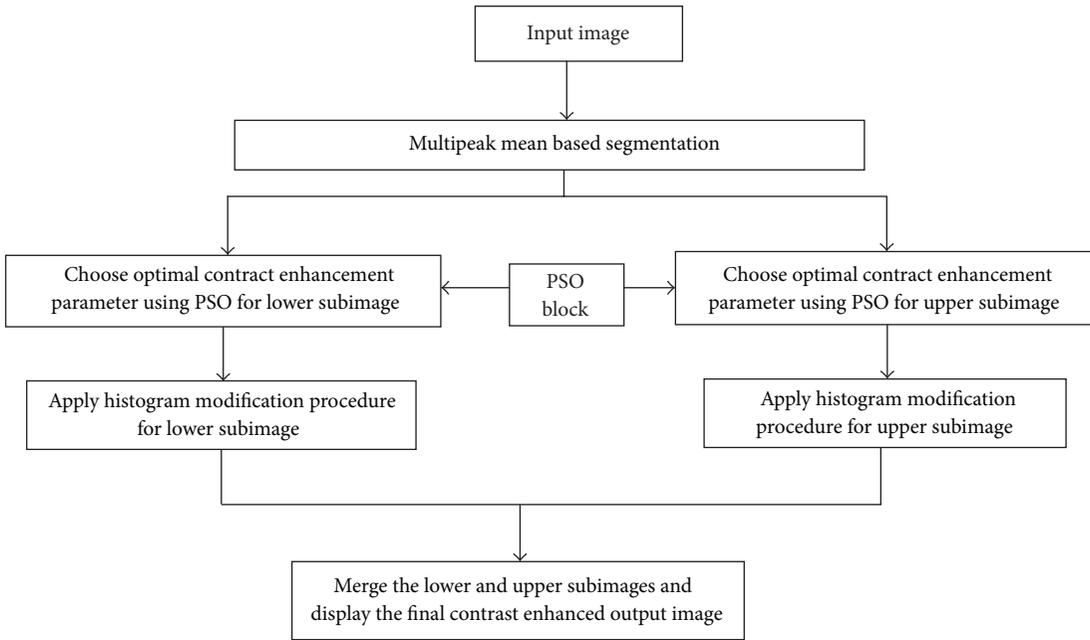


FIGURE 1: Flow chart for MMOHM.

toward its p_{best} and the g_{best} position at each time step. The velocity of each particle is modified by using the following equation:

$$v_{n+1} = v_n + c_1 \text{rand1}() \times (p_{best,n} - \text{currentpos}_n) + c_2 \text{rand2}() \times (g_{best,n} - \text{currentpos}_n), \quad (11)$$

where v_{n+1} is the velocity of particle at $n + 1$ th iteration; V_n is the velocity of particle at n th iteration; c_1 and c_2 are the acceleration factors related to g_{best} and p_{best} ; $\text{rand1}()$ and $\text{rand2}()$ are the random numbers between 0 and 1; g_{best} is the g_{best} position of swarm; p_{best} is the p_{best} position of particle.

Using the modified velocity, the particle's position is updated as

$$\text{currentpos}_{n+1} = \text{currentpos}_n + v_{n+1}, \quad (12)$$

where currentpos_{n+1} is the position of particle at $n + 1$ th iteration; currentpos_n is the position of particle at n th iteration; v_{n+1} is the particle velocity at $n + 1$ th iteration.

The optimized values of φ_1 and φ_2 are found using Algorithm 2.

Two iterations are concerned with PSO algorithm in which the former is involved in processing the number of generations and the latter is used for processing the size of the particles. The performance of this proposed method is

```

INPUT: Image with a total number of  $N$  pixels in the gray-level range  $[X_0, X_{L-1}]$ 
Output: Optimal value of  $\varphi_1$  and  $\varphi_2$ 
BEGIN
Step 1. For each particle
    (a) Initialize particle with feasible random number
    End
Step 2. Do
    (a) For each particle
        (i) Calculate the fitness value (i.e) find the difference between Discrete Entropy
            values of original and enhanced image.
        (ii) If the fitness value is better than the best fitness value ( $pbest$ ) in history then
            Set current value as the new  $pbest$ 
        End
    (b) Choose the particle with the best fitness value of all the particles as the  $gbest$ 
    (c) For each particle
        (i) Calculate particle velocity according to velocity update (11)
        (ii) Update particle position according to position update (12)
    End
    While (maximum iterations are not attained);
Step 3. Output the new population with the optimal values of enhancement parameters  $\varphi_1$  and  $\varphi_2$ 
Step 4. Stop
END.

```

ALGORITHM 2: Optimization_PSO procedure.

decided by number of generations and size of the particles. So, the computational complexity of the proposed method is $O(n^2)$. This process is repeated until a predefined termination condition is satisfied, such as the expiration of a maximum number of iterations. Even though the time complexity is more, the selection of PSO is done because of its incredible performance for finding the optimal values of the enhancement parameters for better contrast enhancement. The PSO algorithm is renowned for its implementation simplicity as there is no need to simulate evolutionary operations such as selection, crossover, and mutation. Another salient feature of the PSO is that the best solution is always maintained without explicitly applying elitism; thus a higher diversification can be applied during the search. This algorithm returns the optimal value of the enhancement parameters once it is terminated.

3.5. Image Quality Assessment. The image quality assessment (IQA) aims to use computational models so as to measure the image quality consistently with subjective evaluations. An IQA index used in the evaluation includes discrete entropy (DE) [22], absolute mean brightness error (AMBE) [16], and natural image quality evaluator (NIQE) [23]. Among these measures, the DE can be considered to be an objective function since it provides better tradeoff than the other measures. NIQE and AMBE can be used as supportive measures to evaluate the degree of enhancement and brightness preservation. A detailed discussion of IQA metrics used in the proposed method is given in the following subsections.

3.5.1. Discrete Entropy (DE). DE is used to measure the richness of details in the image after enhancement, where

an original DE value is obtained in an enhanced image. It indicates that more information is brought out from the images. It is defined as

$$H = - \sum_{k=0}^{L-1} p(k) \log_2(p(k)), \quad (13)$$

where $p(k)$ is the probability density function of the k th gray-level. When the entropy value of an enhanced image is closer to that of the original image, it is considered that the details of the input image are preserved in the enhanced image. The performance of the HE based methods are evaluated by using 80 test images.

3.5.2. Natural Image Quality Evaluator (NIQE). NIQE is used for measuring image quality. It is mainly based on the construction of a “quality aware” collection of statistical features which are again based on simple and successful space domain natural scene statistic (NSS) model. These features are derived from a corpus of natural, undistorted images. The quality of a given test image is expressed as the distance between a multivariate Gaussian (MVG) fit of the NSS features extracted from the test image and a MVG model of the quality aware features extracted from the corpus of natural images and it is given by

$$D(v_1, v_2, \varepsilon_1, \varepsilon_2) = \sqrt{(v_1 - v_2)^T \left(\frac{\varepsilon_1 + \varepsilon_2}{2} \right)^{-1} (v_1 - v_2)}, \quad (14)$$

where v_1 , v_2 and ε_1 , ε_2 are the mean vectors and covariance matrices of the natural MVG model and the distorted image’s MVG model. When the NIQE value of an enhanced image is smaller than that of the original image, it is considered as better image quality [23].

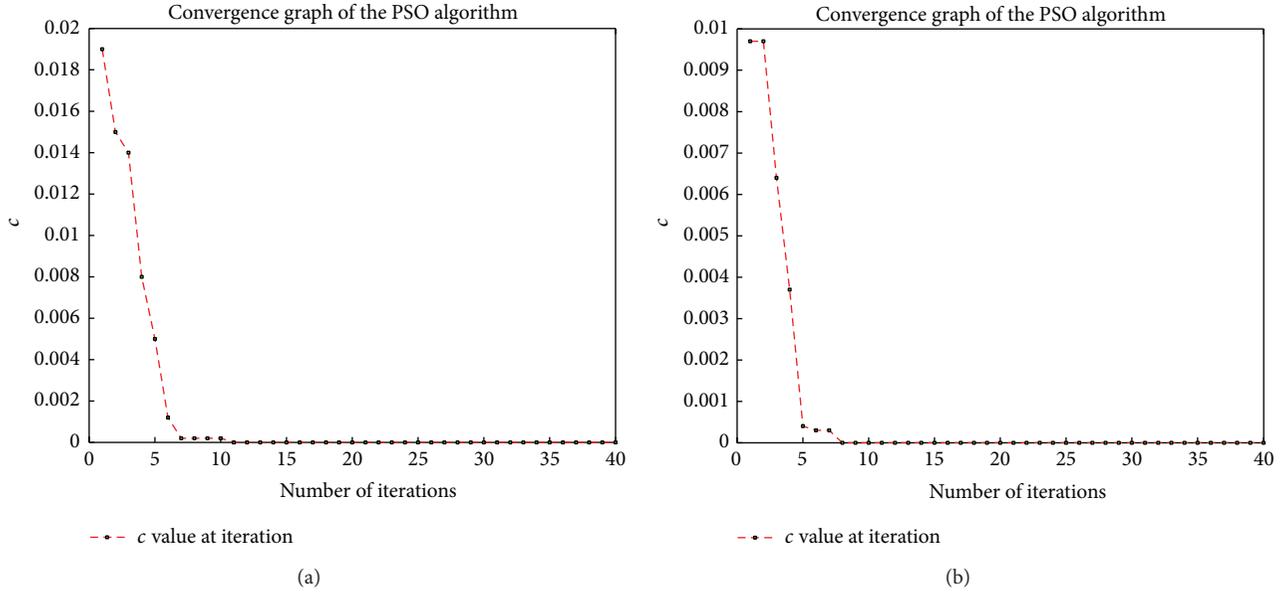


FIGURE 2: Convergence graph for (a) Einstein image and (b) truck image.

3.5.3. *Absolute Mean Brightness Error (AMBE)*. AMBE is used to measure whether the proposed method preserves the original brightness in the enhanced image and it is defined as

$$AMBE(X, Y) = E_n(X) - E_n(Y), \quad (15)$$

where $E_n(X)$ is the average intensity of input image and $E_n(Y)$ is the average intensity of the corresponding enhanced output image. AMBE provides a sense of how the image global appearance gets changed. A median value of AMBE indicates better brightness preservation. Either a very low or a high AMBE value indicates poor performance in the case of contrast enhancement [16].

3.6. *Extending the Proposed Method to Colour Images*. The similar approach extending to the coloured images uses their luminance component only and preserves the chrominance components. Another approach for color image enhancement is to multiply the chrominance values with the ratio of their input and output luminance values to preserve the hue. The former approach is employed in this paper, where RGB image is transformed to the CIE $L^*a^*b^*$ color space and the luminance component L^* is processed for contrast enhancement. The inverse transformation is then applied to obtain the contrast enhanced RGB image. Hence, the proposed method produces better contrast enhancement for color images also.

4. Results and Discussion

The results obtained from existing contemporary methods and the proposed method are simulated on various standard gray scale images like truck, Einstein, aircraft, cameraman, airport, Elaine, bottle, circuit, F16, girl, peppers, pirate, putrajaya, village, house, Jet, and so forth and are compared

with the enhancement quality of the proposed algorithm. 80 test images are used to evaluate the performance and are subjected to PSO based MMOHM process with 40 iterations. The proposed system has been implemented and evaluated on MATLAB. Figure 2 shows PSO based minimum entropy search (convergence graph) for truck and Einstein images. The graphs clearly show the successful implementation of optimal search mechanism. The optimal value of enhancement parameters are obtained in 9th iteration for truck image, since the objective function (difference between DE value of original and enhanced image) reaches minimal value (i.e., zero). The corresponding optimal values for φ_1 and φ_2 are substituted in (7) and (10) to get the modified histogram for lower and upper subimage. Finally, merge the two subhistograms and display the contrast enhanced details preserved output image. Similarly, the optimal value of enhancement parameters are obtained in 11th iteration for Einstein image since the objective function reaches minimal value in 11th iteration. The convergence graph clearly shows the difference between DE values of original and enhanced image for each iteration.

To compare the performance of the proposed method, the same images are enhanced by using the existing contrast enhancement techniques such as GHE, BBHE, DSIHE, WTHe, RMSHE (with recursion level = 2), and RLBHE. In all of these methods, the performance is measured qualitatively by using human visual perception and quantitatively by using three widely used metrics such as DE, AMBE, and NIQE index. The qualitative performance of MMOHM and the contemporary methods are illustrated using Einstein and truck image, which are given in Figures 3 and 5. The original images are given in Figures 3(a) and 5(a). The enhanced images using contemporary enhancement techniques are also shown from Figures 3(b) to 3(g) and Figures 5(b) to 5(g). The encircled portions (Face) in these images clearly

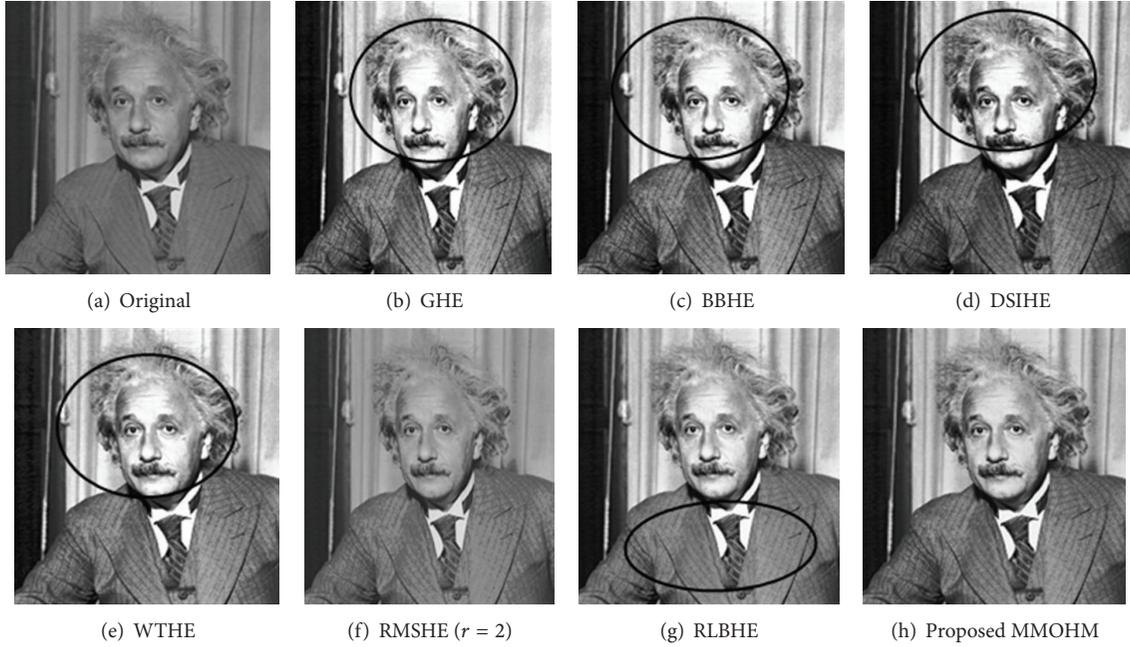


FIGURE 3: (a) Einstein Image; results by various HE techniques (b)–(g), and (h) the proposed MMOHM.

show brightness degradation and over enhancement. HEed image in Figure 3(b) shows that the average brightness is increased instead of increasing the contrast. Though there is not much change in brightness in BBHE, DSIHE, and WTHe, these images are almost similar to each other (Figures 3(c)–3(e)) and the marked portions of them are found to be overenhanced. The same abrupt brightness change and overenhancement can also be noticed in truck image also (Figures 5(c)–5(e)). Figure 3(f) is the results of RMSHE in which the output images are almost similar to original image even for the recursion level = 2. The visual results of RLBHE are shown in Figures 3(g) and 5(g). This method exhibits better results when compared with other contemporary methods. But, this method does not guarantee that it preserves the original image details in the enhanced image. The contrast of the enhanced image is found to increase progressively. Also, the subsequent increase in the contrast has not introduced any unwanted artifacts in the enhanced image. The encircled portions in Figures 3(g) and 5(g) clearly show that RLBHE fails to preserve the original details. Figures 3(h) and 5(h) are the output of the proposed MMOHM. It can be clearly noted that MMOHM exhibits better results in terms of visual perception when compared to the contemporary methods. The contrast of the enhanced image is found to increase progressively noting that the subsequent increase in the contrast has not introduced any unwanted artifacts in the enhanced image.

The histogram patterns of original Einstein image, proposed, and existing contemporary methods are shown in Figure 4. In Figure 4(b), the abrupt change in brightness can well be notified due to uncontrolled distribution of histograms. The encircled portions in Figures 4(c)–4(e)

shows overenhancement. Figure 4(f) looks almost similar to original histogram as in Figure 4(a). The marked portion in the histogram of RLBHE (Figure 4(g)) shows that this method fails to preserve the original details whereas Figure 4(h) exhibits the controlled distribution, which results in the expected contrast enhancement.

The proposed MMOHM method can be extended for color images also. The original fireworks and light house images are shown in Figures 6(a) and 6(b). The resulting images obtained by the various existing methods and proposed method are also given from Figures 6(c) to 6(f). Figures 6(c) and 6(d) show that GHE provides a significant improvement in image contrast. However, it also amplifies the noise level of the images along with some artifacts and undesirable side effects such as washed-out appearance. The enhancement result of RLBHE is visually unpleasing (Figures 6(e) and 6(f)) and some portions are found to be overenhanced. The results of GHE and RLBHE show that they do not prevent the washed-out appearance in overall image due to significant change in brightness. The visual appearance result shows that the proposed method preserves the naturalness of image and also prevents the side effect due to the significant change in brightness effectively (Figures 6(g) and 6(h)). By visually inspecting the images on these figures, it can clearly be noted that only the proposed method is able to generate natural looking image and still offer contrast enhancement. Hence, MMOHM works well for the color images too.

4.1. Experimental Analysis. Initially, the input image gray-level ranging from (X_0, X_{L-1}) is partitioned into 3 equal subsections. These subsections are ranging between (X_0, \dots, X_{m_1}) , $(X_{m_1} + 1, \dots, X_{m_2})$, and $(X_{m_2} + 1, \dots, X_{L-1})$.

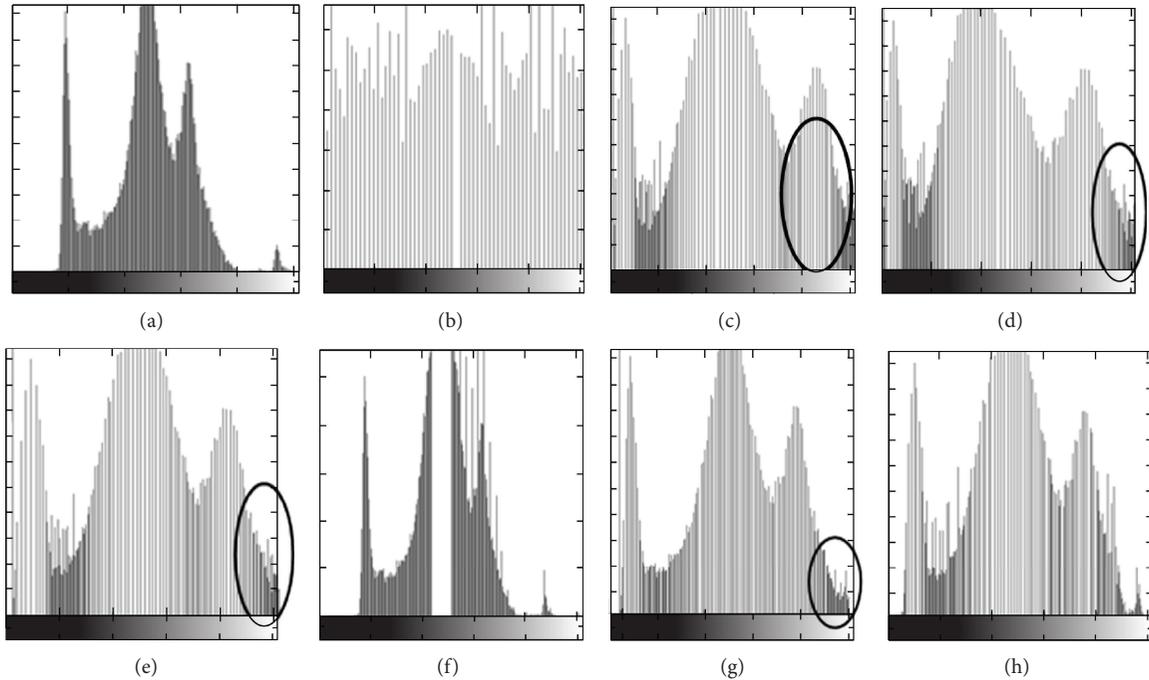


FIGURE 4: Histogram patterns of Einstein image; (a) original; results of (b) GHE, (c) BBHE, (d) DSIHE, (e) WTHE, (f) RMSHE ($r = 2$), (g) RLBHE, and (h) MMOHM.

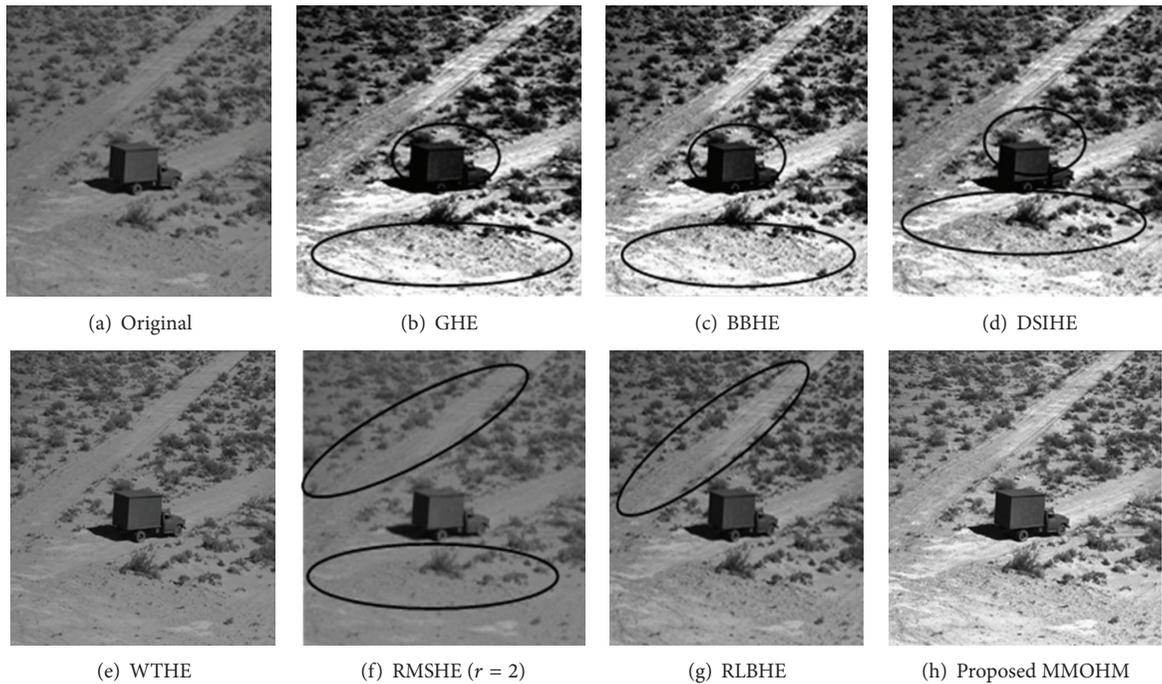


FIGURE 5: (a) Truck Image; results by various HE techniques (b)–(g) and (h) the proposed MMOHM.

Then, peak value is found for each subsection. Finally, $mmean$ of these peak values is computed. The same procedure is to be followed for partitioning the gray-level range into 10 equal subsections and 20 equal subsections of entire gray-level. The decision is to use any of these three cases (3 subsections,

10 subsections, or 20 subsections) which depend on the nature of the input image. This procedure is experimented on three different images, namely, aircraft, u2, and circuit. The aircraft image and the corresponding histogram are shown in Figures 7(a) and 7(b). This image is dominated



FIGURE 6: (a) Fireworks image and (b) light house image. Results by various HE techniques (b)–(f), (g), and (h) the proposed MMOHM.

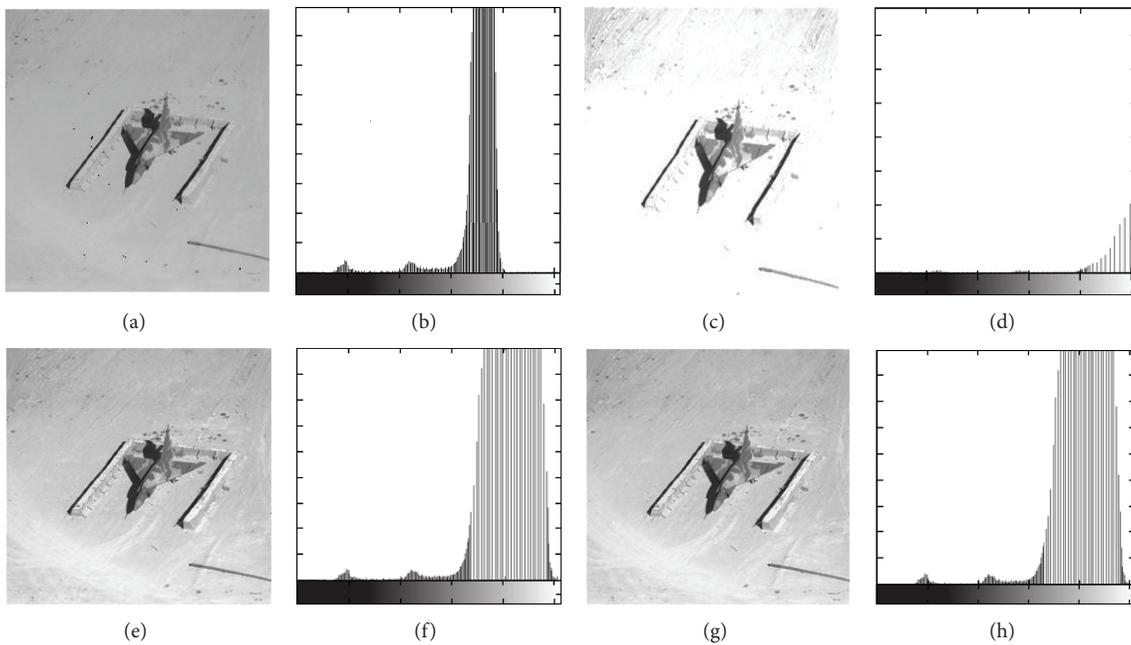


FIGURE 7: (a)–(b) Original aircraft image and the histogram; (c)–(h) MMOHM enhanced images and their histograms based on the peak values are identified from 3, 10, and 20 subsections.

by high frequency components (light image); MMOHM with 3 subsections cannot give better image quality. But, MMOHM with 10 subsections can produce better contrast enhancement. Similarly, MMOHM with 20 subsections can produce slightly better contrast enhancement than MMOHM with 10 subsections. The enhanced images using MMOHM with 3 subsections, 10 subsections, and 20 subsections and their corresponding histograms are shown from Figures 7(c) to 7(h). From the enhanced images and their histograms, it is observed that whenever the number of

subsections increased, we get a more enhanced image in the event of the input image being a light image.

The u2 image and the corresponding histogram are shown in Figures 8(a) and 8(b). Since the image is dominated by low frequency components (darkened image), MMOHM with 3 subsections can give better image quality than the original image. But, MMOHM with 10 subsections can still produce better enhancement than MMHE with 3 subsections. Similarly, MMOHM with 20 subsections can also produce slightly better contrast enhancement than MMOHM with

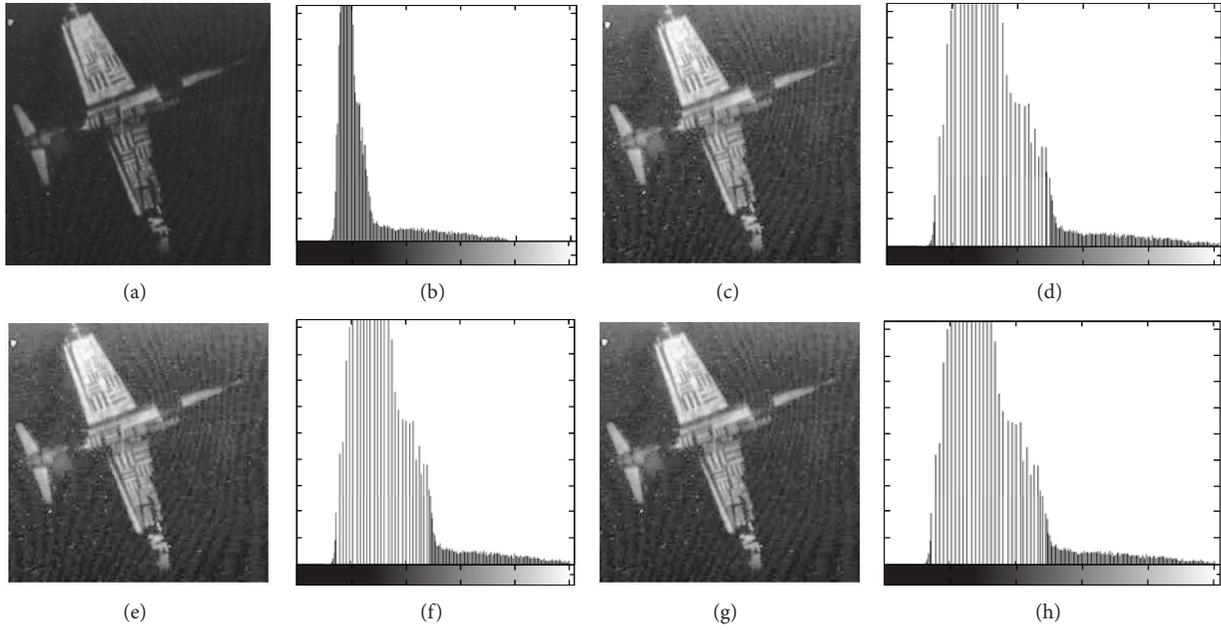


FIGURE 8: (a)-(b) Original U2 image and the histogram; (c)-(h) MMOHM enhanced images and their histograms based on the peak values are identified from 3, 10, and 20 subsections.

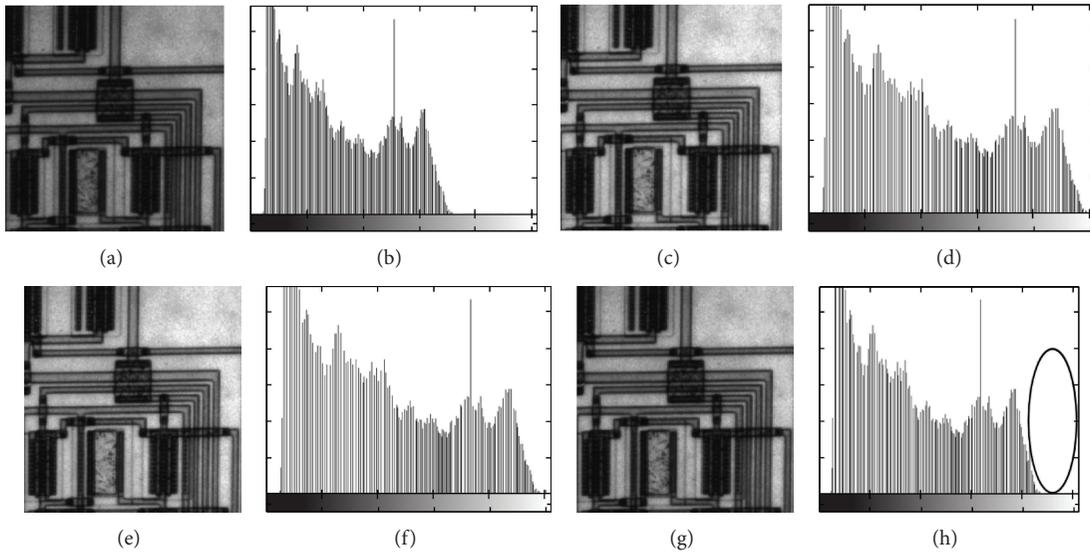


FIGURE 9: (a)-(b) Original circuit image and the histogram; (c)-(h) MMOHM enhanced images and their histograms based on the peak values are identified from 3, 10, and 20 subsections.

10 subsections. The enhanced images using MMOHM with 3 subsections, 10 subsections, and 20 subsections and their corresponding histograms are shown from Figures 8(c) to 8(h). From the enhanced images and their histograms, it is observed that whenever the number of subsections increased, we get a more enhanced image provided the input image is a dark one.

The circuit image and the corresponding histogram are shown in Figures 9(a) and 9(b). Since the images of the histograms are evenly distributed, MMOHM with 3 subsections itself can give better image contrast enhancement. MMOHM

with 10 subsections can also produce more or less similar enhancement like MMOHM with 3 subsections. But, MMOHM with 20 subsections tends to decrease the quality slightly. If the number of subsections is further increased, the enhanced image looks quite similar to the original image. The encircled portion in the circuit image histogram (Figure 9(h)) clearly shows diminution of the gray-level range in the enhanced image. The enhanced images using MMOHM with 3 subsections, 10 subsections, and 20 subsections and their corresponding histograms are shown from Figures 9(c) to 9(h). From the enhanced images and their

TABLE 1: Comparison of DE values.

IMAGE	Original	HE	BBHE	DSIHE	WTHE	RMSHE	RLBHE	BBC	LDR	AGCWD	MMOHM
Truck	6.5461	5.8041	6.4362	6.4251	6.4693	6.3312	6.4948	6.5121	6.6223	6.5356	6.547
Einstein	6.8841	5.9436	6.7379	6.7251	6.7695	6.7208	6.8479	6.8788	6.9211	6.8789	6.882
Aircraft	5.4316	5.0356	5.2968	5.2766	5.286	5.0716	5.3822	5.3912	5.2294	5.2478	5.2561
Cameraman	7.0097	5.9106	6.8081	6.7792	6.7614	6.9259	6.9117	6.9211	6.9412	6.9478	6.9533
Airport	6.7337	5.8853	6.6365	6.624	6.6286	6.6217	6.3464	6.7141	6.6926	6.6923	6.7581
Elaine	6.5734	5.9397	6.4607	6.4591	6.5513	6.2439	6.4852	6.5421	6.5455	6.5547	6.5608
Bottle	7.4472	5.9776	7.2744	7.2818	7.2463	7.3624	7.2614	7.3214	7.3763	7.4022	7.4021
Circuit	6.9439	5.9354	6.9194	6.9316	6.926	6.8395	6.6416	6.8786	6.9211	6.9023	6.9311
F16	6.6744	5.7103	6.5877	6.5601	6.4533	6.4677	6.2236	6.4326	6.3678	6.4281	6.4545
Girl	7.2421	5.9594	7.0295	7.0352	7.051	7.114	6.2508	7.3428	7.1123	7.2322	7.2175
Peppers	7.641	5.9815	7.473	7.4665	7.475	7.5017	7.5913	7.5313	7.5345	7.6101	7.6211
Pirate	7.3118	5.9842	7.2276	7.2276	7.2473	7.1627	6.9915	6.9821	6.9996	7.2114	7.262
Putrajaya	6.8116	5.7758	6.6754	6.6497	6.6594	6.6347	6.7857	6.9845	6.9347	6.8546	6.6565
Village	7.4505	5.9769	7.2994	7.2911	7.3165	7.3001	7.3393	7.3992	7.4091	7.4193	7.4272
House	6.9166	5.8	6.6808	6.6824	6.7963	6.8534	6.5985	6.5623	6.6825	6.7235	6.7991
Jet	6.552	5.8116	6.3681	6.3296	6.3299	6.3865	6.4938	6.5433	6.5537	6.4692	6.4664
Average	6.8856	5.8395	6.7445	6.734	6.7479	6.7211	6.6654	6.8086	6.8027	6.8194	6.8247

histograms, it is observed that whenever the number of sub-sections increased, the enhanced image tends to be original, if the image histogram is evenly distributed.

From the above observations, it is understood that the input image is dominated by high frequency or low frequency components as given in Figures 7(a) and 8(a). One should identify the peaks from large number of subsections and *mmean* value is found based on the peaks identified. This will improve the contrast effectively. When the input image has its histogram components evenly distributed like circuit image (Figure 9(a)), it is better to identify the peaks from smaller number of subsections. The *mmean* value is computed and it is enough to obtain better contrast enhancement.

Further, the qualities of 80 test images which are enhanced using the above mentioned techniques are measured in terms of DE, NIQE, and AMBE. The DE values obtained from some of the test images are given in Table 1. Table 1 enables us to understand that the proposed method proves to produce better entropy values which are closer to the original image. This signifies that the proposed method is found to preserve the details of the original image in the enhanced image also. In Table 2, the NIQE values of the test images for various HE methods along with the proposed method are given. For most of the test images, NIQE is found to have lower values than other methods for corresponding images. This authenticates that the proposed method is found to increase the quality of the output image. Table 3 shows average AMBE measure for all methods obtained from 80 test images. It shows that, the proposed method can preserve the mean brightness better than the existing HE based enhancement techniques since the average AMBE value for all these enhanced test images as 17.6737 and the average

AMBE for original test images as 34.1451. A median AMBE value indicates better brightness preservation.

Most of the HE based enhancement techniques, invariably, attribute to mean shift in output image. This is due to redistribution of intensity values during intensity normalization. However, the proposed technique preserves the mean of the input image. Figure 10(a) shows the comparison of the average DE values of the original, HE, BBHE, RMSHE, WTHE, RLBHE, BBC, LDR, and AGCWD methods along with the proposed method. It is observed that HE drastically changes the original mean of the input image which results in degradation of brightness. But the controlled procedure adopted by MMOHM gives average DE values which are closer to that of original ones. This shows that MMOHM preserves the details of the original image in the enhanced image. The same observation is endorsed from Figure 10(b) by the average NIQE values which make it evident that the MMOHM improves the contrast better than the rest. Figure 10(c) shows the comparison of the average AMBE values of the HE, BBHE, RMSHE, WTHE, RLBHE, BBC, LDR, and AGCWD methods along with the proposed method. From the above results, it is inferred that the proposed method MMOHM produces and preserves brightness along with controlled contrast enhanced output image.

The computation of algorithm was implemented using the MATLAB version 6.0. In a personal computer with 3 GHz Processor, most HE based methods required about less than one minute to process one full HD 256×256 pixels image. The discussion of time complexity provides only rough comparison as the procedure depends upon many factors such as the choice of the hardware platform and the level of the software optimization.

TABLE 2: Comparison of NIQE values.

IMAGE	Original	HE	BBHE	DSIHE	WTHE	RMSHE	RLBHE	BBC	LDR	AGCWD	MMOHM
Truck	7.9686	10.5953	9.9531	10.6911	9.2633	9.0152	7.4289	7.9321	7.8823	7.9125	7.5093
Einstein	7.2088	8.9804	8.8564	8.9838	8.6024	8.0784	7.2105	7.3675	7.367	7.2456	7.1709
Aircraft	10.4572	14.5758	15.1879	15.0399	13.8421	9.1834	8.4665	9.9765	8.5632	10.1902	10.9741
Cameraman	7.4073	9.6771	9.1239	9.0664	8.6863	7.5214	8.4614	7.1021	8.5679	7.7823	7.9686
Airport	8.6812	9.1694	9.5014	9.2956	9.1243	9.8319	8.2903	8.2909	8.5634	8.4569	8.2654
Elaine	6.4449	5.5701	5.929	6.0677	5.7728	7.9715	5.6807	5.9026	5.8992	5.9099	5.8519
Bottle	7.6794	8.156	8.6762	8.5837	8.1654	7.4663	7.6944	7.4189	7.4562	7.6721	7.1281
Circuit	8.591	8.6751	7.5192	8.313	7.9532	9.6849	8.4122	8.6912	8.6578	8.5637	8.3758
F16	8.2282	8.0299	8.0583	8.9568	7.6373	8.5381	7.8178	7.6117	7.9971	7.8924	7.8531
Girl	5.6609	7.1748	6.9261	7.1086	6.8583	5.9396	7.9211	6.9927	6.9725	5.9854	5.7459
Peppers	6.7165	6.7126	6.7949	6.377	6.3459	6.83	6.6507	6.4568	6.392	6.5537	6.3625
Pirate	6.717	6.8432	6.6451	6.6785	6.2363	7.1687	8.6744	8.7671	8.7327	6.8957	6.6795
Putrajaya	6.4962	6.832	5.669	5.9769	5.7253	6.6505	6.8472	5.8623	5.2341	5.7831	5.768
Village	6.4567	8.6245	7.8077	7.6354	7.4104	8.1163	6.6283	6.6508	6.5602	6.921	6.4429
House	9.2706	13.541	13.2168	13.0355	12.6046	9.5735	8.7654	8.9023	8.2395	9.2341	9.3659
Jet	10.6356	11.157	10.6305	10.8457	10.7	9.918	9.3526	9.5826	9.6672	9.4532	10.5586
Average	7.7888	9.0196	8.781	8.916	8.433	8.218	7.7689	7.7193	7.672	7.6532	7.6263

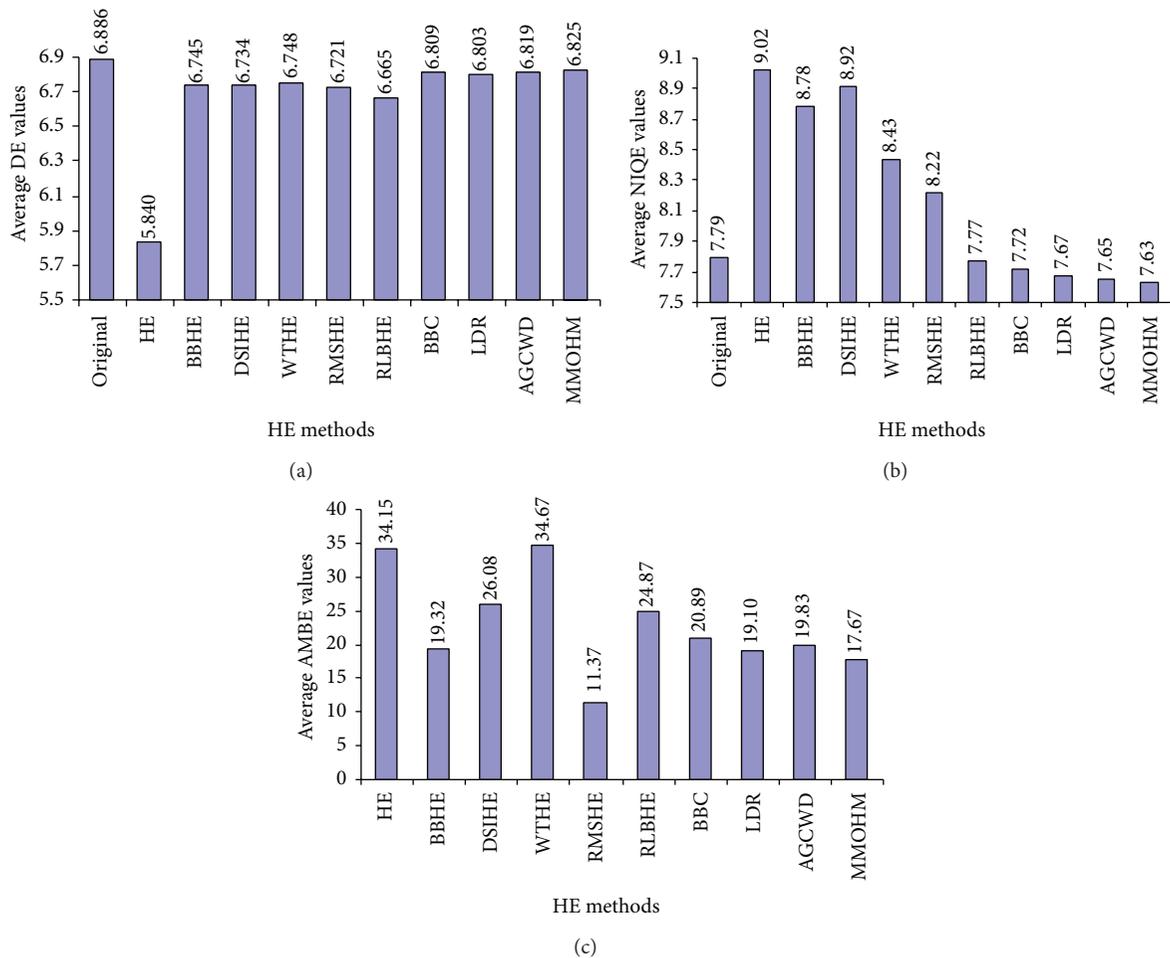


FIGURE 10: Comparison of (a) average DE values, (b) average NIQE values, and (c) average AMBE values.

TABLE 3: Average AMBE values for 80 test images.

Methods	HE	BBHE	DSIHE	WTHE	RMSHE	RLBHE	BBC	LDR	AGCWD	MMOHM
AMBE values	34.14506	19.31967	26.07905	34.66607	11.36573	24.87203	20.8934	19.1032	19.8263	17.67375

5. Conclusion

This paper uses multipoint mean value to partition the input image into two subimages. Then the particle swarm optimization is used to find optimal contrast enhancement parameters to modify the subhistograms. Finally, the union of the two modified subhistograms produces a contrast enhanced and details preserved output image. It is apparent from the experiments that the proposed MMOHM method creates visually pleasant enhancement effects. This method also eliminates the problems of over enhancement and white saturation which may cause degradation in brightness in the input images. Also, MMOHM accomplishes two major desired objectives of brightness preservation and contrast enhancement for any colored or gray scale images without eliminating the original characteristics of the input image. Moreover, it is not recursive in nature like RMSHE, RSWHE, or RSIHE. This is substantially proved with well known metrics such as DE, NIQE, and AMBE. Hence, this method is more suitable to be employed in consumer electronics for enhancing the contrast of the image without compromising on its visual quality.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] S.-C. Huang, F.-C. Cheng, and Y.-S. Chiu, "Efficient contrast enhancement using adaptive gamma correction with weighting distribution," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 1032–1041, 2013.
- [2] Y. Yang, J. Zhang, and X. Huang, "Adaptive image enhancement algorithm combining kernel regression and local homogeneity," *Mathematical Problems in Engineering*, vol. 2010, Article ID 693532, 14 pages, 2010.
- [3] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice Hall, New York, NY, USA, 2nd edition, 2002.
- [4] Q. Wang and R. K. Ward, "Fast image/video contrast enhancement based on weighted thresholded histogram equalization," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 757–764, 2007.
- [5] T. Celik and T. Tjahjadi, "Automatic image equalization and contrast enhancement using Gaussian mixture modeling," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 145–156, 2012.
- [6] Y.-T. Kim, "Contrast enhancement using brightness preserving bi-histogram equalization," *IEEE Transactions on Consumer Electronics*, vol. 43, no. 1, pp. 1–8, 1997.
- [7] Y. Wang, Q. Chen, and B. Zhang, "Image enhancement based on equal area dualistic sub-image histogram equalization method," *IEEE Transactions on Consumer Electronics*, vol. 45, no. 1, pp. 68–75, 1999.
- [8] S.-D. Chen and A. R. Ramli, "Minimum mean brightness error bi-histogram equalization in contrast enhancement," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 4, pp. 1310–1319, 2003.
- [9] S.-D. Chen and A. R. Ramli, "Contrast enhancement using recursive mean-separate histogram equalization for scalable brightness preservation," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 4, pp. 1301–1309, 2003.
- [10] K. S. Sim, C. P. Tso, and Y. Y. Tan, "Recursive sub-image histogram equalization applied to gray scale images," *Pattern Recognition Letters*, vol. 28, no. 10, pp. 1209–1221, 2007.
- [11] S.-D. Chen and A. R. Ramli, "Preserving brightness in histogram equalization based contrast enhancement techniques," *Digital Signal Processing: A Review Journal*, vol. 14, no. 5, pp. 413–428, 2004.
- [12] M. Kim and M. G. Chung, "Recursively separated and weighted histogram equalization for brightness preservation and contrast enhancement," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 3, pp. 1389–1397, 2008.
- [13] N. Sengee and H. K. Choi, "Brightness preserving weight clustering histogram equalization," *IEEE Transactions on Consumer Electronics*, vol. 54, no. 3, pp. 1329–1337, 2008.
- [14] H. Ibrahim and N. S. Pik Kong, "Image sharpening using sub-regions histogram equalization," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 2, pp. 891–895, 2009.
- [15] C. Zuo, Q. Chen, and X. Sui, "Range Limited Bi-Histogram Equalization for image contrast enhancement," *Optik*, vol. 124, no. 5, pp. 425–431, 2013.
- [16] M. Sundaram, K. Ramar, N. Arumugam, and G. Prabin, "Histogram modified local contrast enhancement for mammogram images," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5809–5816, 2011.
- [17] L. Leng, M. Li, and J. S. Zhang, "Histogram equalization algorithm with local adaptive enhancement based on edge details," *Microelectronics and Computer*, vol. 27, pp. 38–41, 2010.
- [18] P. Shanmugavadivu and K. Balasubramanian, "Thresholded and optimized histogram equalization for contrast enhancement of images," *International Journal of Computers and Electrical Engineering*, vol. 40, no. 3, pp. 757–768, 2014.
- [19] C. Lee and C.-S. Kim, "Contrast enhancement based on layered difference representation of 2D histograms," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5372–5384, 2013.
- [20] F.-C. Cheng and S.-C. Huang, "Efficient histogram modification using bilateral bezier curve for the contrast enhancement," *Journal of Display Technology*, vol. 9, no. 1, pp. 44–50, 2013.
- [21] O. Ghita, D. E. Ilea, and P. F. Whelan, "Texture enhanced histogram equalization using TV-L¹ image decomposition," *IEEE Transactions on Image Processing*, vol. 22, no. 8, pp. 3133–3144, 2013.
- [22] C. Wang and Z. Ye, "Brightness preserving histogram equalization with maximum entropy: a variational perspective," *IEEE Transactions on Consumer Electronics*, vol. 51, no. 4, pp. 1326–1334, 2005.
- [23] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a 'completely blind' image quality analyzer," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 209–212, 2013.

Research Article

Optimal Sizing of a Photovoltaic-Hydrogen Power System for HALE Aircraft by means of Particle Swarm Optimization

Victor M. Sanchez,¹ Romeli Barbosa,¹ J. C. Cruz,² F. Chan,¹ and J. Hernandez¹

¹Universidad de Quintana Roo, Boulevard Bahía s/n, 77019 Chetumal, QROO, Mexico

²Instituto Tecnológico de Chetumal, Avenida Insurgentes No. 330, 77013 Chetumal, QROO, Mexico

Correspondence should be addressed to Victor M. Sanchez; vicsan.huerta@gmail.com

Received 6 June 2014; Accepted 4 September 2014

Academic Editor: Baozhen Yao

Copyright © 2015 Victor M. Sanchez et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Over the last decade there has been a growing interest in the research of feasibility to use high altitude long endurance (HALE) aircrafts in order to provide mobile communications. The use of HALEs for telecommunication networks has the potential to deliver a wide range of communication services (from high-quality voice to high-definition videos, as well as high-data-rate wireless channels) cost effectively. One of the main challenges of this technology is to design its power supply system, which must provide the enough energy for long time flights in a reliable way. In this paper a photovoltaic/hydrogen system is proposed as power system for a HALE aircraft due its high power density characteristic. In order to obtain the optimal sizing for photovoltaic/hydrogen system a particle swarm optimizer (PSO) is used. As a case study, theoretical design of the photovoltaic/hydrogen power system for three different HALE aircrafts located at 18° latitude is presented. At this latitude, the range of solar radiation intensity was from 310 to 450 Wh/sq-m/day. The results obtained show that the photovoltaic/hydrogen systems calculated by PSO can operate during one year with efficacies ranging between 45.82% and 47.81%. The obtained sizing result ensures that the photovoltaic/hydrogen system supplies adequate energy for HALE aircrafts.

1. Introduction

Communication and energy technologies play an important role in the economic and social development of any nation. Furthermore, due to the increase in world population, bandwidth and energy consumption is growing. As an alternative to increasing the effectiveness of future communications, it has been proposed the use of high altitude long endurance (HALE) aircrafts, as liberators of bandwidth and enhancers of wireless communication [1–4].

Compared with satellite technology, HALE aircrafts, also known as high altitude platforms (HAPs) and high altitude aircraft and airships (HASS) have a cost of launching and operating smaller, higher capacity data transmission and increased spectral efficiency. Besides, they are considered as substitutes for low earth orbit (LEO) satellites. HALE aircrafts fly in the stratosphere, providing relay services for wireless communication networks with a single coverage area about 100 km in diameter [5]. In addition, there are

a variety of specific applications regarding communications, monitoring large areas of interest, scientific applications, or other missions requiring high resolution images or data almost immediately.

HALE aircrafts are now being actively developed in a number of programmes all around the world [6]. Some relevant projects are Heliplat-HeliNet, SkyTower, SkyStation, SkyLARK, StratoSat, Pathfinder, Pathfinder plus, Centurion, Helios series, and the Zephyr series just to name a few [6, 7]. HALE aircrafts is an incipient technology yet; however the surge of recent activity reflects both the lucrative demand for wireless services as well as advances in solar cells and energy storage systems. In order to ensure project sustainability, including cost-effective benefits, the flight duration for a HALE aircraft must be continuous up to 6 months. This target only can reach it with a regenerative power system, for example, the power system studied in this paper.

In this way, the power subsystem is a key part that determines the implementation and feasibility of them. The use of

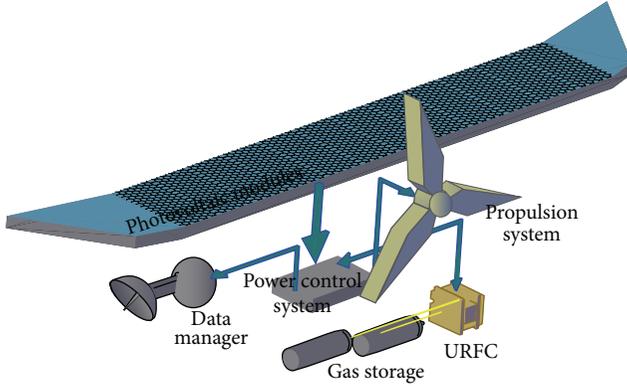


FIGURE 1: Energy flow diagram for HALE aircraft studied.

photovoltaic systems improves significantly the profitability and environmental efficiency of HALE [8]. However, due to its intermittency, solar energy requires a safe and reliable power storage system. One option to solve this problem is to use hydrogen as storage system of renewable energy. Hydrogen storage system is a viable option for many researchers [9], especially for mobile applications. Hydrogen is a suitable energy storage medium that is free of carbon and other impurities; it is also the most abundant element in the universe [10]. A system of production and energy storage based on hydrogen technology coupled with solar energy could provide greater efficacy to HALE aircraft because the closed loop system ensures long endurance and lightweight; furthermore the cost of hydrogen production and use has decreased in recent years.

Proton exchange membrane (PEM) technology (both fuel cells as electrolyzer) is by excellence the technology for an efficient consumption and generation of hydrogen [11]. On the other hand, unitized regenerative fuel cells (URFCs) are an excellent option in situations where weight and volume are a constraint. URFC is a compact electrochemical system integrated both by a fuel cell and a water electrolyzer. HALE aircraft powered by a subsystem based on solar cells and hydrogen (PVS-H₂) should include a comprehensive storage system. Figure 1 shows a general diagram of a PVS-H₂.

Power subsystem should be designed to satisfy two important requirements: (1) the propulsion and positioning of the aircraft and (2) the reception, handling, and transmission of information. The primary system produces electricity as long as the sun's radiation is present (PVS, composed of an array of photovoltaic panels). A control system (ACS) performs the management and conditioning of power generated according to the characteristics of the electrical load demand. The PVS must be designed in order to generate more energy during the day than the load consumes (energy surplus condition); the excess energy is used to produce hydrogen using the URFC (electrolyzer mode). This gas is stored in a storage media, for example, materials-based hydrogen storage system (MHSS). It is important to note that the development of new hydrogen storage technology is a current challenge for the global scientific community [12, 13]. At night, electricity is generated by the URFC (fuel

cell mode). On the other hand, the solar radiation reaching the earth's surface is reduced because a large part of it is scattered, reflected, or absorbed by the atmosphere. However, at the stratosphere, the solar radiation is characterized by high power and low intermittence [14]. Despite that, primary power system is not immune to low insolation level, gas leaks, or an URFC failure that cause hydrogen depletion, so that an emergency backup subsystem must be considered in order to provide the power for the safe landing of the aircraft.

The basic requirements for the sizing of the PVS-H₂ system are to achieve maximum effectiveness as well as to ensure a reliable power supply. Power system sizing has to consider the steady-state characteristics and the profiles transient state of energy sources by location as well as the electrical load demand [13]. Regardless of the fact that, to ensure the survival of the aircraft, power system requires minimum error uncertainty to ensure at least the energy needed to control the position and rotation of the aircraft. On the other hand, due to the relationship between power required for flight and total weight, the optimal sizing of the PVS-H₂ energy system is essential to find one or more effective solutions that meet the operational goals.

This paper proposes the optimal sizing of an energy generating system for a HALE aircraft by means of PSO (particle swarm optimization). PSO is a population based stochastic optimization technique developed by Eberhart and Kennedy in 1995, which was inspired by the social behavior of animals [15]. Originally, PSO was intended to handle nonlinear continuous optimization problems. However due to its simple concepts, fast convergence speed, and easy implementation, PSO has been widely applied to solve intricate optimization problems of real-world engineering fields as in power systems issues [16–20]. The aim of this paper is to determine the optimal configuration of the power system that allows the flight of the HALE aircraft along one year with the best efficacy.

2. Mathematical Formulation

2.1. System's Global Efficacy. The design strategy consists of making an energy balance to evaluate the power system behavior. The system's efficiency (η_s) is calculated according to the first law of thermodynamics:

$$\eta_s = \frac{(\varepsilon_{out} - \varepsilon_s)}{\varepsilon_{in}}, \quad (1)$$

where ε_{out} is the electrical energy consumed by the HALE aircraft, ε_{in} is the electrical energy output of the PVS, and ε_s is the electrical energy stored in the MHSS. ε_s is negative when the hydrogen is produced for the URFC at electrolyzer mode and it is positive when the hydrogen is consumed by the URFC at fuel cell mode. In this work, η_s is analyzed in hourly intervals; the average system efficiency ($\eta_{s,mean}$) is calculated by

$$\eta_{s,mean} = \sum \frac{\eta_s}{n}, \quad (2)$$

where n is hours of the flight. The functionality of the system is defined by the ratio of the actual time of flight ($T_{f,actual}$)

to maximum time of flight ($T_{f,max}$). Hence, system's global efficacy (Eff_S) is determined by the following equation:

$$Eff_S = \left[\frac{T_{f,actual}}{T_{f,max}} \right] \eta_{S,mean}, \quad (3)$$

where $T_{f,max}$ is a constant that, in this work, is equal to 8640 (hours in a year). In this way, Eff_S is in function of ϵ_{in} , ϵ_{out} , and ϵ_s .

2.2. Energy Input. The energy input (ϵ_{in}) of the power subsystem is the energy output of the PVS. This energy is in function of the solar resource available on the site and the physical characteristics of the device, as is described by

$$P_{in} = \eta_{PVS} A_{PVS} G_i, \quad (4)$$

where η_{PVS} is the instantaneous PVS efficiency, A_{PVS} is the total area of the PVS, and G_i is the radiation on the PVS surface. Strictly, η_{PVS} is dependent on three parameters: the temperature, the packing factor, and the module reference efficiency. However, the efficiency used in this work is a global parameter of a hypothetical PVS ($\eta_{PVS} = 16\%$) [7]. G_i depends on the time and latitude of the place; it can be calculated by [14]

$$G_i = G_{SC} \left[1 + 0.033 \cos \left(\frac{360N}{365} \right) \right] \times [\sin(L) \sin(\delta) + \cos(L) \cos(\delta) \cos(h)], \quad (5)$$

where G_i is the extraterrestrial radiation measured on the plane normal to the radiation on the N th day of the year, G_{SC} is the solar constant ($G_{SC} = 1366.1 \text{ W/m}^2$ ASTM E-490), N is the day of the year, L is the latitude of the place, and h is the solar hour. The solar declination (δ), in degrees for any day of the year (N), is calculated approximately by [14]

$$\delta = 23.45 \sin \left[\frac{360}{365} (284 + N) \right]. \quad (6)$$

It is relevant to note that G_i is strongly dependent on the latitude.

One hour time step is employed as base time in the energy balance for the aircraft power system, so that equivalence between power and energy is used as is specified in

$$\begin{aligned} \epsilon_{in} &= P_{in}, \\ \epsilon_{out} &= P_{out}. \end{aligned} \quad (7)$$

2.3. Hydrogen Energy Storage. Figure 2 shows a schematic diagram of the energy transport in the hydrogen storage system.

The hydrogen generated and consumed is in function of the surplus and deficit energy. According to the first law of thermodynamics, without electrochemistry considerations, the FC efficiency (η_{FC}) and the electrolyzer efficiency (η_{PE}) can be defined as follows:

$$\begin{aligned} \eta_{PE} &= \frac{\epsilon_{NHg}}{\epsilon_{surplus}}, \\ \eta_{FC} &= \frac{\epsilon_{deficit}}{\epsilon_{NHc}}. \end{aligned} \quad (8)$$

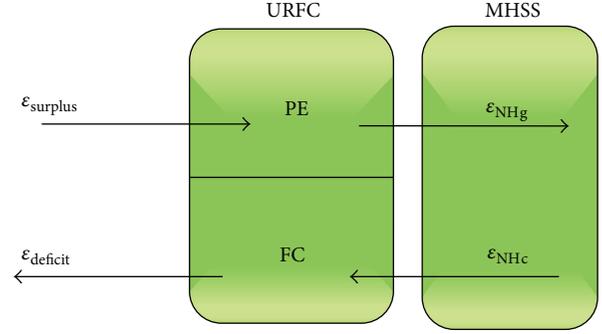


FIGURE 2: Energy transformation diagram of the hydrogen technology system.

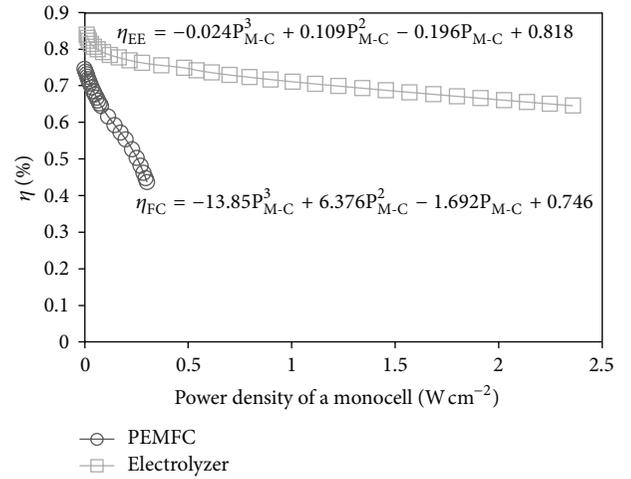


FIGURE 3: Performance and extrapolation equation of the experimental PEMFC and electrolyzer efficiency values with respect to the power density of a monocrystalline cell.

where ϵ_{NHg} is the electrical energy of hydrogen generated by electrolyzer (PE), $\epsilon_{surplus}$ is the energy surplus, $\epsilon_{deficit}$ is the energy deficit, and ϵ_{NHc} is the electrical energy of hydrogen consumed by fuel cell (FC). On the other hand, ϵ_s can be determined by the energy balance of

$$\epsilon_s = \epsilon_{NHg} - \epsilon_{NHc}. \quad (9)$$

By substitution of (8) into (9), a mathematical relation that determines ϵ_s as a function of the URFC efficiency, as well as of the surplus and deficit energies, is obtained in

$$\epsilon_s = \eta_{PE} \epsilon_{surplus} - \frac{\epsilon_{deficit}}{\eta_{FC}}. \quad (10)$$

In this work, the real efficiency of a monocrystalline cell is extrapolated in order to find both efficiencies (η_{PE} and η_{FC}). Figure 3 shows the performance and extrapolation equation of PEMFC and electrolyzer used in this work, which were obtained by experimental polarization curves carried out in our laboratory.

2.4. Energy Output. The electric energy output (ϵ_{out}) of the HALE aircraft is a function of the power consumed by the

propeller (P_{TL}). P_{TL} can be described by a mathematical model based on point mass dynamics. In steady flight and in still air, the lift (L) and the drag (D) forces are defined as follows [21]:

$$\begin{aligned} L &= C_L S_W \left(\frac{\rho}{2} \right) V^2, \\ D &= C_D S_W \left(\frac{\rho}{2} \right) V^2, \end{aligned} \quad (11)$$

where C_L and C_D are the lift and drag coefficients, respectively, ρ is the air density, S_W is the wing area, and V is the relative airspeed. The C_L and C_D heavily depend on the airfoil, the angle of attack, and the Reynolds number. In this work, these coefficients are taken from [7], where C_L , C_D , and ρ are calculated by the ‘‘aerodynamic parameters analysis’’ method. At straight and level flight, according to the third Newton law, the lift force compensates for the weight (W) and the propeller thrust compensates for the drag force; thus the power for level flight (P_{TL}) can be calculated by [21]

$$P_{TL} = C_D \left(\frac{W}{C_L} \right)^{3/2} \left(\frac{2}{\rho S} \right)^{1/2}. \quad (12)$$

Then, ε_{out} can be calculated by the propeller efficiency (η_{PP}):

$$\varepsilon_{out} = \frac{tP_{TL}}{\eta_{PP}}. \quad (13)$$

By substitution of (4), (9), and (13) in (1), we determined the system’s efficiency (η_s) and therefore its global efficacy (Eff_s).

3. Basics on the PSO Algorithm

A PSO algorithm consists of a population continuously updating the knowledge of the given searching space. This population is formed by individuals called particles. Each one represents a possible solution finding the global best position by competition as well as cooperation among themselves after some iteration. Each particle keeps track of its coordinates in the problem space which are associated with the best solution (fitness) it has achieved so far. This value is called pbest. Overall best value is tracked by the global version of the particle swarm optimizer also and its location, obtained so far by any particle in the population. This location is called gbest. Each particle moves in the searching space with a velocity V , which is dynamically updated based on its previous velocity. At each time step, each particle moves toward its pbest and gbest locations. Acceleration is weighted by a random term, with separate random numbers being generated for acceleration toward pbest and gbest locations, as is described in

$$\begin{aligned} V_{i,j}^{\text{iter}+1} &= wV_{i,j}^{\text{iter}} + C_1 * \text{rand}() \\ &* (\text{pbest} - X_i) + C_2 * \text{Rand}() * (\text{gbest} - X_i) \\ &\text{for } i = 1, 2, \dots, \text{NIND}; j = 1, 2, \dots, \text{NVAR}, \end{aligned} \quad (14)$$

TABLE 1: Hale aircrafts specifications taken from [19].

Aircraft	Total mass (kg)	Empty mass (kg)	Wing area (m ²)
Pathfinder	252	207	70.8
Pathfinder +	315	247	871
Helios HP01	719	600	186.6

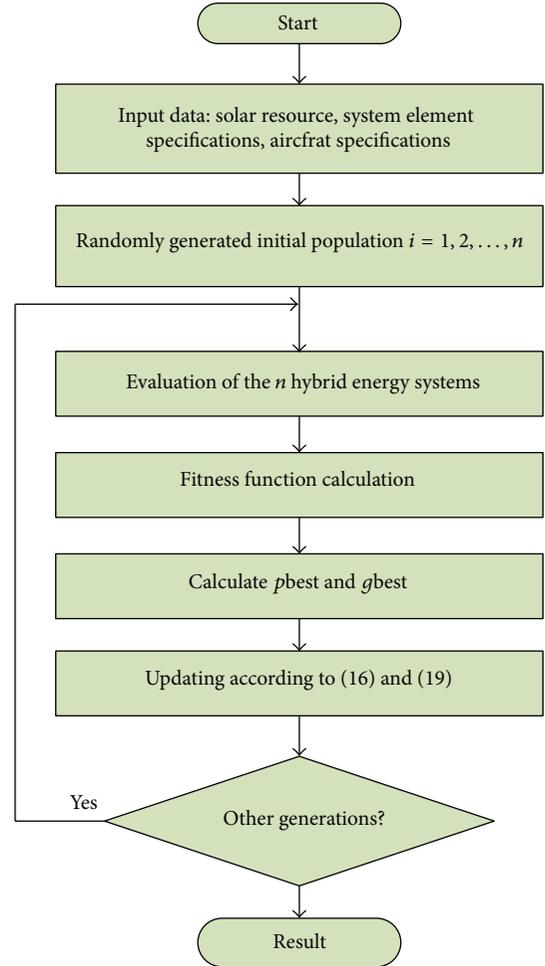


FIGURE 4: Flowchart of the optimization process.

where V is the particle’s velocity; iter is the current iteration; C_1 and C_2 are two positive learning factors; X_i is the i th particle’s actual position; rand() and Rand() are two randomly generated values within the range [0, 1]; NIND is the number of particles; NVAR is the number of variables and w is known as the inertia weight. It plays the role of balancing the global and local searching [22]. In order to avoid a premature convergence problem in PSO, a linearly decreased inertia weight over time is used for this parameter.

The position of particles is updated during each iteration. This is done by adding the velocity value to the particle’s position as follows:

$$X_{i,j}^{\text{iter}+1} = X_{i,j}^{\text{iter}} + V_{i,j}^{\text{iter}+1}. \quad (15)$$

TABLE 2: Input data used for the cases of study.

Parameter	Value	Unit	Description
M_{ctr}	8	kg	Mass of control, energy manage, and communication system
N	$1 < N \leq 360$	day	Days of the year
h	$1 < h \leq 24$	h	Hours of a day
G_{SC}	1366.1	W/m ²	Solar constant (ASTM E-490)
C_D	0.028	—	Drag coefficient (20 km altitude, 29.49 m/s level flight velocity and 155185 Re number) [17]
L/D	28.5	—	Ratio of lift-drag (20 km altitude, 29.49 m/s level flight velocity and 155185 Re number) [17]
η_{PP}	0.6	—	Efficiency of propulsion system
η_{PVS}	0.15	—	Efficiency of photovoltaic system
ω_{PVS}	1	kg/m ²	Mass density of solar cells
ω_{URFC}	1500	W/kg	Specific power density of URFC

This process is repeated until a criterion is met, usually a sufficiently good fitness or a maximum number of iterations.

3.1. PSO Analysis Strategy. The main objective of energy system's sizing is to determine the most efficient configuration

$$F^{obj} = \max \left[\left(\frac{\sum_{i=1}^m T_{f,actual,i}}{T_{f,max}} \right) \frac{\sum_{j=PVS,URFC,MSS,ctrl,structure} (P_{TL} = f(W_j)) + \sum_{k=\min}^n P_{ss,k}}{\sum_{l=1}^n P_{PV,l}} \right]. \quad (16)$$

Hence, the optimal configuration for the energy system must satisfy the flight of the aircraft during one year with the best efficacy. As we can note in (16), for a given photovoltaic power, the overall efficiency decreases if an oversizing of the system's elements occurs due to that this means more weight on the aircraft.

3.2. Operating Constraints. The first constraint is related to the maximum weight that the aircraft can transport and which is denoted by

$$\sum_{j=PVS,URFC,ctrl,structure} W_j \leq W_{aircraft,max}. \quad (17)$$

On the other hand, the second constraint is the total area available for the photovoltaic panels and which is constrained by the area on the aircraft wings:

$$\sum_{l=1}^o A_{PVS} \leq A_{wing,max}. \quad (18)$$

3.3. Fitness Function. Different techniques for handling constraints in evolutionary algorithms have been proposed in the literature [23]. In this work, constraints are handled by penalizing the objective function. Using this technique, the

that allows aircrafts to fly during one year. For this reason, we propose a maximization of the power system efficacy as is described by the objective function stated by (16) and which is based on (3):

fitness function is constituted by the objective function plus the penalization terms, as follows:

$$F^{fitness} = F^{obj} - \text{abs} \left\{ \sum_{n=1}^i \left(K_1 \left(\sum_{(PVS,URFC,ctrl,structure)} W_j \right) \right) \right\}, \quad (19)$$

where K_1 is a penalization constant and W_j is the weight of the power system evaluated in order to find the maximum efficacy. In each PSO iteration, particular systems formed by the n particles are evaluated to meet the flight time of the aircraft during the year with the best efficacy.

The flowchart of the proposed optimization process is depicted in Figure 4.

The main steps of the proposed optimization process are described in the following subsections.

3.4. Input Data and Initial Population of the PSO. The input data for optimizing the energy system by PSO are the solar resource available at the site as well as aircraft specifications and power system elements. The PSO determines the optimal configuration by evaluating photovoltaic, fuel cell and storage powers needed to maintain the aircraft flying. The optimizer generates three vectors with n particles each one (in this paper $n = 50$ particles), where each vector represents the photovoltaic, fuel cell and storage powers. Each power system is evaluated over a period of 8640 hours with the operating strategy depicted in Figure 5.

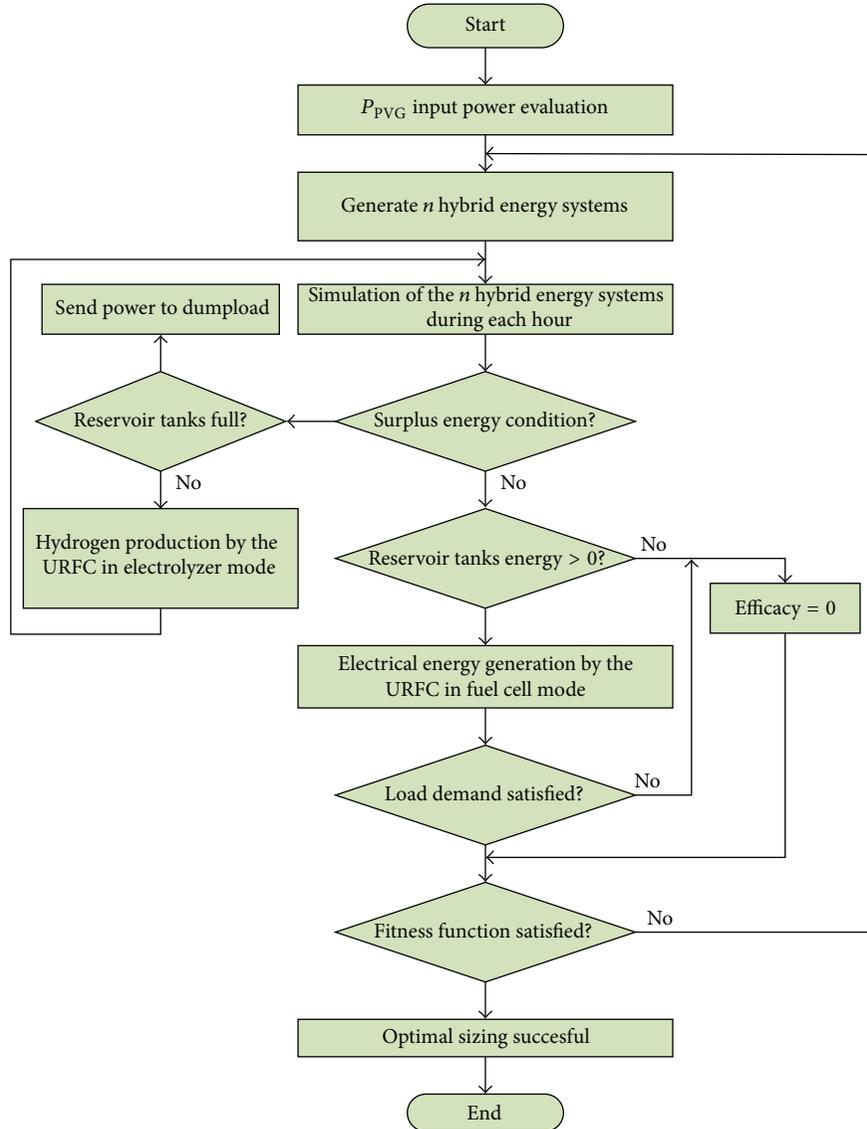


FIGURE 5: Flowchart of the energy management strategy for optimal sizing.

4. Cases of Study and Computational Results

In order to validate the sizing strategy for the energy system proposed in this work, we have selected three HALE aircrafts as benchmark test. The total mass, empty mass, and wing area of the HALE aircrafts were obtained from the open literature and they are shown in Table 1 [7].

We have selected Chetumal city, Mexico (18° latitude), as testing site in order to estimate the solar resource available by means of (5). One annual cycle is used as period of time for the analysis in this study. Table 2 shows other input data assumed in this work.

On the other hand, Table 3 shows the PSO parameters used in order to solve the sizing problem. A linear decreasing function for the inertia term is used in order to reduce the influence of past velocities.

TABLE 3: PSO parameters.

Item	Symbol	Value
Particles	—	50
Maximum number of iterations	—	300
Acceleration constants	C_1	2
	C_2	2
Initial inertia weight	w_b	0.9
Final inertia weight	w_f	0.4

A set of ten simulations for the power system sizing of each HALE aircraft was performed in order to obtain trustworthy results, due to the stochastic nature of the PSO. Thereby, Figure 6 shows the PSO convergence rate for the efficacy maximization of the HALE aircrafts power systems.

TABLE 4: Cases of study results.

Aircraft		PPVS/kW	PSS/kWh	PURFC/kW	MASS/kg	EFFS/%
Pathfinder	Mean	5.37	24.49	11.51	191.01	46.82
	StDev	0.25	3.90	1.50	5.16	1.05
	(CV)	(4.6%)	(15.9%)	(13.0%)	(2.7%)	(2.2%)
	Min	5.20	22.92	8.38	187.92	44.55
	Max	5.97	35.18	12.22	205.36	47.60
Pathfinder plus	Mean	6.04	32.36	15.01	224.09	47.81
	StDev	0.33	6.04	0.95	7.69	0.46
	(CV)	(5.4%)	(18.7%)	(6.3%)	(3.4%)	(1.0%)
	Min	5.80	28.88	12.40	217.77	46.90
	Max	6.70	44.33	15.40	238.66	48.28
Helios HP01	Mean	18.31	88.88	33.58	589.91	45.82
	StDev	0.80	11.19	4.01	14.89	0.92
	(CV)	(4.4%)	(12.6%)	(12.0%)	(2.5%)	(2.0%)
	Min	17.10	73.81	24.35	565.60	44.22
	Max	20.01	104.44	36.28	608.09	46.80

Table 4 presents the optimization results of each study. Nominal powers of PVS (PPVS), storage system (PSS), URFC (PURFC), HALE total mass, and global efficacy values are shown in the columns of Table 4. A statistical analysis is performed on the results obtained for each HALE aircraft studied. Mean values, standard deviation, and maximum and minimum values are shown also.

A variation coefficient (CV) is used as an indicator of the difference between the standard deviation and the mean value in each case study. According to CV value, the results obtained by PSO show a low dispersion.

The optimization results indicate that a mean value of 46.82% for the efficacy is obtained for HALE Pathfinder, with an energy power system of 5.37 kW, 24.29 kWh, and 11.51 kW, for PPVS, PSS, and PURFC, respectively. This configuration implicates a total mass of 191 kg (mean value) for HALE aircraft, which represents 60.99 kg (24.2%) less than the original total mass according to Table 1.

Similarly, the results obtained indicate that the total mass for the Pathfinder Plus mass is 90.91 kg (28.86%) lighter with respect to the total mass reported in Table 1. The same case occurs with HELIOS HPI; the optimized configuration of the energy power system implicates an aircraft 129.09 kg (17.95%) more light. These results do not mean an increase of extrapayload; however, it would be used as a redesign parameter for the aircraft's structure. It is noteworthy that maximum and minimum values for the efficacy and the total mass are not associated with the power capacities of the power system elements that form the optimized configurations (PPVS, PSS, and PURFC).

Finally, the variation coefficient value has a maximum deviation for the storage power system (PSS, CV = 18.7% @ Pathfinder plus), whereas a less deviation of this parameter is obtained for PPVS (CV = 4.4% @ Helios HP01). Nevertheless, this parameter has small variations in efficacy and total mass values (2.2% and 3.4%, resp.).

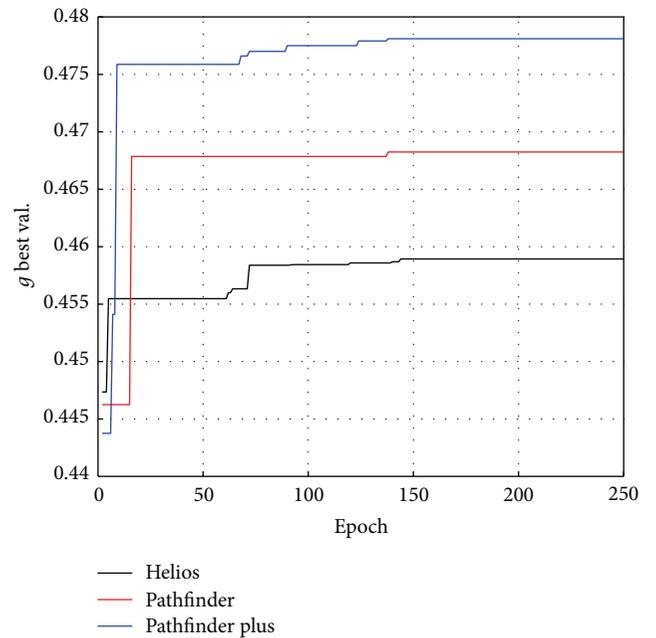


FIGURE 6: Convergence rate for PSO algorithm.

5. Conclusions

This paper has provided the optimal sizing of the energy power supply for HALE aircrafts by means of PSO. The method proposed allows an easy way to obtain the optimal configuration for the energy power system without deep knowledge about the relationship between the power generated by energy system elements, electric power demanded for the propeller, and HALE aircraft mass. Furthermore, optimal configurations for the photovoltaic/hydrogen systems consider the system thermodynamic efficiency during an annual cycle, and experimental data of local weather and the

electrochemical performance of the URFC were considered in the energy balance. On the other hand, although simplified models have been used for the power system components, the results obtained by the proposed approach enable the identification of opportunities for improvement of the design of HALE aircrafts. The results obtained demonstrate that the optimal sizing of the power system is in function of each component's weight. Besides, for the three HALE aircrafts the efficacy of the optimal power systems ranges between 45.82% to 47.81%. This result highlights the adequate combination of solar energy with hydrogen technology.

Also, we must note that the required PPVS is comparatively less than the PURFC in the power systems optimized, which shows an opportunity area to improve the power density in URFCs. Finally, the results obtained by PSO show a low dispersion, which demonstrates the robustness of the optimization process proposed in this work; besides that they contribute to the future implementation of HALE aircrafts in telecommunication applications.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors wish to thank PROMEP, CONACYT and Quintana Roo Government for supporting this project under Grants UQROO-EXB-072, Fordecyt 116157, and QR00-2011-001-174895, respectively.

References

- [1] S. Ohmori, Y. Yamao, and N. Nakajima, "Future generations of mobile communications based on broadband access technologies," *IEEE Communications Magazine*, vol. 38, no. 12, pp. 134–142, 2000.
- [2] T. C. Tozer and D. Grace, "High-altitude platforms for wireless communications," *Electronics & Communication Engineering Journal*, vol. 13, no. 3, pp. 127–137, 2001.
- [3] T. Van Do, G. Buchholz, L. Juhász, and J. Pakai, "Providing telecommunications services for rural areas from the HAP (High Altitude Platform): a case study in the region of Hungary," in *Proceedings of the Telecommunications Network Strategy and Planning Symposium*, pp. 1–5, November 2006.
- [4] Z. Yang and A. Mohammed, "Wireless communications from high altitude platforms: applications, deployment and development," in *Proceedings of the IEEE 12th International Conference on Communication Technology (ICCT '10)*, pp. 1476–1479, Nanjing, China, November 2010.
- [5] G. M. Djuknic, J. Freidenfelds, and Y. Okunev, "Establishing wireless communications services via high-altitude aeronautical platforms: a concept whose time has come?" *IEEE Communications Magazine*, vol. 35, no. 9, pp. 128–135, 1997.
- [6] S. D. Ilcev, "Stratospheric communication platforms as an alternative for space program," *Aircraft Engineering and Aerospace Technology*, vol. 83, no. 2, pp. 105–111, 2011.
- [7] G. Xian-Zhong, H. Zhong-Xi, G. Zheng, Z. Xiong-Feng, L. Jian-Xia, and C. Xiao-Qian, "Parameters determination for concept design of solar-powered, high-altitude long-endurance UAV," *Aircraft Engineering and Aerospace Technology*, vol. 85, no. 4, pp. 293–303, 2013.
- [8] S. Jashnani, T. R. Nada, M. Ishfaq, A. Khamker, and P. Shaholia, "Sizing and preliminary hardware testing of solar powered UAV," *The Egyptian Journal of Remote Sensing and Space Science*, vol. 16, no. 2, pp. 189–198, 2013.
- [9] O. Erdinc and M. Uzunoglu, "Recent trends in PEM fuel cell-powered hybrid systems: investigation of application areas, design architectures and energy management approaches," *Renewable and Sustainable Energy Reviews*, vol. 14, no. 9, pp. 2874–2884, 2010.
- [10] B. Khandelwal, A. Karakurt, P. R. Sekaran, V. Sethi, and R. Singh, "Hydrogen powered aircraft: the future of air transport," *Progress in Aerospace Sciences*, vol. 60, pp. 45–59, 2013.
- [11] G. Gahleitner, "Hydrogen from renewable electricity: An international review of power-to-gas pilot plants for stationary applications," *International Journal of Hydrogen Energy*, vol. 38, no. 5, pp. 2039–2061, 2013.
- [12] S. Dutta, "A review on production, storage of hydrogen and its utilization as an energy resource," *Journal of Industrial and Engineering Chemistry*, vol. 20, no. 4, pp. 1148–1156, 2014.
- [13] M. A. Elhadidy and S. M. Shaahid, "Promoting applications of hybrid (wind+photovoltaic+diesel+battery) power systems in hot regions," *Renewable Energy*, vol. 29, no. 4, pp. 517–528, 2004.
- [14] S. Kalogirou, *Solar Energy Engineering: Process and Systems*, Elsevier, London, UK, 1st edition, 2009.
- [15] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science (MHS '95)*, pp. 39–43, Nagoya, Japan, October 1995.
- [16] L. L. Lai, *Intelligent System Applications in Power Engineering—Evolutionary Programming and Neural Networks*, John Wiley & Sons, New York, NY, USA, 1998.
- [17] M. R. Al-Rashidi and M. E. El-Hawary, "A survey of particle swarm optimization applications in electric power systems," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 913–918, 2009.
- [18] K. Mahadevan and P. S. Kannan, "Comprehensive learning particle swarm optimization for reactive power dispatch," *Applied Soft Computing Journal*, vol. 10, no. 2, pp. 641–652, 2010.
- [19] S. M. Hakimi and S. M. Moghaddas-Tafreshi, "Optimal sizing of a stand-alone hybrid power system via particle swarm optimization for Kahnouj area in south-east of Iran," *Renewable Energy*, vol. 34, no. 7, pp. 1855–1862, 2009.
- [20] A. Y. Alanis, E. Rangel, J. Rivera, N. Arana-Daniel, and C. Lopez-Franco, "Particle swarm based approach of a real-time discrete neural identifier for linear induction motors," *Mathematical Problems in Engineering*, vol. 2013, Article ID 715094, 9 pages, 2013.
- [21] X. Z. Gao, Z. X. Hou, Z. Guo, J. X. Liu, and X. Q. Chen, "Energy management strategy for solar-powered high-altitude long-endurance aircraft," *Energy Conversion and Management*, vol. 70, pp. 20–30, 2013.
- [22] Y. H. Shiand and R. C. Eberhart, "Parameter selection in particle swarm optimizer," in *Proceedings of the 7th Annual Conference Evolutionary Programming*, pp. 25–27, 1998.
- [23] Z. Michalewicz, N. Attia, A. V. Sebald, and L. Fogel, "Evolutionary optimization of constrained problems," in *Proceedings of the 3rd Annual Conference Evolutionary Programming*, pp. 98–108, 1994.

Research Article

Hybrid Biogeography Based Optimization for Constrained Numerical and Engineering Optimization

Zengqiang Mi,¹ Yikun Xu,¹ Yang Yu,¹ Tong Zhao,¹ Bo Zhao,² and Liqing Liu¹

¹State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, North China Electric Power University, Baoding 071003, China

²Electronics-Controlling Lab of Construction Vehicle, Jilin University, Changchun 130022, China

Correspondence should be addressed to Yang Yu; ncep_u.yy@163.com

Received 16 April 2014; Accepted 18 July 2014

Academic Editor: Baozhen Yao

Copyright © 2015 Zengqiang Mi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Biogeography based optimization (BBO) is a new competitive population-based algorithm inspired by biogeography. It simulates the migration of species in nature to share information. A new hybrid BBO (HBBO) is presented in the paper for constrained optimization. By combining differential evolution (DE) mutation operator with simulated binary crossover (SBX) of genetic algorithms (GAs) reasonably, a new mutation operator is proposed to generate promising solution instead of the random mutation in basic BBO. In addition, DE mutation is still integrated to update one half of population to further lead the evolution towards the global optimum and the chaotic search is introduced to improve the diversity of population. HBBO is tested on twelve benchmark functions and four engineering optimization problems. Experimental results demonstrate that HBBO is effective and efficient for constrained optimization and in contrast with other state-of-the-art evolutionary algorithms (EAs), the performance of HBBO is better, or at least comparable in terms of the quality of the final solutions and computational cost. Furthermore, the influence of the maximum mutation rate is also investigated.

1. Introduction

With the development of science and engineering, related optimization problems become more and more complex. Optimization methods are being confronted with great challenges brought by some undesirable but unavoidable characteristics of optimization problems such as being high-dimensional, nondifferentiable, nonconvex, noncontinuous, and so on. Efficient optimization methods are urgently required by the complicated optimization problems in the real world. Therefore, various evolutionary algorithms have been applied to solve difficult optimization problems in recent decades, which include GAs [1], particle swarm optimization approach (PSO) [2], DE [3, 4], ant colony optimization (ACO) [5], artificial bee colony strategy (ABC) [6, 7], and BBO [8, 9].

Biogeography based optimization (BBO) is a new population-based algorithm. It simulates the blossom and extinction of species in different habitats based on the mathematical model of biogeography. Decision variables of better solutions tend to be shared in the migration operation and decision

variables of each solution are probabilistically replaced to improve the diversity of population in mutation operation. Due to good search ability, BBO has been applied to PID parameter tuning [10], parameter estimation of chaotic system [11], complex system optimization [12], satellite image classification [13], and so forth.

In comparison with other EAs, owing to direct-copying-based migration and random mutation, exploration ability of BBO is not so efficient despite outstanding exploitation. In other words, BBO can be easily trapped into local optimum and suffer from premature convergence owing to lack of corresponding exploration to balance its exploitation.

In order to overcome the weakness of BBO, lots of improved BBO variants have been proposed. Ma [14] presented six different migration mathematical models and made basic improvements on the migration operator. Gong et al. [15] hybridized the DE with the migration operator of BBO to balance the exploration and exploitation of BBO. Motivated by blended crossover operator in GAs, Ma and Simon [16] got decision variables of an offspring by blending

corresponding decision variables of two parent individuals based on different weighting constants. Li and Yin [17] proposed multiparent migration in which three consecutive individuals are chosen to generate three new individuals by basic BBO and then the new individuals are modified like multiparent crossover in GA; besides, the mutation was improved based on Gaussian operator. Li et al. [18] updated the decision variables not selected in migration by generating a perturbation from the neighborhood solutions and Gaussian operator was integrated into mutation. Wang and Xu [11] integrated DE mutation operator into migration operator of BBO and simplex search was introduced to improve the searching accuracy. Sayed et al. [10] formed new decision variables of an offspring by combining corresponding decision variables from two different parents with weighted constants related to the rank of their fitness in migration. Boussaïd et al. [19] proposed a new hybrid BBO in which new solutions are first generated by DE mutation and then modified by migration of original BBO. Xiong et al. [20] utilized four individuals' features to construct a new solution in proposed polyphyletic migration and orthogonal learning was introduced to further enhance converge speed toward global optimum.

In order to balance the exploration and exploitation of BBO, a new hybrid BBO called as HBBO is proposed in the paper. The unique points of HBBO are shown as the following. On one hand, a new hybrid mutation operator combining DE mutation and SBX is presented in HBBO while operators of EAs are often hybridized with migration operator in most of BBO variants. On the other hand, HBBO provides a new method to extend BBO to optimize constrained problems well due to only a few BBO variants available for constrained optimization in previous literatures. In addition, DE is applied to evolve one half of population to improve convergence speed further and chaotic search is introduced to enhance the diversity of population. Experiments have been conducted on twelve benchmark functions and four engineering optimization problems, and HBBO is compared with many other state-of-the-art algorithms from the quality of solutions obtained and computational cost. Furthermore, the influence of maximum mutation rate on HBBO is studied.

The rest of the paper is organized as follows. Constrained optimization, basic BBO, mutation strategies of DE, and SBX are briefly introduced in Section 2. In Section 3, the HBBO method proposed in the paper is specifically depicted. The comparison with six state-of-the-art algorithms on twelve benchmark functions is presented in Section 4. In Section 5, HBBO is compared with other methods on four well known engineering optimization problems. Section 6 further demonstrates the efficiency of HBBO and presents the investigation on the influence of maximum mutation rate. Finally, the work is concluded in Section 7.

2. Preliminary

2.1. Constrained Optimization. Constrained optimizations are always inevitable in scientific study and engineering

design. A general constrained optimization problem can be written as the following

$$\begin{aligned} & \text{Minimize} && f(\mathbf{x}) \\ & \text{subject to} && g_i(\mathbf{x}) \leq 0, \quad i = 1, 2, \dots, p \\ & && h_j(\mathbf{x}) = 0, \quad j = 1, 2, \dots, q, \end{aligned} \quad (1)$$

where \mathbf{x} represents the solution vector $\mathbf{x} = [x_1, x_2, \dots, x_D]^T$, D is the dimensionality of a solution in the paper, p is the number of inequality constraints, and q is the number of equality constraints. In common practice, equality constraints are often transformed to inequality constraints with a given small tolerance δ . For example, the equality constraint above can be converted to $|h_i(\mathbf{x})| \leq \delta$. In the paper, the feasible-based rule by Deb [21] is applied to handle constraint. In the constraint handling mechanism, fitness value and constraint violation are considered separately based on the following criterions: (a) any feasible solution is preferred to any infeasible solution; (b) between two feasible solutions, the one having smaller objective function value is preferred; (c) between two infeasible solutions, the one having smaller constraint violation is preferred.

2.2. Biogeography Based Optimization. Biogeography is the study of the distribution of species on earth surface over time. BBO is proposed based on the mathematical model of biogeography by Simon in 2008 [8]. In BBO, every solution is analogous to a habit; habit suitability index (HSI) is utilized to measure habits just like fitness function in other EAs; the elements that characterize habitability are called suitability index variables (SIVs) which are identical to the decision variables in other EAs. A good solution is similar to a habitat with high HSI which have a large number of species and vice versa. The species in habitats with high HSI tend to emigrate to habitats with low HSI. That is, habitats with high HSI tend to share their features while habitats with low HSI are inclined to accept the features from good habitats.

In BBO, each individual evolves by immigration and mutation operator. The SIVs of individuals are probabilistically shared in migration operator as shown in Algorithm 1, where X_{ij} is the j th SIV of i th individual in the population, λ_i here represents the immigration rate of X_i , and μ_k is the emigration rate of X_k , which are related to the number of species in the corresponding habitat; NP is the population size in the paper.

The following mathematical model is applied to calculate immigration rate and emigration rate owing to its outstanding performance in [22]:

$$\begin{aligned} \lambda_i &= I \left(1 - \frac{S_i}{n} \right) \\ \mu_i &= E \left(\frac{S_i}{n} \right)^2, \end{aligned} \quad (2)$$

where S_i is the number of species in habitat X_i , I and E are, respectively, the maximum value of immigration and emigration rate; n is equal to S_{\max} .

```

Target individual  $X_i$  for migration
For  $j = 1$  to  $D$  do
  Select  $X_{ij}$  with probability  $\propto \lambda_i$ 
  If  $X_{ij}$  is selected
    For  $k = 1$  to NP do
      Select  $X_k$  with probability  $\propto \mu_k$ 
      If  $X_k$  is selected
        Replace  $X_{ij}$  with  $X_{kj}$ 
      End if
    End for
  End if
End for

```

ALGORITHM 1: Migration operator.

```

Target individual  $X_i$  for mutation
For  $j = 1$  to  $D$ 
  Select  $X_{ij}$  based on  $m_i$ 
  If  $X_{ij}$  is selected
    Replace it with a randomly generated SIV
  End if
End for

```

ALGORITHM 2: Mutation operator.

In mutation operator, it is probabilistically decided whether or not to replace each SIV in a solution by a randomly generated SIV in the light of mutation rate. The detail of mutation operator is shown in Algorithm 2. The mutation rate m can be calculated as follows:

$$m_i = m_{\max} \left(1 - \frac{P_i}{P_{\max}} \right), \quad (3)$$

where $P_{\max} = \max(P_i)$, P_i represents the priori probability of existence for i th individual; m_{\max} is a user-defined parameter which represents the maximum mutation rate.

More details about basic BBO can be found in [8, 23].

2.3. Differential Evolution. DE algorithm is a population-based stochastic search method proposed by Storn and Price in 1997 [3]. Due to simple structure, few parameters, easy use, and fast convergence speed, DE has obtained wide application in various regions. DE generates new individuals by perturbing a randomly chosen individual with weighted differences for some couples of different individuals. Only when the offspring outperforms corresponding parent, the offspring survives as the parent for next generation. Mutation operator is the most important part in DE. In this part, only three widely applied mutation strategies are briefly introduced as follows:

(1) *rand/1*:

$$Y_{ij} = X_{r_1j} + F(X_{r_2j} - X_{r_3j}), \quad (4)$$

(2) *best/1*:

$$Y_{ij} = X_{g_j} + F(X_{r_1j} - X_{r_2j}), \quad (5)$$

(3) *rand to best/1*:

$$Y_{ij} = X_{r_1j} + F(X_{g_j} - X_{r_1j}) + F(X_{r_2j} - X_{r_3j}), \quad (6)$$

where X_g is the best individual in population; r_1, r_2, r_3 are uniformly distributed different numbers in the range $[1, NP]$; F is mutation scaling factor; Y represents the new individuals generated by mutation operator.

2.4. SBX of GA. Genetic algorithms simulate the evolutionary process in nature to solve optimization problems. In GA, some good individuals are chosen based on Deb's feasible-based rule. Different individuals can share information in crossover operator. SBX is one of the most popular crossover operators which can explore the neighborhood region of parent individual as follows:

$$C_{1j} = \frac{1}{2} \left[(1 - \beta_j) Parent_{1j} + (1 + \beta_j) Parent_{2j} \right] \quad (7)$$

$$C_{2j} = \frac{1}{2} \left[(1 + \beta_j) Parent_{1j} + (1 - \beta_j) Parent_{2j} \right], \quad (8)$$

where C_{ij} is the j th decision variable of the i th offspring individual; $Parent_{ij}$ is the j th decision variable of i th parent individual selected. β can be obtained by the random number u in $[0, 1]$ based on (9), where η is the distribution index for crossover. The detail of SBX can be found in [24]. Consider the following:

$$\beta(u) = \begin{cases} (2u)^{1/(\eta+1)} & \text{if } u \leq 0.5 \\ \frac{1}{[2(1-u)]^{1/(\eta+1)}} & \text{if } u > 0.5. \end{cases} \quad (9)$$

3. Proposed Algorithm HBBO

In mutation operator of basic BBO, SIVs are replaced probabilistically by new SIVs randomly generated. Although the mutation of BBO can improve the diversity of population, the random operation brings blindness to search. To modify the defect, a new hybrid mutation operator is proposed, in which DE mutation operator and SBX are mixed to generate promising SIV as shown in Algorithm 3. From Algorithm 3, it can be seen that two candidate SIVs are generated for each SIV mutated: one is gotten by DE *rand/1* mutation and the other by SBX. One point should be stated specially: X_{r_1} in DE *rand/1* mutation and $Parent_1$ and $Parent_2$ in SBX are all randomly selected from the first half of parent population which is sorted based on Deb's feasible rule (better one in front); X_{r_2}, X_{r_3} in DE *rand/1* mutation are randomly selected from the whole population. The core idea of hybrid mutation is based on the following considerations. First, owing to well-known performance in locating the region of global optimum, DE mutation can explore new search space with more clear direction towards global optimum instead of the random mutation in the original BBO. Second, SBX can explore the neighbor region of parent individual so that it can be combined with DE to explore search space efficiently. Third, the combination of DE mutation and SBX can balance the exploitation ability of BBO.

```

Target individual  $X_i$  for mutation;
For  $j = 1$  to  $D$ 
  If  $X_{ij}$  is selected for mutation as basic BBO
    Get two candidate SIVs of offspring;
    (1) Get a temp SIV by DE rand/1 mutation;
    (2) If  $rand < 0.5$  ( $rand$  is random number in  $[0, 1]$ )
      Get another temp SIV by (7);
    Else
      Get another temp SIV by (8);
    End if
  Else
    the  $j$ th SIV of  $i$ th individual in population Island survives as SIV of offspring;
    (population Island contains new individuals gotten by migration operator)
  End if
End for
Two temp offspring individuals are gotten for  $X_i$ , and the better one survives as offspring;

```

ALGORITHM 3: Hybrid mutation operator.

In order to speed up convergence, DE is further hybridized with BBO. The first half of parent population also evolves by two DE mutation strategies (*rand/1* and *rand to best/1*) and two new individuals are generated for each one in the first half. The best one among these two new individuals and corresponding parent individual survives to replace corresponding one in the second half of parent population.

For convenience and easy use, self-adaption mechanism for mutation scaling factor of DE proposed in [25] is applied, in which each individual is given an independent mutation scaling factor. The self-adaption mechanism is written as follows:

$$F_i^{G+1} = \begin{cases} F_l + rand_1 \cdot F_u & \text{if } rand_2 < 0.3 \\ F_i^G & \text{otherwise,} \end{cases} \quad (10)$$

where F_i^{G+1} represents mutation scaling factor for i th individual in $(G+1)$ th generation, F_i^G represents mutation scaling factor for i th individual in G th generation; F_l, F_u are the lower boundary and the change range of mutation scaling factor, respectively, and $rand_1$ and $rand_2$ are uniformly distributed numbers in $[0, 1]$.

Based on unique ergodicity, inherent stochastic property, and irregular chaos, chaotic search can reach each situation in given space so that it can contribute to the escape from the local optimum and is often integrated into EAs to enhance global search ability. Hence, the chaotic search is brought in for the first half of population. In the paper, logistic maps are used to generate chaotic sequences as follows:

$$W_{i+1,j} = \varepsilon \cdot W_{ij} \cdot (1 - W_{ij}) \quad 0 < W_{1j} < 1, \quad (11)$$

where $W_{1,j}, W_{i,j}, W_{i+1,j}$ are, respectively, the first i th, $(i+1)$ th element of sequence $\{W_{1,j}, W_{2,j}, \dots, W_{N,j}\}$; N represents the size of population for chaotic search; $j = 1, 2, \dots, D$; ε is the control parameter; sequence $\{W_{1,j}, W_{2,j}, \dots, W_{N,j}\}$ is chaotic when $\varepsilon = 4$ and $W_{1j} \neq 0.25, 0.5, 0.75$. We can apply the following equation to perform chaotic search for

i -th individual X_i in parent population by vector $W_i = [W_{i1}, W_{i2}, \dots, W_{iD}]^T$:

$$X'_i = X_i + R^G \cdot (2W_i - 1), \quad (12)$$

where X'_i is the new individual generated by chaotic search for X_i ; R^G represents the search radius vector in G th generation; each dimension in R^G represents the search radius for the variables in corresponding dimension of solutions in parent population.

In the initial phase, large chaotic search radius is helpful for escape from the local optimum; small chaotic search radius can improve the accuracy of search at the later stage of evolution. The search radius R_j^G for j th decision variable is adapted as follows:

$$R_j^G = Bound_j \cdot \left(0.005 - 0.004 \frac{G}{G_{\max}} \right), \quad (13)$$

where $Bound_j$ is the initial search radius for j th decision variable; G_{\max} is the maximum number of generations.

In order to maintain solutions feasible, any new decision variable generated in evolution process should be repaired if it violates boundary. Suppose that t_j is the j th decisions variable in certain new individual generated during evolution process. If t_j violates given boundaries, it can be modified as follows:

$$t_j = lb_j + (ub_j - lb_j) \cdot randnum_j, \quad (14)$$

where lb_j, ub_j are, respectively, the lower boundary and upper boundary of j th decision variable; $randnum_j$ is uniformly distributed number in $[0, 1]$ in each dimension.

The whole procedure of HBBO is described in Algorithm 4 in detail. From Algorithm 4, it can be seen that operations are mainly concentrated on the first half of population. It can be explained as follows. First, the convergence speed can be improved by focusing operations on the first half. Second, the information of the second half

```

Generate the initial population POP and vector F of mutation scaling factors;
Evaluate the fitness and constraint violations of each individual in POP;
For each generation do
  Sort the individuals in POP based on Deb's feasibility-based rule (better in front);
  For each one in POP's first half
    Get two new individuals by two DE mutation strategies (rand/1, rand to best/1);
    Evaluate the fitness value and constraint violations of these two new individuals;
    Among these two new individuals and corresponding parent individual, the best one is stored into population
    Tempbest;
  End for
  Update the vector P of prior probability;
  For each one in the first half of POP
    Generate a new individual by Algorithm 1, and store it into population Island;
  End for
  For each one in the first half of POP
    Get one offspring by Algorithm 3 and replace the corresponding individual in population Island with it;
  End for
  Go on chaotic search for the first half of POP and the new individuals generated are stored into population tempIsland;
  Make a contrast between the corresponding ones in Island and tempIsland, and the first half of POP, the best one survives
  as the corresponding one in POP for next generation;
  The population Tempbest replace the second half of POP as the parent ones for next generation;
  Update F, R by (10), (13) respectively;
End for

```

ALGORITHM 4: Hybrid biogeography based optimization.

TABLE 1: Main characteristics of the twelve selected benchmark functions.

Benchmark function	D	Type of objective function	ρ	LI	NI	NE	a
G01	13	Quadratic	0.0003%	9	0	0	6
G02	20	Nonlinear	99.9970%	1	1	0	1
G03	10	Nonlinear	0.0000%	0	0	1	1
G04	5	Quadratic	26.9668%	0	6	0	2
G05	4	Nonlinear	0.0000%	2	0	3	3
G06	2	Nonlinear	0.0064%	0	2	0	2
G07	10	Quadratic	0.0002%	3	5	0	6
G08	2	Nonlinear	0.8575%	0	2	0	2
G09	7	Nonlinear	0.5235%	0	4	0	2
G10	8	Linear	0.0007%	3	3	0	3
G11	2	Quadratic	0.0000%	0	0	1	1
G12	3	Quadratic	4.774%	0	9 ³	0	0

is also utilized in migration and hybrid mutation operator to generate promising solutions. Third, the risk of trapping into stagnation brought by concentration of operations can be relieved by chaotic search. Consequently, the focus of operations can make the search of HBBO efficient.

4. Simulation Tests on Benchmark Functions

4.1. Parameter Setting and Statistical Results Obtained by HBBO. In order to validate the performance of the proposed HBBO on numerical optimization, twelve benchmark test functions are adopted. The selected benchmark problems propose a good challenge and measure for constrained optimization techniques. Main characteristics of the selected benchmark functions are shown in detail in Table 1 where

D is the dimensionality of a solution for test function, ρ represents the ratio of feasible region to search space, NI is the number of nonlinear inequality constraints, LI is the number of linear inequality constraints, NE is the number of nonlinear equality constraints, and a is the number of constraints active at the optimal solution. The ρ metric can be computed as the following:

$$\rho = \frac{|F|}{|S|}, \quad (15)$$

where $|S|$ is the number of solutions generated randomly ($|S| = 1,000,000$ in the paper), and $|F|$ is the number of feasible solutions found in all the solutions randomly generated. All the benchmark functions selected are depicted explicitly in Appendix A.

TABLE 2: Statistic results for twelve benchmark functions obtained by HBBO.

Function	Optimal	Best	Mean	Median	Worst	SD	FFEs
G01	-15	-15	-14.799953	-15	-13	$6.10E - 1$	50,100
G02	-0.803619	-0.8036179	-0.7805965	-0.7852652	-0.7330360	$1.870E - 2$	75,200
G03	-1	-1.0050100	-1.0050100	-1.0050100	-1.0050100	$2.88E - 12$	150,100
G04	-30665.539	-30665.53867	-30665.5387	-30665.5387	-30665.5387	$1.11E - 11$	37,600
G05	5126.4981	5126.4842	5126.4842	5126.4842	5126.4842	$9.23E - 4$	375,100
G06	-6961.81388	-6961.81388	-6961.81388	-6961.81388	-6961.81388	$3.70E - 12$	50,100
G07	24.3062091	24.3062091	24.3062091	24.3062091	24.3062091	$6.49E - 10$	125,400
G08	-0.095825	-0.095825	-0.095825	-0.095825	-0.095825	$1.06E - 17$	10,100
G09	680.630057	680.630057	680.630057	680.630057	680.630057	$5.16E - 13$	75,100
G10	7049.248021	7049.248021	7049.248021	7049.248021	7049.248021	$1.08E - 7$	137,600
G11	0.75	0.74990	0.74990	0.74990	0.74990	$1.13E - 16$	75,100
G12	-1	-1	-1	-1	-1	0	12,600

For each test function, we performed 30 independent runs in matlab 7.0. The parameters of HBBO for experiments are set as follows: $E = I = 1$ is chosen as recommended in [8]; m_{\max} is set to be 0.8 which is much bigger than the corresponding value in basic BBO because big m_{\max} can improve mutation probabilities of individuals in population and enhance population diversity; based on the suggestions of mutation factor in DE in [3] and numerous experiments, $F_l = 0.75$ and $F_u = 0.15$ are chosen; $\eta = 20$ is chosen in the light of the effect of η on the search ability of SBX [24].

Through various tests, an appropriate set of population size NP for all the selected functions is found with which HBBO can present desirable performance. In the set found, population size NP for each benchmark function is given as the following: 200 for G02, 150 for G07, and 100 for the rest of benchmark functions. In each run, the maximum generations are given as the following: 200 for G01 and G06, 150 for G02 and G04, 600 for G03, 1500 for G05, 334 for G07, 40 for G08, 300 for G09 and G11, 550 for G10, and 50 for G12. In G03 and G05, the toleration value for equation constraint equals 0.001 as recommended in [19]; the toleration value for equation constraint of G11 is set to be 0.0001 as suggested in [26].

Table 2 summarizes the statistical features of results for twelve test functions obtained by HBBO and number of fitness function evaluations (FFEs) required. From Table 2, we can see that HBBO can get optimal solution in all 30 runs for seven benchmark functions (G04, G06, G07, G08, G09, G10, and G12); for G01, HBBO can get the optimal solutions in some runs; the *best* results obtained by HBBO are very close to the known best solution for G02; for three benchmark functions (G03, G05, and G11), the results gained by HBBO are very close to the optimal solutions or the known best solution.

4.2. Comparison with Other State-of-the-Art Methods. In this part, the proposed approach HBBO is compared with other six state-of-the-art optimization technologies.

The following are the six state-of-the-art optimization technologies: conventional BBO with DE mutation technology (CBO-DM) [19], hybrid PSO with DE strategy (PSO-DE) [27], coevolutionary DE algorithm (CDE) [28], changing

range genetic algorithm (CRGA) [26], self-adaptive penalty function based algorithm (SAPF) [29], and simple multimembered evolution strategy (SMES) [30]. The statistic results of other six approaches are compared with that of HBBO in Table 3, which are gotten from the original references. The “NA” in tables of the paper indicates the results of compared algorithms are not available. It should be noted that the best results obtained by algorithms are marked in boldface in the following tables. As far as computational cost is concerned, CBO-DM, SAPF, CDE, SMES, respectively, need 350,000, 500,000, 248,000, and 240,000 FFEs for all the test functions; PSO-DE needs 70,100 FFEs for G04, 17,600 FFEs for G12, and 140,100 FFEs for the rest of test functions; 1,350 to 68,000 FFEs are required for CRGA; the computational cost for CRGA is given in detail in [26].

With respect to CBO-DM, a variant of BBO, similar results are obtained by HBBO for five functions (G04, G06, G08, G09, and G12); in two functions (G07, G10), HBBO has better performance in the respect of considered metrics (*best*, *mean*, and *worst*); in G02, HBBO gets better *best* value with greater variability; the results of HBBO are obviously inferior but comparable for G01; the results obtained by HBBO are only lightly inferior for three test functions (G03, G05, and G11). In addition, the computational cost is far less than that of CBO-DM for all selected benchmark functions except G05. Consequently, HBBO is powerful competitor for CBO-DM on constrained optimization.

In contrast with other five state-of-the-art methods, the performance of HBBO is obviously inferior for function G01; HBBO can get better or similar solutions for the selected test functions except for G01, G03, and G11. In G03 and G11, the results obtained by HBBO are only lightly inferior to those of SMES. Furthermore, the computational cost is very competitive with respect to other methods for all selected test functions except G05.

5. Simulation Tests on Engineering Optimization Problems

In this part, four well-known engineering optimization problems are utilized to validate the performance of HBBO on

TABLE 3: Statistical features of results for twelve benchmark functions obtained by HBBO and other six state-of-the-art algorithms.

Function	Metrics	HBBO	CBO-DM	PSO-DE	CRGA	SAPF	SMES	CDE
G01	Best	-15	-15.000	15.000000	-14.9977	-15.000	-15.000	-15.0000
	Mean	-14.799953	-15.000	-15.000000	-14.9850	-14.552	-15.000	-15.0000
	Worst	-13	-15.000	-15.000000	-14.9467	-13.097	-15.000	-15.0000
G02	Best	-0.8036179	-0.803557	-0.8036145	-0.802959	-0.803202	-0.803601	-0.794669
	Mean	-0.7805965	-0.802774	-0.756678	-0.764494	-0.755798	-0.785238	-0.785480
	Worst	-0.7330360	-0.792576	-0.6367995	-0.722109	-0.745712	-0.751322	-0.779837
G03	Best	-1.0050100	-1.000	-1.0050100	-0.9997	-1.000	-1.000	NA
	Mean	-1.0050100	-1.000	-1.0050100	-0.9972	-0.964	-1.000	NA
	Worst	-1.0050100	-1.000	-1.0050100	-0.9931	-0.887	-1.000	NA
G04	Best	-30665.53867	-30665.539	-30665.539	-30665.520	-30665.401	-30665.539	-30665.539
	Mean	-30665.5387	-30665.539	-30665.539	-30664.398	-30665.922	-30665.539	-30665.536
	Worst	-30665.5387	-30665.539	-30665.539	-30660.313	-30656.471	-30665.539	-30665.509
G05	Best	5126.4842	5126.498	NA	5126.500	5126.907	5126.599	NA
	Mean	5126.4842	5126.498	NA	5507.041	5214.232	5174.492	NA
	Worst	5126.4842	5126.498	NA	6112.075	5564.642	5304.167	NA
G06	Best	-6961.81388	-6961.814	-6961.8139	-6956.251	-6961.046	-6961.814	-6961.814
	Mean	-6961.81388	-6961.814	-6961.8139	-6740.288	-6953.061	-6961.284	-6960.603
	Worst	-6961.81388	-6961.814	-6961.8139	-6077.123	-6943.304	-6952.482	-6901.285
G07	Best	24.3062091	24.326	24.3062091	24.882	24.838	24.327	NA
	Mean	24.3062091	24.345	24.306210	25.746	27.328	24.475	NA
	Worst	24.3062091	24.378	24.3062172	27.381	33.095	24.843	NA
G08	Best	-0.095825	-0.095825	-0.095826	-0.095825	-0.095825	-0.095825	NA
	Mean	-0.095825	-0.095825	-0.095826	-0.095819	-0.095635	-0.095825	NA
	Worst	-0.095825	-0.095825	-0.095826	-0.095808	-0.092697	-0.095825	NA
G09	Best	680.630057	680.630	680.63006	680.726	680.773	680.632	680.771
	Mean	680.630057	680.630	680.63006	681.347	681.246	680.643	681.503
	Worst	680.630057	680.630	680.63006	682.965	682.081	680.719	685.144
G10	Best	7049.248021	7059.802	7049.2480	7114.743	7069.981	7051.903	NA
	Mean	7049.248021	7075.832	7049.2480	8785.149	7238.964	7253.047	NA
	Worst	7049.248021	7098.254	7049.2482	10826.09	7489.406	7638.366	NA
G11	Best	0.74990	0.75	0.749999	0.750	0.749	0.75	NA
	Mean	0.74990	0.75	0.749999	0.752	0.751	0.75	NA
	Worst	0.74990	0.75	0.749999	0.757	0.757	0.75	NA
G12	Best	-1	-1.000000	1.000000	-1.000000	-1.000000	-1.000	-1.000000
	Mean	-1	-1.000000	-1.000000	-1.000000	-0.99994	-1.000	-1.000000
	Worst	-1	-1.000000	-1.000000	-1.000000	-0.999548	-1.000	-1.000000

solving real-world optimization problems. The four engineering optimization problems contain welded beam design problem, tension/compression spring design problem, speed reducer design problem, and three-bar truss design problem, which are listed in Appendix B. Parameters in HBBO for these four engineering optimization problems are as follows: population size and maximum generations are, respectively, 50 and 200 for welded beam design problem, 50 and 350 for tension/compression spring design problem, 100 and 100 for speed reducer design problem, and 50 and 60 for three-bar truss design problem; other parameters for HBBO are set in the same way as Section 4. For each engineering optimization problem, 30 independent runs are performed. Table 4 showed the statistic results for the four engineering optimization

problems solved by HBBO. We will evaluate performance of HBBO in respect of the quality of results and computational cost.

In order to demonstrate the superiority of HBBO, it is compared with other state-of-the-art algorithms on the four engineering problems. Welded beam and tension/compression spring design problems are also attempted by PSO-DE [27], CDE [28], coevolutionary particle swarm optimization (CPSO) [31], $(\mu + \lambda)$ -evolutionary strategy $((\mu + \lambda)$ -ES) [32], unified particle swarm optimization (UPSO) [33], and ABC [7]. PSO-DE [27], $(\mu + \lambda)$ -ES [32], and ABC [7] have also already performed on speed reducer design problem. PSO-DE [27] and Ray and Liew [34] have also been applied to solve three-bar truss design problem. The comparison of

TABLE 4: Statistic results for four engineering optimization problems solved by HBBO.

Engineering optimization problem	<i>Best</i>	<i>Mean</i>	<i>Median</i>	<i>Worst</i>	SD	FFEs
Welded beam design	1.724852309	1.724852309	1.724852309	1.724852309	$1.14E - 15$	25,050
Tension/compression spring design	0.012665233	0.012665393	0.012665234	0.012666698	$3.18E - 07$	43,800
Speed reducer design	2996.348165	2996.348165	2996.348165	2996.348165	$5.52E - 12$	25,100
Three-bar truss design	263.89584338	263.89584338	263.89584338	263.89584338	$7.08E - 14$	7,550

TABLE 5: Statistic results for welded beam design obtained by HBBO and other six state-of-the-art methods.

Method	<i>Best</i>	<i>Mean</i>	<i>Worst</i>	FFEs
HBBO	1.7248523	1.7248523	1.7248523	25,050
PSO-DE	1.7248531	1.7248579	1.7248811	33,000
CDE	1.733461	1.768158	1.824105	240,000
CPSO	1.728024	1.748831	1.782143	200,000
$(\mu + \lambda)$ -ES	1.724852	1.777692	NA	30,000
UPSO	1.92199	2.83721	NA	100,000
ABC	1.724852	1.741913	NA	30,000

TABLE 6: Statistic results of HBBO and other six state-of-the-art methods for tension/compression spring design.

Method	<i>Best</i>	<i>Mean</i>	<i>Worst</i>	FFEs
HBBO	0.012665233	0.012665393	0.012666698	43,800
PSO-DE	0.012665233	0.012665233	0.012665233	42,100
CDE	0.0126702	0.012703	0.012790	240,000
CPSO	0.0126747	0.01273	0.012924	200,000
$(\mu + \lambda)$ -ES	0.012689	0.013165	NA	30,000
UPSO	0.01312	0.02294	NA	100,000
ABC	0.012665	0.012709	NA	30,000

TABLE 7: Statistic results of HBBO and other six state-of-the-art methods for speed reducer design.

Method	<i>Best</i>	<i>Mean</i>	<i>Worst</i>	FFEs
HBBO	2996.348165	2996.348165	2996.348165	25,100
PSO-DE	2996.348165	2996.348165	2996.348166	70,100
$(\mu + \lambda)$ -ES	2996.348	2996.348	NA	30,000
ABC	2997.058	2997.058	NA	30,000

statistical results and computational cost for four engineering optimization problems between HBBO and other algorithms is shown in Tables 5, 6, 7, and 8.

From Tables 5, 6, 7, and 8, it can be seen that HBBO outperforms other compared algorithms for the given engineering optimization problems except tension/compression spring design problem for which PSO-DE has best performance. For tension/compression spring design problem, HBBO get similar *best* result and the *mean* and *worst* results of it are just lightly inferior in contrast with PSO-DE.

TABLE 8: Statistic results of HBBO and other six state-of-the-art methods for three-bar truss design.

Method	<i>Best</i>	<i>Mean</i>	<i>Worst</i>	FFEs
HBBO	263.89584338	263.89584338	263.89584338	7,550
PSO-DE	263.89584338	263.89584338	263.89584338	17,600
Ray and Liew	263.89584654	263.90335672	263.96975638	17,610

6. Discussions

In this part, HBBO is compared with the original BBO and self-adapting DE (SADE) to demonstrate the searching efficiency of HBBO further. In addition, the influence of maximum mutation rate on searching efficiency of HBBO is investigated.

6.1. Comparison with the Original BBO and SADE. The detail of the original BBO can be gotten from [8] and SADE proposed in [25] is compared. Specific parameter setting of these two algorithms is the same as the original references while the parameters related to test functions such as population size and constraint tolerance are in accordance with Section 4. Deb's feasible rule is applied in BBO and SADE to handle constraint. Here, HBBO is adopted in an identical way as described in Section 4.

Figure 1 illustrates typical evolution processes of objective function value of best solution in population when four benchmark functions (G02, G03, G07, and G09) are, respectively, solved by HBBO, BBO, and SADE. From Figure 1, it can be seen that HBBO have fastest convergence speed while BBO is often trapped into stagnation. So it can be concluded that the exploration and exploitation of BBO are well enhanced and balanced by new mutation operator and further hybridization with DE and chaotic search.

6.2. Influence of Maximum Mutation Rate on HBBO. The maximum value of mutation rate m_{\max} is related to the probability that individuals mutate by new hybrid mutation operator so that it affects the balance degree of exploration and exploitation of HBBO. In order to investigate the effect of m_{\max} on search efficiency of HBBO, m_{\max} is set to be different values including 0.05, 0.1, 0.4, 0.8, and 1. The investigation experiments are performed on five benchmark functions (G01, G02, G03, G07, and G10).

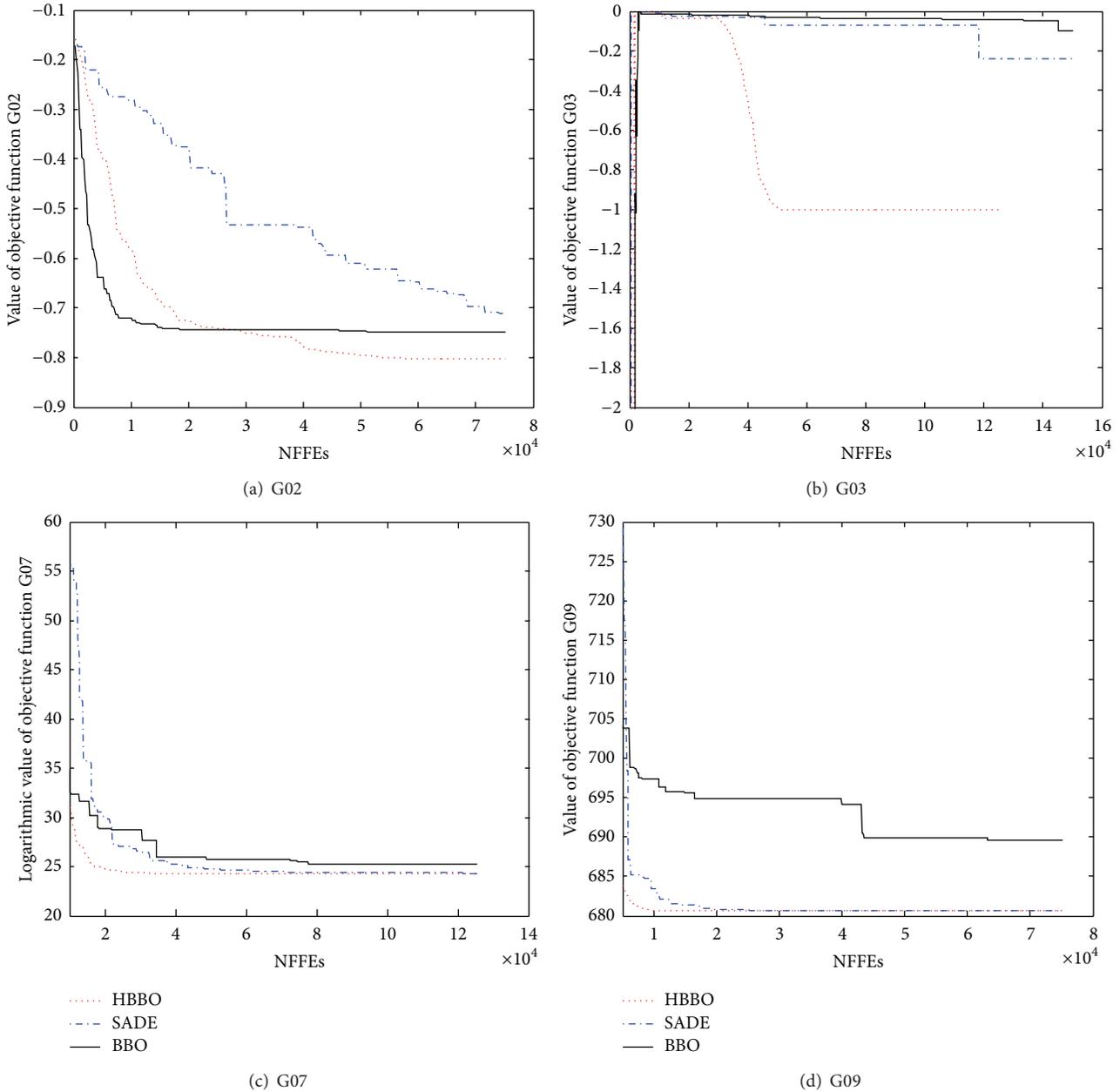


FIGURE 1: Objective function value curves of four test functions solved by HBBO, SADE, and BBO.

The other parameters are in accordance with description in Section 4. For each value of m_{\max} and test function, we perform 30 independent runs. The statistical features of results obtained by HBBO with different m_{\max} are summarized in Table 9.

From Table 9, we can see that the value of m_{\max} has significant influence on search efficiency of HBBO. The influence can be stated from three respects as the following. First, HBBO with different value of m_{\max} has different performance on the five test functions. Second, HBBO with too small or too big m_{\max} could not solve the selected test functions well while HBBO with middle value of m_{\max} has better comprehensive performance. Third, the fittest m_{\max} for each test function is different. The influence above may

be explained as follows. Too small m_{\max} cannot balance the exploitation of BBO well while too big m_{\max} will destroy the exploitation of BBO; different test functions have different characteristics so that they have different requirements of exploration and exploitation of optimization algorithm.

7. Conclusions

The paper proposes a new hybrid biogeography based optimization (HBBO) for constrained optimization. For the presented algorithm HBBO, a new mutation operator was proposed to generate promising solutions by merging DE mutation with SBX; a half of the population also evolved by two mutation strategies of DE. Chaotic search was introduced

TABLE 9: Statistical features of the results obtained by HBBO with different m_{\max} .

Function	Metrics	HBBO				
		$m_{\max} = 0.05$	$m_{\max} = 0.1$	$m_{\max} = 0.4$	$m_{\max} = 0.8$	$m_{\max} = 1$
G01	<i>Best</i>	-15	-15	-15	-15	-15
	<i>Mean</i>	-14.666665	-14.633610	-14.600000	-14.799953	-14.632939
	<i>Worst</i>	-12.999949	-12.008289	-13	-13	-12
	SD	$7.58E - 1$	$8.49E - 1$	$8.14E - 1$	$6.10E - 1$	$8.50E - 1$
G02	<i>Best</i>	-0.8036071	-0.8036171	-0.8036185	-0.8036179	-0.8036173
	<i>Mean</i>	-0.7780832	-0.7787019	-0.7799297	-0.7805965	-0.7775819
	<i>Worst</i>	-0.71807066	-0.7501355	-0.7248475	-0.7330360	-0.7225584
	SD	$2.03E - 2$	$1.58E - 2$	$1.83E - 2$	$1.87E - 2$	$2.19E - 2$
G03	<i>Best</i>	-1.0044781	-1.0050100	1.0050100	1.0050100	-1.0050100
	<i>Mean</i>	-0.7515293	-1.0050100	1.0050100	1.0050100	-1.0050100
	<i>Worst</i>	-0.2971747	-1.0050100	1.0050100	1.0050100	-1.0050100
	SD	$2.3E - 1$	$2.53E - 10$	$4.10E - 10$	$2.88E - 12$	$5.39E - 12$
G07	<i>Best</i>	24.3062091	24.3062091	24.3062091	24.3062091	24.3062091
	<i>Mean</i>	24.3062100	24.3062091	24.3062091	24.3062091	24.3062091
	<i>Worst</i>	24.3062351	24.3062091	24.3062091	24.3062091	24.3062091
	SD	$4.74E - 6$	$7.00E - 10$	$2.35E - 10$	$6.49E - 10$	$7.97E - 10$
G10	<i>Best</i>	7049.248021	7049.248021	7049.248021	7049.248021	7049.248021
	<i>Mean</i>	7054.178611	7049.248022	7049.248021	7049.248021	7049.248027
	<i>Worst</i>	7194.608835	7049.248076	7049.248021	7049.248021	7049.248200
	SD	26.52	$1.02E - 5$	$4.20E - 8$	$1.08E - 7$	$3.27E - 5$

for escape from stagnation and Deb's feasibility-based rule was applied to handle constraints. Furthermore, self-adaption mechanism for the mutation scaling factor of DE was utilized to avoid bothering of choosing an appropriate parameter.

Simulation experiments were performed on twelve benchmark test functions and four well-known engineering optimization problems. HBBO can obtain better or comparable results in contrast with other state-of-the-art optimization technologies. At the same time, the low computation cost is the obvious advantage of our HBBO. In short, HBBO is an effective and efficient method for constrained optimization. In addition, the influence of maximum mutation rate was investigated and the results demonstrate HBBO with maximum mutation rate of middle value has better comprehensive performance.

Maximum immigration rate I and emigration rate E affect the migration probability and the prior probability of existence for individuals in population. How to select the fittest E and I for HBBO will be one possible focus of our research in the future. Besides, HBBO will be extended to multiobjectives optimization.

Appendices

A. Benchmark Test Functions

A.1. G01.

$$\begin{aligned} \text{Minimize } & f(\vec{x}) = 5 \sum_{i=1}^4 x_i - 5 \sum_{i=1}^4 x_i^2 - \sum_{i=5}^{13} x_i \\ \text{subject to } & g_1(\vec{x}) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0, \\ & g_2(\vec{x}) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0, \end{aligned}$$

$$\begin{aligned} g_3(\vec{x}) &= 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0, \\ g_4(\vec{x}) &= -8x_1 + x_{10} \leq 0, \\ g_5(\vec{x}) &= -8x_2 + x_{11} \leq 0, \\ g_6(\vec{x}) &= -8x_3 + x_{12} \leq 0, \\ g_7(\vec{x}) &= -2x_4 - x_5 + x_{10} \leq 0, \\ g_8(\vec{x}) &= -2x_6 - x_7 + x_{11} \leq 0, \\ g_9(\vec{x}) &= -2x_8 - x_9 + x_{12} \leq 0, \end{aligned} \tag{A.1}$$

where the bounds are $0 \leq x_i \leq 1$ ($i = 1, \dots, 9$), $0 \leq x_i \leq 100$ ($i = 10, 11, 12$), and $0 \leq x_{13} \leq 1$. The global optimum is at $\vec{x}^* = (1, 1, 1, 1, 1, 1, 1, 1, 1, 3, 3, 3, 1)$ with

$$f(\vec{x}^*) = -15. \tag{A.2}$$

A.2. G02.

$$\begin{aligned} \text{Minimize } & f(\vec{x}) = - \left| \frac{\sum_{i=1}^n \cos^4(x_i) - 2 \prod_{i=1}^n \cos^2(x_i)}{\sqrt{\sum_{i=1}^n i x_i^2}} \right| \\ \text{subject to } & g_1(\vec{x}) = 0.75 - \prod_{i=1}^n x_i \leq 0, \\ & g_2(\vec{x}) = \sum_{i=1}^n x_i - 7.5n \leq 0, \end{aligned} \tag{A.3}$$

where $n = 20$ and $0 \leq x_i \leq 10$ ($i = 1, \dots, n$). The global optimum is unknown; the best objective function value reported is $f(\vec{x}^*) = -0.803619$.

A.3. G03.

$$\begin{aligned} \text{Maximize } f(\vec{x}) &= (\sqrt{n})^n \prod_{i=1}^n x_i \\ \text{subject to } h(\vec{x}) &= \sum_{i=1}^n x_i^2 - 1 = 0, \end{aligned} \quad (\text{A.4})$$

where $n = 10$ and $0 \leq x_i \leq 1$ ($i = 1, 2, \dots, n$). The optimum solution is $x_i^* = 1/\sqrt{n}$ ($i = 1, 2, \dots, n$) with $f(\vec{x}^*) = 1$.

A.4. G04.

$$\begin{aligned} \text{Maximize } f(\vec{x}) &= 5.3578547x_3^2 + 0.8356891x_1x_5 \\ &\quad + 37.293239x_1 - 40792.141 \\ \text{subject to } g_1(\vec{x}) &= 85.334407 + 0.0056858x_2x_5 \\ &\quad + 0.0006262x_1x_4 \\ &\quad - 0.0022053x_3x_5 - 92 \leq 0, \\ g_2(\vec{x}) &= -85.334407 - 0.0056858x_2x_5 \\ &\quad - 0.0006262x_1x_4 \\ &\quad + 0.0022053x_3x_5 \leq 0, \\ g_3(\vec{x}) &= 80.51249 + 0.0071317x_2x_5 \\ &\quad + 0.0029955x_1x_2 \\ &\quad + 0.0021813x_3^2 - 110 \leq 0, \\ g_4(\vec{x}) &= -80.51249 - 0.0071317x_2x_5 \\ &\quad - 0.0029955x_1x_2 \\ &\quad - 0.0021813x_3^2 + 90 \leq 0, \\ g_5(\vec{x}) &= 9.300961 + 0.0047026x_3x_5 \\ &\quad + 0.0012547x_1x_3 \\ &\quad + 0.0019085x_3x_4 - 25 \leq 0, \\ g_6(\vec{x}) &= -9.300961 - 0.0047026x_3x_5 \\ &\quad - 0.0012547x_1x_3 \\ &\quad - 0.0019085x_3x_4 + 20 \leq 0, \end{aligned} \quad (\text{A.5})$$

where $78 \leq x_1 \leq 102$, $33 \leq x_2 \leq 45$, and $27 \leq x_i \leq 45$ ($i = 3, 4, 5$). The optimum solution is $\vec{x}^* = (78, 33, 29.995256025682, 45, 36.775812905788)$ with $f(\vec{x}^*) = 30665.539$.

A.5. G05.

$$\begin{aligned} \text{Minimize } f(\vec{x}) &= 3x_1 + 0.000001x_1^3 + 2x_2 \\ &\quad + \left(\frac{0.000002}{3}\right)x_2^3 \\ \text{subject to } g_1(\vec{x}) &= -x_4 + x_3 - 0.55 \leq 0, \\ g_2(\vec{x}) &= -x_3 + x_4 - 0.55 \leq 0, \\ h_3(\vec{x}) &= 1000 \sin(-x_3 - 0.25) \\ &\quad + 1000 \sin(-x_4 - 0.25) + 894.8 \\ &\quad - x_1 = 0, \\ h_4(\vec{x}) &= 1000 \sin(x_3 - 0.25) \\ &\quad + 1000 \sin(x_3 - x_4 - 0.25) + 894.8 \\ &\quad - x_2 = 0, \\ h_5(\vec{x}) &= 1000 \sin(x_4 - 0.25) \\ &\quad + 1000 \sin(x_4 - x_3 - 0.25) \\ &\quad + 1294.8 = 0, \end{aligned} \quad (\text{A.6})$$

where $0 \leq x_1 \leq 1200$, $0 \leq x_2 \leq 1200$, $-0.55 \leq x_3 \leq 0.55$, and $-0.55 \leq x_4 \leq 0.55$. The best known solution is $\vec{x}^* = (679.9453, 1026.067, 0.1188764, 0.3962336)$, where $f(\vec{x}^*) = 5126.4981$.

A.6. G06.

$$\begin{aligned} \text{Minimize } f(\vec{x}) &= (x_1 - 10)^3 + (x_2 - 20)^3 \\ \text{subject to } g_1(\vec{x}) &= -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0, \\ g_2(\vec{x}) &= -(x_1 - 6)^2 - (x_2 - 5)^2 - 82.81 \leq 0, \end{aligned} \quad (\text{A.7})$$

where $13 \leq x_1 \leq 100$ and $0 \leq x_2 \leq 100$. The optimum solution is $\vec{x}^* = (14.095, 0.84296)$ with $f(\vec{x}^*) = -6961.81388$.

A.7. G07.

$$\begin{aligned} \text{Minimize } f(\vec{x}) &= x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 \\ &\quad + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 \\ &\quad + 2(x_6 - 1)^2 + 5x_7^2 \\ &\quad + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 \\ &\quad + (x_{10} - 7)^2 + 45 \end{aligned}$$

subject to $g_1(\vec{x}) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0$,
 $g_2(\vec{x}) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0$,
 $g_3(\vec{x}) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0$,
 $g_4(\vec{x}) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2$
 $+ 2x_3^2 - 7x_4 - 120 \leq 0$,
 $g_5(\vec{x}) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0$,
 $g_6(\vec{x}) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5$
 $- 6x_6 \leq 0$,
 $g_7(\vec{x}) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2$
 $+ 3x_5^2 - x_6 - 30 \leq 0$,
 $g_8(\vec{x}) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0$,

(A.8)

where $-10 \leq x_i \leq 10$ ($i = 1, 2, \dots, 10$). The optimum solution is $\vec{x}^* = (2.171996, 2.363683, 8.773926, 5.095984, 0.9906548, 1.430574, 1.321644, 9.828726, 8.280092, 8.375927)$ with $f(\vec{x}^*) = 24.3062091$.

A.8. G08.

Maximize $f(\vec{x}) = \frac{\sin^3(2\pi x_1) \sin(2\pi x_2)}{x_1^3(x_1 + x_2)}$
subject to $g_1(\vec{x}) = x_1^2 - x_2 + 1 \leq 0$,
 $g_2(\vec{x}) = 1 - x_1 + (x_2 - 4)^2 \leq 0$,

(A.9)

where $0 \leq x_1 \leq 10$ and $0 \leq x_2 \leq 10$. The optimal solution is $\vec{x}^* = (1.2279713, 4.2453733)$ where $f(\vec{x}^*) = 0.095825$.

A.9. G09.

Minimize $f(\vec{x}) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4$
 $+ 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4$
 $- 4x_6x_7 - 10x_6 - 8x_7$
subject to $g_1(\vec{x}) = -127 + 2x_1^2 + 3x_2^4 + x_3$
 $+ 4x_4^2 + 5x_5 \leq 0$,
 $g_2(\vec{x}) = -282 + 7x_1 + 3x_2 + 10x_3^2$
 $+ x_4 - x_5 \leq 0$,
 $g_3(\vec{x}) = -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0$,
 $g_4(\vec{x}) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2$
 $+ 5x_6 - 11x_7 \leq 0$,

(A.10)

where $-10 \leq x_i \leq 10$ ($i = 1, 2, \dots, 7$). The optimum solution is $\vec{x}^* = (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.038131, 1.594227)$ with $f(\vec{x}^*) = 680.630057$.

A.10. G10.

Minimize $f(\vec{x}) = x_1 + x_2 + x_3$
subject to $g_1(\vec{x}) = -1 + 0.0025(x_4 + x_6) \leq 0$,
 $g_2(\vec{x}) = -1 + 0.0025(x_5 + x_7 - x_4) \leq 0$,
 $g_3(\vec{x}) = -1 + 0.01(x_8 - x_5) \leq 0$,
 $g_4(\vec{x}) = -x_1x_6 + 833.33252x_4 + 100x_1$
 $- 83333.333 \leq 0$,
 $g_5(\vec{x}) = -x_2x_7 + 1250x_5 + x_2x_4$
 $- 1250x_4 \leq 0$,
 $g_6(\vec{x}) = -x_3x_8 + 1250000 + x_3x_5$
 $- 2500x_5 \leq 0$,

(A.11)

where $100 \leq x_1 \leq 10,000$, $1000 \leq x_i \leq 10,000$ ($i = 2, 3$), and $100 \leq x_i \leq 10,000$ ($i = 4, 5, \dots, 8$). The optimum solution is $\vec{x}^* = (579.3066, 1359.9707, 5109.9707, 182.0177, 295.601, 217.928, 286.165, 395.6012)$ with $f(\vec{x}^*) = 7049.248021$.

A.11. G11.

Minimize $f(\vec{x}) = x_1^2 + (x_2 - 1)^2$
subject to $h(\vec{x}) = x_2 - x_1^2 = 0$,

(A.12)

where $-1 \leq x_1 \leq 1$ and $-1 \leq x_2 \leq 1$. The optimum solution is $\vec{x}^* = (\pm 1/\sqrt{2}, 1/2)$ with $f(\vec{x}^*) = 0.75$.

A.12. G12.

Maximize $f(\vec{x}) = (100 - (x_1 - 5)^2 - (x_2 - 5)^2$
 $- (x_3 - 5)^2) \times (100)^{-1}$
subject to $g(\vec{x}) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2$
 $- 0.0625 \leq 0$,

(A.13)

where $0 \leq x_i \leq 10$ ($i = 1, 2, 3$) and $p, q, r = 1, 2, \dots, 9$. A point (x_1, x_2, x_3) is feasible if and only if there exist p, q, r such that the above inequality holds. The optimum solution is $\vec{x}^* = (5, 5, 5)$ with $f(\vec{x}^*) = 1$.

B. Engineering Design Problems

B.1. Welded Beam Design Problem. A welded beam is designed for the minimum cost subject to constraints on shear stress (τ); bending stress in the beam (θ); buckling load

on the bar (P_c); end deflection of the beam (δ); and side constraints. There are four design variables $h(x_1)$, $l(x_2)$, $t(x_3)$, and $b(x_4)$.

$$\begin{aligned}
 &\text{Minimize} && f(\vec{x}) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14 + x_2) \\
 &\text{subject to} && g_1(\vec{x}) = \tau(\vec{x}) - \tau_{\max} \leq 0, \\
 &&& g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{\max} \leq 0, \\
 &&& g_3(\vec{x}) = x_1 - x_4 \leq 0, \\
 &&& g_4(\vec{x}) = 0.1047x_1^2 + 0.04811x_3x_4(14 + x_2) \\
 &&& \quad - 5.0 \leq 0, \\
 &&& g_5(\vec{x}) = 0.125 - x_1 \leq 0, \\
 &&& g_6(\vec{x}) = \delta(\vec{x}) - \delta_{\max} \leq 0, \\
 &&& g_7(\vec{x}) = P - P_c(\vec{x}) \leq 0,
 \end{aligned} \tag{B.1}$$

where $\tau(\vec{x}) = \sqrt{(\tau')^2 + 2\tau'\tau''(x_2/2R) + (\tau'')^2}$, $\tau' = P/\sqrt{2}x_1x_2$, $\tau'' = MR/J$, $M = P(L + x_2/2)$, $R = \sqrt{x_2^2/4 + ((x_1 + x_3)/2)^2}$, $\delta(\vec{x}) = 4PL^3/Ex_3^3x_4$, $J = 2\{\sqrt{2}x_1x_2[x_2^2/12 + ((x_1 + x_3)/2)^2]\}$, $\sigma(\vec{x}) = 6PL/x_4x_3^2$, $P_c(\vec{x}) = (4.013E\sqrt{x_3^6x_4^6/36/L^2})(1 - (x_3/2L)\sqrt{E/4G})$, $P = 6000$ lb, $L = 14$ in, $E = 30 \times 10^6$ psi, $G = 12 \times 10^6$ psi, $\tau_{\max} = 13,600$ psi, $\sigma_{\max} = 30,000$ psi, $\delta_{\max} = 0.25$ in, $0.1 \leq x_1 \leq 2$, $0.1 \leq x_2 \leq 10$, $0.1 \leq x_3 \leq 10$, $0.1 \leq x_4 \leq 2$.

B.2. Tension/Compression Spring Design Problem. In this problem, the objective is to minimize the weight of a tension/compression spring subject to constraints on minimum deflection, shear stress, surge frequency, and limits on outside diameter and on design variables. The design variables are the mean coil diameter $D(x_2)$; the wire diameter $d(x_1)$; and the number of active coils $P(x_3)$.

$$\begin{aligned}
 &\text{Minimize} && f(\vec{x}) = (x_3 + 2)x_2x_1^2 \\
 &\text{subject to} && g_1(\vec{x}) = 1 - \frac{x_2^3x_3}{71785x_1^4} \leq 0, \\
 &&& g_2(\vec{x}) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0, \\
 &&& g_3(\vec{x}) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0, \\
 &&& g_4(\vec{x}) = \frac{x_1 + x_2}{1.5} - 1 \leq 0,
 \end{aligned} \tag{B.2}$$

where $0.05 \leq x_1 \leq 2$, $0.25 \leq x_2 \leq 1.3$, and $2 \leq x_3 \leq 15$.

B.3. Speed Reducer Design Problem.

$$\begin{aligned}
 &\text{Minimize} && f(\vec{x}) = 0.7854x_1x_2^2 \\
 &&& \quad \times (3.3333x_3^2 + 14.9334x_3 - 43.0934) \\
 &&& \quad - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) \\
 &&& \quad + 0.7854(x_4x_6^2 + x_5x_7^2) \\
 &\text{subject to} && g_1(\vec{x}) = \frac{27}{x_1x_2^2x_3} - 1 \leq 0, \\
 &&& g_2(\vec{x}) = \frac{397.5}{x_1x_2^2x_3^2} - 1 \leq 0, \\
 &&& g_3(\vec{x}) = \frac{1.93x_4^3}{x_2x_6^4x_3} - 1 \leq 0, \\
 &&& g_4(\vec{x}) = \frac{1.93x_5^3}{x_2x_7^4x_3} - 1 \leq 0, \\
 &&& g_5(\vec{x}) = \frac{[(745x_4/x_2x_3)^2 + 16.9 \times 10^6]^{1/2}}{110x_6^3} \\
 &&& \quad - 1 \leq 0, \\
 &&& g_6(\vec{x}) = \frac{[(745x_5/x_2x_3)^2 + 157.5 \times 10^6]^{1/2}}{85x_7^3} \\
 &&& \quad - 1 \leq 0, \\
 &&& g_7(\vec{x}) = \frac{x_2x_3}{40} - 1 \leq 0, \\
 &&& g_8(\vec{x}) = \frac{5x_2}{x_1} - 1 \leq 0, \\
 &&& g_9(\vec{x}) = \frac{x_1}{12x_2} - 1 \leq 0, \\
 &&& g_{10}(\vec{x}) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0, \\
 &&& g_{11}(\vec{x}) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0,
 \end{aligned} \tag{B.3}$$

where $2.6 \leq x_1 \leq 3.6$, $0.7 \leq x_2 \leq 0.8$, $17 \leq x_3 \leq 28$, $7.3 \leq x_4 \leq 8.3$, $7.3 \leq x_5 \leq 8.3$, $2.9 \leq x_6 \leq 3.9$, and $5.0 \leq x_7 \leq 5.5$.

B.4. Three-Bar Truss Design Problem.

$$\begin{aligned}
 &\text{Minimize} && f(\vec{x}) = (2\sqrt{2}x_1 + x_2) \times l \\
 &\text{subject to} && g_1(\vec{x}) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \leq 0,
 \end{aligned}$$

$$g_2(\vec{x}) = \frac{x_2}{\sqrt{2x_1^2 + 2x_1x_2}}P - \sigma \leq 0,$$

$$g_3(\vec{x}) = \frac{1}{\sqrt{2}x_2 + x_1}P - \sigma \leq 0,$$
(B.4)

where $0 \leq x_1 \leq 1$ and $0 \leq x_2 \leq 1$; $l = 100$ cm, $P = 2$ kN/cm², and $\sigma = 2$ kN/cm².

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research is supported by the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant no. 20120036130001, the Fundamental Research Funds for the Central Universities of China under Grant no. 2014MS93, and the Independent Research Funds of State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources of China under Grant no. 201414.

References

- [1] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading, Mass, USA, 1989.
- [2] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE Conference on Neural Networks*, vol. 4, pp. 1942–1948, November–December 1995.
- [3] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [4] R. Storn, "System design by constraint adaptation and differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 1, pp. 22–34, 1999.
- [5] C. Blum, "Ant colony optimization: Introduction and recent trends," *Physics of Life Reviews*, vol. 2, no. 4, pp. 353–373, 2005.
- [6] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [7] B. Akay and D. Karaboga, "Artificial bee colony algorithm for large-scale problems and engineering design optimization," *Journal of Intelligent Manufacturing*, vol. 23, no. 4, pp. 1001–1014, 2012.
- [8] D. Simon, "Biogeography-based optimization," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 6, pp. 702–713, 2008.
- [9] D. Simon, R. Rarick, M. Ergezer, and D. Du, "Analytical and numerical comparisons of biogeography-based optimization and genetic algorithms," *Information Sciences*, vol. 181, no. 7, pp. 1224–1248, 2011.
- [10] M. M. Sayed, M. S. Saad, H. M. Emara, and E. E. Abou El-Zahab, "A novel method for PID tuning using a modified biogeography-based optimization algorithm," in *Proceedings of the 24th Chinese Control and Decision Conference (CCDC '12)*, pp. 1642–1647, Taiyuan, China, May 2012.
- [11] L. Wang and Y. Xu, "An effective hybrid biogeography-based optimization algorithm for parameter estimation of chaotic systems," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15103–15109, 2011.
- [12] D. Du and D. Simon, "Complex system optimization using biogeography-based optimization," *Mathematical Problems in Engineering*, vol. 2013, Article ID 456232, 17 pages, 2013.
- [13] V. K. Panchal, P. Singh, N. Kaur, and H. Kundra, "Biogeography based satellite image classification," *International Journal of Computer Science and Information Security*, vol. 6, no. 2, pp. 269–274, 2009.
- [14] H. Ma, "An analysis of the equilibrium of migration models for biogeography-based optimization," *Information Sciences*, vol. 180, no. 18, pp. 3444–3464, 2010.
- [15] W. Gong, Z. Cai, and C. X. Ling, "DE/BBO: a hybrid differential evolution with biogeography-based optimization for global numerical optimization," *Soft Computing*, vol. 15, no. 4, pp. 645–665, 2010.
- [16] H. Ma and D. Simon, "Blended biogeography-based optimization for constrained optimization," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 3, pp. 517–525, 2011.
- [17] X. Li and M. Yin, "Multi-operator based biogeography based optimization with mutation for global numerical optimization," *Computers and Mathematics with Applications*, vol. 64, no. 9, pp. 2833–2844, 2012.
- [18] X. Li, J. Wang, J. Zhou, and M. Yin, "A perturb biogeography based optimization with mutation for global numerical optimization," *Applied Mathematics and Computation*, vol. 218, no. 2, pp. 598–609, 2011.
- [19] I. Boussaid, A. Chatterjee, P. Siarry, and M. Ahmed-Nacer, "Biogeography-based optimization for constrained optimization problems," *Computers and Operations Research*, vol. 39, no. 12, pp. 3293–3304, 2012.
- [20] G. Xiong, D. Shi, and X. Duan, "Enhancing the performance of biogeography-based optimization using polyphyletic migration operator and orthogonal learning," *Computers and Operations Research*, vol. 41, pp. 125–139, 2014.
- [21] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2–4, pp. 311–338, 2000.
- [22] W. Guo, L. Wang, and Q. Wu, "An analysis of the migration rates for biogeography-based optimization," *Information Sciences*, vol. 254, pp. 111–140, 2014.
- [23] D. Simon, "The Matlab code of biogeography-based optimization," 2008, <http://academic.csuohio.edu/simond/bbo/>.
- [24] M. M. Raghuvanshi and O. G. Kakde, "Survey on multiobjective evolutionary and real coded genetic algorithms," in *Proceedings of the 8th Asia Pacific Symposium on Intelligent and Evolutionary Systems*, pp. 150–161, Cairns, Australia, December 2004.
- [25] J. Brest, S. Greiner, B. Bošković, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, pp. 646–657, 2006.
- [26] A. Amirjanov, "The development of a changing range genetic algorithm," *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 19, pp. 2495–2508, 2006.

- [27] H. Liu, Z. Cai, and Y. Wang, "Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization," *Applied Soft Computing Journal*, vol. 10, no. 2, pp. 629–640, 2010.
- [28] F. Huang, L. Wang, and Q. He, "An effective co-evolutionary differential evolution for constrained optimization," *Applied Mathematics and Computation*, vol. 186, no. 1, pp. 340–356, 2007.
- [29] B. Tessema and G. G. Yen, "A self adaptive penalty function based algorithm for constrained optimization," in *Proceeding of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 246–253, Vancouver, Canada, July 2006.
- [30] E. Mezura-Montes and C. C. Coello, "A simple multimembered evolution strategy to solve constrained optimization problems," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 1, pp. 1–17, 2005.
- [31] Q. He and L. Wang, "An effective co-evolutionary particle swarm optimization for constrained engineering design problems," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 1, pp. 89–99, 2007.
- [32] M. E. Mezura and C. C. Coello, "Useful infeasible solutions in engineering optimization with evolutionary algorithms," in *Proceedings of the 4th Mexican International Conference on Artificial Intelligence*, pp. 625–662, Monterrey, Mexico, November 2005.
- [33] K. E. Parsopoulos and M. N. Vrahatis, "Unified Particle Swarm Optimization for solving constrained engineering optimization problems," in *Proceedings of the International Conference on Natural Computation (ICNC '05)*, pp. 582–591, Changsha, China, August 2005.
- [34] T. Ray and K. M. Liew, "Society and civilization: an optimization algorithm based on the simulation of social behavior," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 4, pp. 386–396, 2003.

Research Article

An Adapted Firefly Algorithm for Product Development Project Scheduling with Fuzzy Activity Duration

Minmei Huang,¹ Jijun Yuan,² and Jing Xiao³

¹*School of Public Administration, South China Normal University, Guangzhou 510006, China*

²*School of Finance, Guangdong University of Finance & Economics, Guangzhou 510320, China*

³*School of Computer Science, South China Normal University, Guangzhou 510631, China*

Correspondence should be addressed to Minmei Huang; huangmm@scnu.edu.cn

Received 3 June 2014; Revised 11 August 2014; Accepted 27 August 2014

Academic Editor: Fang Zong

Copyright © 2015 Minmei Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Efficient scheduling plays an important role in product development project management, especially for the product development project with fuzzy activity times. In this research a trapezoidal fuzzy number is used to represent fuzzy activity duration, and an improved magnitude of the trapezoidal fuzzy number is adopted for fuzzy time comparison. Firstly, a mathematical model for the scheduling problem with minimizing the project completion time for the product development project is established. Then, an adapted fuzzy firefly algorithm is developed to solve the model. The priority value based coding method is used; the fuzzy parallel schedule generation scheme is adopted to generate feasible solutions, and the brightness comparisons are made before updating fireflies' locations in the proposed algorithm. Finally, the performance of the proposed algorithm is presented by computational experiments based on PSPLIB benchmarks. An example of resource allocation of an electronic product development project is also used to illustrate the effectiveness and efficiency of the proposed algorithm.

1. Introduction

Product development has become a source of competitive advantage for many industries in recent years [1]. In order to bring the new product to the market as early as possible, it is essential to schedule product development project efficiently with the constraints of time, resources, and cost. Because of the uniqueness of new product development project and the influence of uncertainty, the precise prediction of activity duration is difficult. Thus the activity duration is always estimated by experts' experiences and is usually imprecise. In resource constrained project scheduling problems (RCPSP), time parameters are considered to be deterministic. Thus traditional approaches for RCPSP cannot accommodate product development project in uncertain environment.

Some research has been performed on fuzzy resource constrained project scheduling problems. The methods in the literature can be grouped into two categories: the heuristic methods and the artificial intelligence based methods. (1) For heuristic methods, traditional critical path method (CPM),

plan evaluation, and review technique (PERT) were adopted to solve project scheduling problems with uncertain times without resource constraints [2, 3]. Hapke and Slowinski [4] extended the known priority heuristic method for solving RCPSP with fuzzy time parameters. Bhaskar et al. [5] proposed a heuristic method for resource constrained project scheduling problem with fuzzy activity times. The heuristic method was based on priority rule for parallel schedule generation scheme, and the proposed priority rule was called Schedule Performance Index (SPI). (2) For artificial intelligence based methods, Wang [1] proposed the fuzzy beam search algorithm to determine a schedule with the minimum schedule risk, and the start time of each activity was selected to maximize the minimum satisfaction degrees of all temporal constraints. Wang [6] developed a genetic algorithm based on fuzzy set theory for uncertain product development projects. Ke and Liu [7] built three types of fuzzy models to solve the project scheduling problem with fuzzy activity duration and developed a hybrid intelligent algorithm to solve the fuzzy models.

The metaheuristics firefly algorithm (FA) inspired from intelligent social behavior of fireflies was recently presented by Yang [8]. Firefly algorithm is one of the biology-derived algorithms and it was proved by Yang [9] that FA is more efficient than particle swarm algorithms when dealing with multimodal functions. The FA has also been applied successfully to nonlinear design problems [10], constrained continuous optimization tasks [11], permutation flowshop scheduling problems [12], and resource constrained project scheduling problems [13, 14]. Yuan [15] proposed a modified firefly algorithm to solve multiobjective constraint optimization problem, and the proposed method was applied to the optimization design of motor product family. Yang [16] developed a multiobjective firefly algorithm (MOFA) for continuous optimization, and Luna et al. [17] applied MOFA to the software project scheduling problem.

The objective of this research is to develop an effective method by adapting firefly algorithm to handle product development project scheduling problem with fuzzy activity duration times. The paper is organized as follows. In Section 2 fuzzy set theory is used to represent uncertain activity duration; the magnitude of trapezoidal fuzzy number is employed to rank the fuzzy time parameters, and then the mathematical model for the problem is described. Section 3 proposes the fuzzy firefly algorithm, which is based on priority value coding and parallel schedule generation scheme, and an example is used to illustrate the algorithm. Computational experiments on 30 benchmark datasets and an electronic product development project are conducted in Section 4. Finally, Section 5 concludes the paper, and the future research directions are proposed.

2. Problem Description

2.1. Definition on Fuzzy Activity Duration. For most product development projects, it is difficult to precisely give activity durations. Activity durations are often estimated by human experts and confronted with vague and imprecise judgmental statements [6]. Therefore, fuzzy set theory can be used for estimating the activity times. A trapezoidal membership function can represent the most general form of a fuzzy number. We represent the activity duration with a trapezoidal fuzzy number in this paper.

Various definitions for the concept of fuzzy numbers are presented in literature. In this paper two of them are used for fuzzy activity duration.

Definition 1 (see [18]). A fuzzy activity duration \tilde{A} is a fuzzy set like $u : R \rightarrow [0, 1]$ satisfying the following properties:

- (1) u is upper semicontinuous;
- (2) $u(x) = 0$ if x is outside of interval $[a, d]$;
- (3) there are real numbers $a, b, c,$ and d such that $a \leq b \leq c \leq d$ and
 - (i) $u(x)$ is monotonic increasing on $[a, b]$;
 - (ii) $u(x)$ is monotonic decreasing on $[c, d]$;
 - (iii) $u(x) = 1, b \leq x \leq c$;

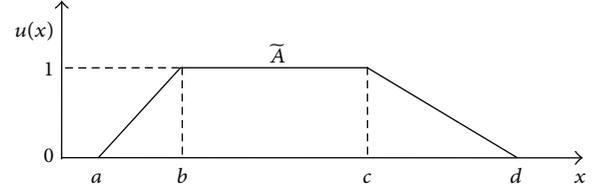


FIGURE 1: Activity duration represented by trapezoidal fuzzy number.

and the membership function u can be expressed as

$$u(x) = \begin{cases} \frac{x-a}{b-a}, & a \leq x < b, \\ 1, & b \leq x < c, \\ \frac{d-x}{d-c}, & c \leq x < d, \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

A fuzzy activity duration represented by four real parameters $a, b, c,$ and d can be written as $\tilde{A} = (a, b, c, d)$ and denoted by a trapezoidal (a, b, c, d) (see Figure 1).

If two fuzzy activity durations are defined as $\tilde{A} = (a_1, b_1, c_1, d_1)$ and $\tilde{B} = (a_2, b_2, c_2, d_2)$, respectively, then the following operations can be expressed as

Addition:

$$\tilde{A} \oplus \tilde{B} = (a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2). \quad (2)$$

Subtraction:

$$\tilde{A} \ominus \tilde{B} = (a_1 - d_2, b_1 - c_2, c_1 - b_2, d_1 - a_2). \quad (3)$$

Maximum:

$$\begin{aligned} & \max \{\tilde{A}, \tilde{B}\} \\ &= (\max \{a_1, a_2\}, \max \{b_1, b_2\}, \max \{c_1, c_2\}, \max \{d_1, d_2\}). \end{aligned} \quad (4)$$

Minimum:

$$\begin{aligned} & \min \{\tilde{A}, \tilde{B}\} \\ &= (\min \{a_1, a_2\}, \min \{b_1, b_2\}, \min \{c_1, c_2\}, \min \{d_1, d_2\}). \end{aligned} \quad (5)$$

Ma et al. [19] proposed another definition called fuzzy number in the parametric form.

Definition 2 (see [19]). A fuzzy activity duration \tilde{A} is a pair (\underline{u}, \bar{u}) of functions $(\underline{u}(r), \bar{u}(r)), 0 \leq r \leq 1$ which satisfy the following requirements:

- (1) $\underline{u}(r)$ is a bounded monotonic increasing left continuous function;

- (2) $\bar{u}(r)$ is a bounded monotonic decreasing left continuous function;
- (3) $\underline{u}(r) \leq \bar{u}(r)$, $0 \leq r \leq 1$.

The fuzzy trapezoidal number can be represented as $\tilde{A} = (x_0, y_0, \sigma, \beta)$, with left fuzziness $\sigma > 0$ and right fuzziness $\beta > 0$, and the membership function is presented as follows:

$$u(x) = \begin{cases} \frac{x - x_0 + \sigma}{\sigma}, & x_0 - \sigma \leq x \leq x_0, \\ 1, & x_0 \leq x \leq y_0, \\ \frac{y_0 + \beta - x}{\beta}, & y_0 \leq x \leq y_0 + \beta, \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

and its parametric form is described as follows:

$$\begin{aligned} \underline{u}(r) &= x_0 - \sigma + \sigma r, \\ \bar{u}(r) &= y_0 + \beta - \beta r, \end{aligned} \quad (7)$$

and if $x_0 = y_0$ then \tilde{A} is a triangular fuzzy number, denoted by $\tilde{A} = (x_0, \sigma, \beta)$.

In this paper, Definition 1 is used to express fuzzy activity duration, while in scheduling procedure the fuzzy activity duration is transformed into parametric form by Definition 2 in order to compare the fuzzy times.

2.2. Comparison of Fuzzy Times. While solving product development project scheduling problems, it is required to compare fuzzy times to generate a feasible solution. For example, the start time of an activity must be greater than or equal to the maximum finish time of its predecessors. Many methods have been developed to rank fuzzy numbers. For example, Asady and Zendehnam [20] developed a method based on distance minimization. Abbasbandy and Hajjari [21] proposed a method for ranking the trapezoidal fuzzy numbers that can overcome some drawbacks of distance minimization, while there still exist shortcomings in their methods. In this paper an improved magnitude of the trapezoidal fuzzy number proposed by Ezzati et al. [22] is adopted, which is easy to handle and will overcome the above drawbacks.

Definition 3 (see [22]). For a trapezoidal fuzzy time $\tilde{t}_j = (x_0, y_0, \sigma, \beta)$ with parametric form $\tilde{t}_j = (\underline{u}(r), \bar{u}(r))$, we define

$$\begin{aligned} \text{Mag}(\tilde{t}_j) &= \frac{1}{2} \left(\int_0^1 (\underline{u}(r) + \bar{u}(r) + x_0 + y_0) f(r) dr \right), \\ \text{Mag}'(\tilde{t}_j) &= \frac{1}{2} \left(\int_0^1 (\underline{u}'(r) - \bar{u}'(r) - \underline{u}(1) + \bar{u}(1)) dr \right), \end{aligned} \quad (8)$$

where the weight function $f(r)$ is nonnegative and increasing on $(0, 1)$ with $f(0) = 0$, $f(1) = 1$, and $\int_0^1 f(r) dr = 1/2$. Function $f(r)$ can be chosen according to the actual situation. In this paper we use $f(r) = r$.

For any two trapezoidal fuzzy times \tilde{t}_i and \tilde{t}_j , the improved magnitude of the trapezoidal fuzzy time \tilde{t}_j compared with \tilde{t}_i is defined as

$$R_{ji}(\tilde{t}_j, \delta_{ji}) = \text{Mag}(\tilde{t}_j) + \delta_{ji} \text{Mag}'(\tilde{t}_j), \quad (9)$$

where

$$\delta_{ji} = \begin{cases} 0, & \text{Mag}(\tilde{t}_i) \neq \text{Mag}(\tilde{t}_j) \\ 1, & \text{Mag}(\tilde{t}_i) = \text{Mag}(\tilde{t}_j). \end{cases} \quad (10)$$

And the improved magnitude of the trapezoidal fuzzy time \tilde{t}_i compared with \tilde{t}_j is defined as $R_{ij}(\tilde{t}_i, \delta_{ij}) = \text{Mag}(\tilde{t}_i) + \delta_{ij} \text{Mag}'(\tilde{t}_i)$. It is obvious that $\delta_{ji} = \delta_{ij}$.

The ranking of these two fuzzy numbers can be defined as

- (1) $R_{ij}(\tilde{t}_i, \delta_{ij}) > R_{ji}(\tilde{t}_j, \delta_{ji})$, if and only if $\tilde{t}_i > \tilde{t}_j$;
- (2) $R_{ij}(\tilde{t}_i, \delta_{ij}) < R_{ji}(\tilde{t}_j, \delta_{ji})$, if and only if $\tilde{t}_i < \tilde{t}_j$;
- (3) $R_{ij}(\tilde{t}_i, \delta_{ij}) = R_{ji}(\tilde{t}_j, \delta_{ji})$, if and only if $\tilde{t}_i \sim \tilde{t}_j$.

Then we formulate: $\tilde{t}_i \geq \tilde{t}_j$, if and only if $\tilde{t}_i > \tilde{t}_j$ or $\tilde{t}_i \sim \tilde{t}_j$;
 $\tilde{t}_i \leq \tilde{t}_j$, if and only if $\tilde{t}_i < \tilde{t}_j$ or $\tilde{t}_i \sim \tilde{t}_j$.

2.3. Problem Formulation. The product development project scheduling problem can be described as follows. A project can be split into n activities. Because of the technological constraints, there are precedence constraints in activities. Different kinds of resources are required when performing activities. The resources are limited, so the project is resource constrained. We assume in this paper that activities in product development project are single modes; activity durations are represented by trapezoidal fuzzy numbers; preemption is not allowed; only renewable resources are considered. The objective is to minimize the project completion time (makespan). Notation section gives some notations which are used in this paper.

The product development project scheduling problem with fuzzy activity duration can be described as follows:

$$\min \quad \bar{C} \quad (11)$$

$$\text{s.t.} \quad \bar{s}_1 = \bar{b} \quad (12)$$

$$\bar{c}_j = \bar{s}_j \oplus \bar{t}_j, \quad \forall j, j = 1, 2, \dots, n \quad (13)$$

$$\bar{s}_j \geq \max_{h \in P_j} \bar{c}_h, \quad \forall j, j = 2, 3, \dots, n \quad (14)$$

$$\sum_{j \in A_k} r_{jk} \leq R_k, \quad \forall k, k = 1, 2, \dots, K \quad (15)$$

$$\bar{C} = \bar{s}_n. \quad (16)$$

In the formulation, the objective function (11) minimizes the project fuzzy makespan. Equation (12) shows that the first activity must start at the project fuzzy ready time. Equation (13) calculates the fuzzy finish time for each activity j .

TABLE 1: An example of priority value representation.

Activity number k	1	2	3	4	5	6
Priority value \mathbf{X}_i	0.1	0.5	0.4	0.2	0.3	0.7

Constraint (14) represents that activity must follow the precedence relationships; that is, the fuzzy start time of activity j should be greater than or equal to the fuzzy finish times of all its predecessors. Constraint (15) forces the total units of resource utilized to be no greater than the available resource capacity for every time period. Equation (16) represents that the fuzzy completion time (makespan) of product development project is the fuzzy start time of the dummy end activity.

3. Fuzzy Firefly Algorithm for Product Development Project Scheduling with Fuzzy Activity Duration

3.1. Coding. In the paper a firefly algorithm is adapted to solve the project scheduling problem. The classic firefly algorithm was developed by Yang [8–10], and it was based on the idealized behavior of the flashing characteristics of fireflies. The flashing characteristics can be idealized as follows.

- (1) All fireflies are unisex so that one firefly is attracted to other fireflies regardless of their sex.
- (2) Attractiveness is proportional to their brightness; thus for any two flashing fireflies, the less bright one will move towards the brighter one. The attractiveness and brightness of two fireflies decrease as distance between them increases. If no one is brighter than a particular firefly, it moves randomly.
- (3) The brightness or light intensity of a firefly is affected or determined by the landscape of the objective function to be optimized.

Let $\mathbf{X} = (x_{ik})_{\text{pop} \times n} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_{\text{pop}})^T$ denote pop fireflies in the searching space. Vector \mathbf{X}_i denotes the location of firefly i ($i = 1, 2, \dots, \text{pop}$) in the population, and \mathbf{X}_i can be translated to a possible solution of the scheduling problem. In order to provide an activity sequence satisfying precedence relationships, priority value based real number coding method is employed to represent the firefly location. A firefly location \mathbf{X}_i is a priority value list with n entries written as $\mathbf{X}_i = (x_{i1}, x_{i2}, \dots, x_{in})$, where x_{ik} is the priority of activity k ($k = 1, 2, \dots, n$). An example of priority value representation is shown in Table 1. Consider a project with six activities numbered from 1 to 6; the number in the position j of the list represents the priority of activity j . For example, the priority of activity 4 is 0.2. When decoding, the activity with greater priority value will be considered first.

3.2. Decoding. Many researchers have concluded that the parallel schedule generation scheme gives generally better results than the serial schedule generation scheme [4]. So in this paper parallel schedule generation scheme is adopted to

generate feasible schedule from priority value list. For each firefly at location \mathbf{X}_i , the fuzzy parallel schedule generation scheme is described as follows.

Step 1 (initialization). The current fuzzy moment \tilde{t} is set to \tilde{b} , the set SA of activities scheduled by time \tilde{t} is set to ϕ , and the temporal set \tilde{T} that stores fuzzy time is set to ϕ .

Step 2. Compose a set $E(\tilde{t})$ of activities which have not been scheduled yet and whose immediate predecessors have been completed by time \tilde{t} .

Step 3. Find activity e who has the greatest priority value in the priority list \mathbf{X}_i , and for each time period from \tilde{t} to $\tilde{t} \oplus \tilde{t}_e$,

if the available renewable resources satisfy the requirements of activity e ,

activity e 's fuzzy start time: $\tilde{s}_e = \tilde{t}$,

activity e 's fuzzy finish time: $\tilde{c}_e = \tilde{t} \oplus \tilde{t}_e$;

allocate required resources to activity e , update resource availabilities;

insert \tilde{c}_e into set \tilde{T} ;

remove e from $E(\tilde{t})$;

insert e into SA;

else, if resource availabilities cannot satisfy the requirements of e , remove e from $E(\tilde{t})$.

Step 4. If $E(\tilde{t}) \neq \phi$, go to Step 3; else $\tilde{t} = \min(\tilde{T})$, remove \tilde{t} from \tilde{T} , and go to Step 5.

Step 5. If the number of activities in set SA $< n$, go to Step 2; else a feasible schedule is obtained and fuzzy completion time for the project $f(\mathbf{X}_i) = \tilde{C}$, stop.

Note that because the finish times of activities in \tilde{T} are all fuzzy numbers, the fuzzy ranking method called improved magnitude of the trapezoidal fuzzy number described in Section 2.2 is used to select the smallest activity finish time in Step 4.

3.3. Brightness and Attractiveness. There are two important issues in the firefly algorithm about brightness and attractiveness. The attractiveness of a firefly is determined by its brightness. As for project scheduling, the objective is to minimize the project completion time. For a firefly at a particular location \mathbf{X}_i , if the objective function $f(\mathbf{X}_i)$ for the project decoded by fuzzy parallel schedule generation scheme described in Section 3.2 is greater than others, then the firefly is less bright. The brightness $I(\mathbf{X}_i)$ of a firefly at a particular location \mathbf{X}_i can be chosen as $I(\mathbf{X}_i) \propto 1/f(\mathbf{X}_i)$. Because the completion time is a fuzzy number, the magnitude of trapezoidal fuzzy number is used to represent the light intensity of firefly i compared with firefly j ; that is

$$I_{ij}(\mathbf{X}_i) = \frac{1}{R_{ij}(f(\mathbf{X}_i), \delta_{ij})}. \quad (17)$$

The attractiveness varies depending on the distance d_{ij} between firefly i and firefly j . Light intensity decreases when the distance increases from its source, and light is also absorbed in the media. As a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, attractiveness function $\beta(d_{ij})$ of a firefly can be a monotonically decreasing function presented as

$$\beta(d_{ij}) = \beta_0 e^{-\gamma d_{ij}}, \quad (18)$$

where d_{ij} is the distance between firefly i and firefly j , γ is the light absorption coefficient, and β_0 is the attractiveness at $d_{ij} = 0$. We take $\beta_0 = 1$, $\gamma = 1$.

3.4. Distance and Movement. Two fireflies i and j at \mathbf{X}_i and \mathbf{X}_j can be decoded to two schedules with two fuzzy project completion times. The distance between these two fireflies cannot be represented directly by the Cartesian distance. We use the difference between the improved magnitudes of these two fuzzy completion times as the distance d_{ij} ; that is

$$d_{ij} = \|\mathbf{X}_i - \mathbf{X}_j\| = |R_{ij}(f(\mathbf{X}_i), \delta_{ij}) - R_{ji}(f(\mathbf{X}_j), \delta_{ji})|. \quad (19)$$

If $I_{ij}(\mathbf{X}_i) < I_{ji}(\mathbf{X}_j)$, firefly i will be attracted by j , the movement is determined by

$$\mathbf{X}'_i = \mathbf{X}_i + \beta_0 e^{-\gamma d_{ij}} (\mathbf{X}_j - \mathbf{X}_i) + \lambda \left(\text{rand} - \frac{1}{2} \right), \quad (20)$$

where vector $\mathbf{X}'_i = (x'_{i1}, x'_{i2}, \dots, x'_{in})$ is the new location after moving, the second term is determined by the attractiveness, the third term is randomization with the randomization parameter λ and a random number rand drawn uniformly from $(0, 1)$.

If firefly i at \mathbf{X}_i is the brightest in the population, then it will move randomly by

$$\mathbf{X}'_i = \mathbf{X}_i + \lambda \left(\text{rand} - \frac{1}{2} \right). \quad (21)$$

3.5. Algorithm Procedure. The procedure of fuzzy firefly algorithm for product development project with fuzzy activity duration is given in Figure 2, and described as follows.

Step 1 (initialization). Set $t = 1$; generate initial population with the population size pop and the location of firefly i is \mathbf{X}_i , $i = 1, 2, \dots, \text{pop}$.

Step 2 (decoding). For each firefly at location \mathbf{X}_i , use fuzzy parallel schedule generation scheme to generate a feasible schedule.

Step 3 (brightness computation). For each firefly at location \mathbf{X}_i , calculate $\text{Mag}(f(\mathbf{X}_i))$ and $\text{Mag}'(f(\mathbf{X}_i))$ by (8), which will be used in (17) to calculate the firefly's brightness compared with other fireflies.

Step 4 (choosing the optimal solution). For the brightest firefly in the t th generation, move randomly by (21).

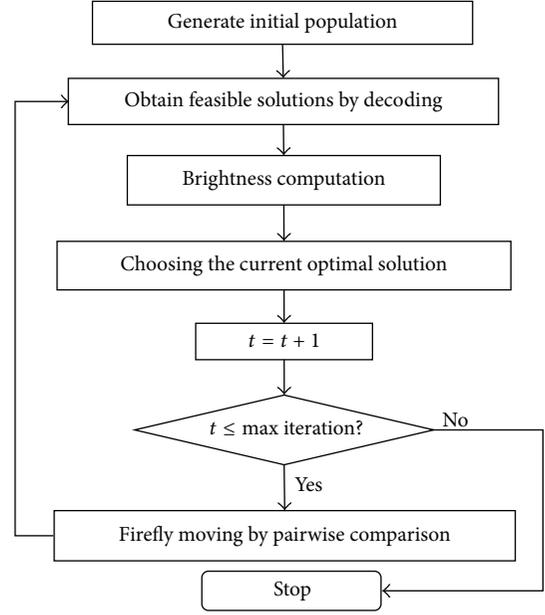


FIGURE 2: Procedure of fuzzy firefly algorithm.

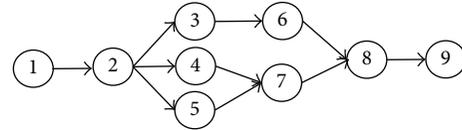


FIGURE 3: AON of an example.

TABLE 2: Activity parameters.

Act. number	Fuzzy duration	Resource required
1	(0, 0, 0, 0)	0
2	(4, 6, 9, 12)	1
3	(7, 9, 11, 14)	2
4	(6, 8, 10, 12)	4
5	(7, 8, 10, 11)	3
6	(2, 3, 5, 6)	1
7	(5, 7, 8, 10)	4
8	(9, 12, 14, 16)	3
9	(0, 0, 0, 0)	0

Select the brighter position from \mathbf{X}_i and \mathbf{X}'_i as the optimal solution of the present generation.

Step 5 (stopping criteria checking). Update $t = t + 1$.

If $t \leq \text{maxiteration}$, go to Step 6.

Else go to Step 7.

Step 6 (firefly moving). Every firefly is compared with all the other fireflies in the population except itself. That is, compare firefly i ($i = 1, 2, \dots, \text{pop}$) with firefly j ($\forall j = 1, 2, \dots, \text{pop}$, and $j \neq i$), if $I_{ij}(\mathbf{X}_i) < I_{ji}(\mathbf{X}_j)$, move firefly i towards firefly j by (20); update location of firefly i after each movement.

Go to Step 2.

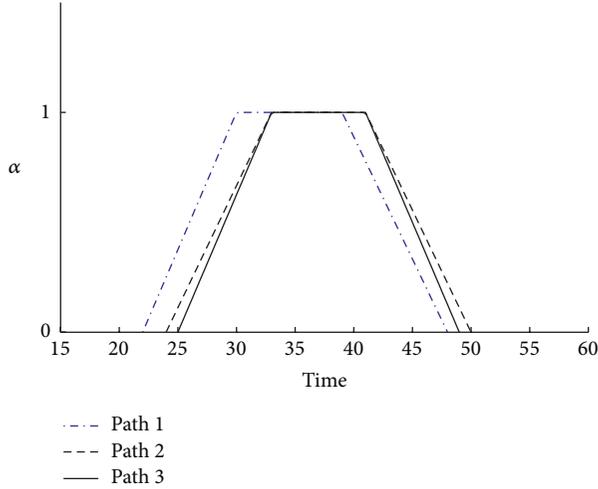


FIGURE 4: Path length.

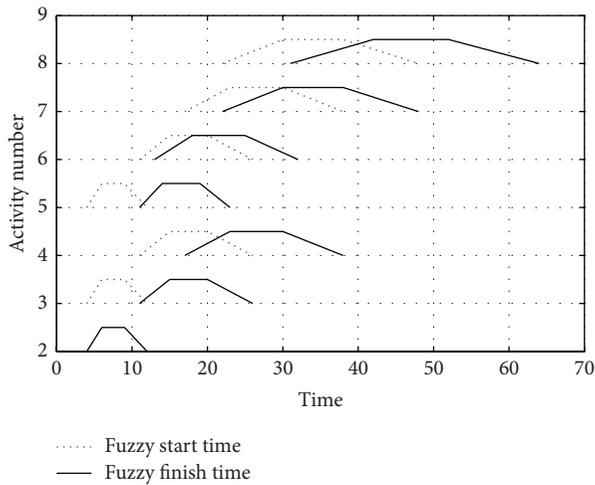


FIGURE 5: Fuzzy Gantt chart for the example project.

Step 7. The optimal solution is found; output the result and stop.

3.6. An Example. An example is used to illustrate the method developed in this paper. We consider a project which consists of nine activities and is represented by AON shown in Figure 3. The first and the last activities are dummy start and dummy end activities, respectively. Only one kind of renewable resource is considered in this example. The total availability of the resource is 5 units in every moment. The fuzzy project ready time is set to $\tilde{b} = (0, 0, 0, 0)$. The corresponding activity information is listed in Table 2.

There are three possible paths from the dummy start to the dummy end node. The paths and their lengths without resource constraints, respectively, are

- Path 1: 1-2-3-6-8-9, fuzzy path length: (22, 30, 39, 48);
- Path 2: 1-2-4-7-8-9, fuzzy path length: (24, 33, 41, 50);
- Path 3: 1-2-5-7-8-9, fuzzy path length: (25, 33, 41, 49).

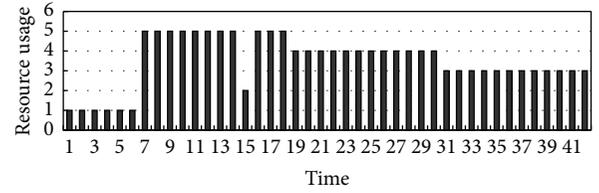
FIGURE 6: The resource usage profile at $\alpha = 1$ in the optimistic case.

TABLE 3: Activities' start times and finish times.

Activity number	Fuzzy start time	Fuzzy finish time
1	(0, 0, 0, 0)	(0, 0, 0, 0)
2	(0, 0, 0, 0)	(4, 6, 9, 12)
3	(4, 6, 9, 12)	(11, 15, 20, 26)
4	(11, 15, 20, 26)	(17, 23, 30, 38)
5	(4, 6, 9, 12)	(11, 14, 19, 23)
6	(11, 15, 20, 26)	(13, 18, 25, 32)
7	(17, 23, 30, 38)	(22, 30, 38, 48)
8	(22, 30, 38, 48)	(31, 42, 52, 64)
9	(31, 42, 52, 64)	(31, 42, 52, 64)

Figure 4 shows the lengths of the paths. It is clear that Path 1 is the shortest. By the fuzzy ranking method proposed by Abbasbandy and Hajjari [21], the lengths of Path 2 and Path 3 are equal. However, it is evident from Figure 4 that the result is not reasonable. We use the improved magnitude of the trapezoidal fuzzy number, the magnitude of the two paths are

the improved magnitude of path 2: $37 + 13 = 50$;

the improved magnitude to path 3: $37 + 12 = 49$.

As $50 > 49$, $(24, 33, 41, 50) \succ (25, 33, 41, 49)$, we know that path 2 is the critical path, and the project completion time without resource constraints is (24, 33, 41, 50).

When resource constraints are considered, the fuzzy firefly algorithm is used to solve this problem. The population of fireflies is $\text{pop} = 4$, set maxiteration = 2, the randomization parameter in (20) and (21) is set to $\lambda = 0.1$. The fuzzy start times and finish times of each activity are listed in Table 3. It shows the project completion time is (31, 42, 52, 64). Figure 5 presents the fuzzy Gantt chart without dummy start and end activities. The resource usage profile for all lower bounds (optimistic case) of α -level activity durations at $\alpha = 1$ is calculated and presented in Figure 6.

4. Computational Experiments

Two experiments are conducted. In the first experiment, we use the proposed method to solve benchmark problems and discuss the results. In the second experiment an electronic product development project scheduling problem proposed by Wang [1] is studied. The algorithm is coded in MATLAB7.8.

TABLE 4: Results for numerical experiments.

Number	Optimal solution or lower bound	Critical path lower bound \tilde{c}_{cplb}	Fuzzy makespan \tilde{c}_t	Difference dev.
1	43	(27, 35, 38, 50)	(32, 39, 43, 55)	4.5833
2	47	(31, 38, 42, 52)	(35, 45, 50, 61)	7.3333
3	47	(30, 39, 43, 52)	(35, 46, 51, 62)	7.5000
4	62	(39, 46, 55, 67)	(47, 54, 62, 74)	7.5000
5	39	(22, 26, 31, 43)	(28, 36, 40, 54)	9.3333
6	48	(26, 35, 38, 52)	(33, 42, 49, 61)	8.8333
7	60	(44, 53, 60, 76)	(48, 57, 64, 81)	4.0833
8	53	(38, 47, 53, 68)	(43, 51, 57, 73)	4.1667
9	49	(29, 38, 42, 51)	(35, 45, 51, 63)	8.1667
10	45	(27, 34, 37, 45)	(34, 42, 45, 61)	8.5833
11	38	(26, 32, 34, 44)	(28, 35, 38, 49)	3.5000
12	51	(32, 39, 46, 55)	(37, 44, 51, 62)	5.1667
13	43	(29, 41, 43, 54)	(29, 41, 43, 54)	0
14	43	(27, 38, 42, 52)	(28, 39, 43, 54)	1.0833
15	51	(35, 47, 51, 63)	(35, 47, 51, 63)	0
16	47	(37, 46, 47, 59)	(37, 46, 47, 59)	0
17	47	(31, 38, 47, 57)	(31, 38, 47, 57)	0
18	54	(36, 46, 50, 61)	(41, 49, 55, 70)	4.5000
19	54	(38, 46, 51, 65)	(49, 57, 63, 79)	11.6667
20	43	(30, 36, 41, 52)	(29, 37, 44, 59)	2.1667
21	77	(60, 70, 77, 94)	(62, 74, 83, 100)	4.8333
22	68	(45, 58, 65, 79)	(49, 65, 73, 89)	7.4167
23	68	(44, 60, 67, 85)	(53, 66, 74, 90)	6.5833
24	91	(56, 75, 79, 97)	(67, 85, 93, 115)	12.4167
25	73	(47, 58, 68, 86)	(58, 73, 78, 101)	12.5833
26	66	(32, 46, 52, 67)	(47, 61, 66, 85)	14.8333
27	72	(44, 51, 60, 75)	(55, 69, 77, 97)	17.3333
28	75	(51, 62, 71, 89)	(59, 75, 80, 99)	10.6667
29	85	(56, 65, 75, 89)	(63, 79, 92, 111)	15.3333
30	80	(52, 67, 76, 98)	(59, 75, 84, 104)	7.7500

4.1. Numerical Experiments and Results. In order to test the proposed fuzzy firefly algorithm, we have taken 30 datasets from PSPLIB [23] and generated instances with fuzzy activity duration. The 10 instances are chosen from benchmark *j301*, 10 from *j302*, and 10 from *j601*, respectively. As the activity duration of benchmarks in PSPLIB is deterministic, we use the following method to generate fuzzy activity duration. The deterministic duration can be denoted by t_j , the fuzzy activity duration is denoted by $\tilde{t}_j = (a_j, b_j, c_j, d_j)$, the method to generate fuzzy activity times is as follows:

$$c_j = t_j, \quad (22)$$

$$a_j = \text{floor}(\text{rand}(t_j * (1 - 40\%), t_j)), \quad (23)$$

$$b_j = \text{ceil}(\text{rand}(t_j * (1 - 40\%), t_j)), \quad (24)$$

$$d_j = \text{ceil}(\text{rand}(t_j, t_j * (1 + 30\%))). \quad (25)$$

Equation (22) sets $c_j = t_j$, that is the pessimistic case of α -level activity duration when $\alpha = 1$. Equations (23) and (24) show that a_j and b_j are randomly generated from the interval

$(t_j * (1 - 40\%), t_j)$. The floor function is used to generate an integer for a_j , and the lower value is bounded by 1. The ceiling function is used to generate an integer for b_j , and $b_j > a_j$. d_j is randomly generated from the interval $(t_j, t_j * (1 + 30\%))$ and the ceiling function is used to generate an integer for the pessimistic estimation as shown in (25).

Though the optimal solutions for problem datasets *j301* and *j302* are known and the lower bounds of *j601* problems are also given in PSPLIB, they are solutions for problems with deterministic activity duration. We calculate the critical path lower bounds for problems with fuzzy activity duration without considering resource constraints, denoted by \tilde{c}_{cplb} . The solution obtained by our proposed method is compared with \tilde{c}_{cplb} , the difference from lower bound is denoted by dev and computed as follows:

$$\text{dev} = R(\tilde{c}_n, \delta) - R(\tilde{c}_{\text{cplb}}, \delta), \quad (26)$$

where $R(\tilde{c}_n, \delta)$ and $R(\tilde{c}_{\text{cplb}}, \delta)$ are the improved magnitudes of the project makespan and the critical path lower bound, respectively.

TABLE 5: Project data for an electronic product development project [1].

Act. number	Fuzzy duration \tilde{t}_j	Resource required				Successors S_j
		r_{j1}	r_{j2}	r_{j3}	r_{j4}	
1	(0, 0, 0, 0)	0	0	0	0	2
2	(5, 6, 6, 7)	3	0	0	0	3
3	(10, 12, 12, 13)	5	3	3	2	4, 5
4	(4, 5, 5, 6)	4	2	4	2	6
5	(7, 8, 8, 10)	4	2	2	2	6
6	(2, 4, 4, 6)	5	0	0	0	7
7	(3, 4, 4, 5)	4	2	2	2	8
8	(2, 3, 3, 4)	4	3	3	2	9, 10, 11, 13
9	(2, 3, 3, 5)	0	4	2	0	14
10	(3, 4, 4, 6)	2	1	5	3	15
11	(4, 5, 5, 6)	4	0	0	0	12
12	(7, 9, 9, 10)	5	0	0	0	16
13	(3, 4, 4, 5)	3	0	0	1	16
14	(1, 2, 2, 3)	3	2	0	1	16
15	(1, 2, 2, 3)	3	0	3	2	16
16	(1, 2, 2, 2)	4	2	2	2	17
17	(3, 4, 4, 5)	5	2	2	0	18, 19
18	(4, 5, 5, 6)	5	2	4	2	20
19	(4, 5, 5, 6)	4	2	4	3	20
20	(5, 6, 6, 7)	5	2	3	4	21, 22, 23
21	(5, 7, 7, 9)	4	0	0	0	24
22	(3, 5, 5, 7)	5	1	4	4	24
23	(4, 5, 5, 6)	0	3	0	0	24
24	(1, 2, 2, 3)	5	1	1	1	25
25	(8, 10, 10, 12)	2	1	5	3	26, 27, 28
26	(4, 5, 5, 6)	2	1	5	3	30
27	(4, 6, 6, 8)	0	0	4	3	29
28	(6, 7, 7, 8)	0	4	0	2	30
29	(10, 12, 12, 14)	2	2	4	4	30
30	(1, 2, 2, 4)	4	2	2	2	31, 32, 33, 39, 40, 42
31	(9, 11, 11, 10)	4	2	3	0	34
32	(10, 13, 13, 14)	5	0	3	2	37
33	(8, 10, 10, 11)	0	0	4	2	38
34	(6, 8, 8, 9)	4	3	2	0	35
35	(3, 4, 4, 5)	4	2	3	3	36
36	(4, 6, 6, 7)	2	2	0	4	41
37	(1, 3, 3, 4)	0	0	2	0	41
38	(3, 4, 4, 5)	0	0	2	2	41
39	(1, 2, 2, 3)	1	0	3	1	41
40	(2, 3, 3, 4)	0	0	2	2	41
41	(1, 1, 1, 2)	4	0	4	3	45, 46, 49
42	(3, 4, 4, 5)	0	4	0	0	43
43	(1, 2, 2, 3)	3	3	0	0	44
44	(1, 2, 2, 3)	3	0	3	0	45, 46
45	(3, 4, 4, 5)	2	3	4	1	52

TABLE 5: Continued.

Act. number	Fuzzy duration \tilde{t}_j	Resource required				Successors S_j
		r_{j1}	r_{j2}	r_{j3}	r_{j4}	
46	(4, 5, 5, 6)	2	2	4	2	47
47	(4, 5, 5, 6)	5	2	2	3	48
48	(3, 4, 4, 5)	2	0	4	2	52
49	(5, 7, 7, 8)	2	0	2	3	50
50	(6, 8, 8, 10)	4	2	3	2	51
51	(2, 3, 3, 5)	4	2	3	3	52
52	(2, 4, 4, 7)	4	2	3	3	53
53	(0, 0, 0, 0)	0	0	0	0	—

In this paper the parameters for performing fuzzy firefly algorithm were set as follows: population size $pop = 10$, maximum iteration $maxiteration = 10$, and the randomization parameter $\lambda = 0.1$. The computational results are shown in Table 4.

The numbers in column two show the optimal solutions for $j30$ problems and lower bounds for $j60$ problems with deterministic activity duration. The third column represents the critical path lower bounds for the problems with trapezoidal fuzzy activity duration. The fourth column represents the best solutions obtained by the proposed fuzzy firefly algorithm. It shows that the third number of the best solution for eleven problems (1, 4, 10, 11, 12, 13, 14, 15, 16, 17, 26) is the same as the optimal solution. For problems 7, 8, 13, 15, 16, 17, and 21, the third number of the critical path lower bounds is the same as optimal solution, which means the resource constraints for these problems are not binding [5]. Our method gains the optimal solutions for problems 13, 15, 16, 17. The fifth column represents the deviations of the best solution from the critical path lower bound. The deviations of the 4 problems discussed above are zeros. The deviations for other problems vary between 1.0833 and 17.3333. It also indicates that the deviation is greater when there are more activities in a project, because the difficulty of solving $j60$ is greater than that of $j30$.

4.2. *The Experiment on Electronic Product Development Project.* Wang [1] provides an electronic product development project, which consists of 7 phases and is divided into 51 activities. We add dummy start and end activities to the project, and the information of this project is shown in Table 5. The project ready time is $\tilde{b} = (0, 0, 0, 0)$. There are 4 types of resources: system engineers, software engineers, hardware engineers, and supporting engineers. The resource availability of each type of resource is 5, 4, 5, 4, respectively.

The proposed fuzzy firefly algorithm is used to solve the electronic product development project. The project makespan obtained by the proposed method is (178, 235, 235, 288), and the fuzzy finish times for the solutions are shown in Table 6. Figure 7 shows the resource usage for every type of resource. The utilization rates for system engineers, software engineers, hardware engineers, and supporting engineers are 69.4%, 41.1%, 56.3%, and 51.7%, respectively. Wang's results

TABLE 6: Solutions for the electronic product development project.

Act. number	Fuzzy finish time \tilde{c}_j
1	(0, 0, 0, 0)
2	(5, 6, 6, 7)
3	(15, 18, 18, 20)
4	(19, 23, 23, 26)
5	(26, 31, 31, 36)
6	(28, 35, 35, 42)
7	(31, 39, 39, 47)
8	(33, 42, 42, 51)
9	(35, 45, 45, 56)
10	(40, 51, 51, 63)
11	(37, 47, 47, 57)
12	(49, 64, 64, 78)
13	(40, 51, 51, 62)
14	(41, 53, 53, 65)
15	(42, 55, 55, 68)
16	(50, 66, 66, 80)
17	(53, 70, 70, 85)
18	(57, 75, 75, 91)
19	(61, 80, 80, 97)
20	(66, 86, 86, 104)
21	(71, 93, 93, 113)
22	(74, 98, 98, 120)
23	(70, 91, 91, 110)
24	(75, 100, 100, 123)
25	(83, 110, 110, 135)
26	(101, 133, 133, 163)
27	(87, 116, 116, 143)
28	(107, 140, 140, 171)
29	(97, 128, 128, 157)
30	(108, 142, 142, 175)
31	(135, 176, 176, 213)
32	(126, 165, 165, 200)
33	(116, 152, 152, 186)
34	(141, 184, 184, 222)
35	(144, 188, 188, 227)
36	(148, 194, 194, 234)
37	(127, 168, 168, 204)
38	(121, 159, 159, 195)
39	(136, 178, 178, 216)
40	(118, 155, 155, 190)
41	(149, 195, 195, 236)
42	(111, 146, 146, 180)
43	(112, 148, 148, 183)
44	(145, 190, 190, 230)
45	(152, 199, 199, 241)
46	(169, 222, 222, 270)
47	(173, 227, 227, 276)
48	(176, 231, 231, 281)
49	(157, 206, 206, 249)
50	(163, 214, 214, 259)
51	(165, 217, 217, 264)

TABLE 6: Continued.

Act. number	Fuzzy finish time \tilde{c}_j
52	(178, 235, 235, 288)
53	(178, 235, 235, 288)

TABLE 7: Comparison of solutions under different resource availabilities.

	Resource availability	Fuzzy project makespan
Our result	5 4 5 4	(178, 235, 235, 288)
Wang [1]	5 4 5 4	(180, 239, 239, 294)
Wang [1]	6 4 5 4	(178, 235, 235, 288)
Wang [1]	6 5 5 4	(178, 235, 235, 288)
Wang [1]	5 4 6 5	(178, 235, 235, 288)

show that the project makespan is (180, 239, 239, 294), and the resource utilization rates are 0.69, 0.33, 0.56, and 0.51 [1]. The comparison shows that our method can improve the resource utilization rate of software engineers and finish the project earlier. When considering the allocation of more resources to help complete the project earlier, Wang [1] provides 12 schedules under distinct resource availability. Table 7 shows that our results have the same effect with 3 of the schedules proposed by Wang, which illustrates that our method can allocate resource more effectively, and shorten the makespan without adding more resources to the project.

5. Conclusion

This paper concentrates on the problem of product development project scheduling with fuzzy activity duration. Trapezoidal fuzzy numbers are used to represent the uncertain time parameters; mathematical model for the problem is established, and the fuzzy firefly algorithm is proposed. For the proposed method the improved magnitude of the trapezoidal fuzzy number is adopted in 4 circumstances: deciding the fuzzy start times of activities, representing firefly brightness, calculating the distances between fireflies, and comparing optimal solutions. Computational experiment results indicate that the proposed fuzzy firefly algorithm is helpful for managers to effectively allocate resources with the objective of minimizing project makespan. The future research will extend the fuzzy firefly algorithm to handle multiobjective product development project scheduling problems in an uncertain environment.

Notations

- j : Activity number of the product development project, $j = 1, 2, \dots, n$, where the first activity and the last activity are the dummy start and dummy end activities, respectively
- S_j : The set of successors of activity j
- P_j : The set of predecessors of activity j
- \tilde{t}_j : Fuzzy duration of activity j represented by a trapezoidal fuzzy number

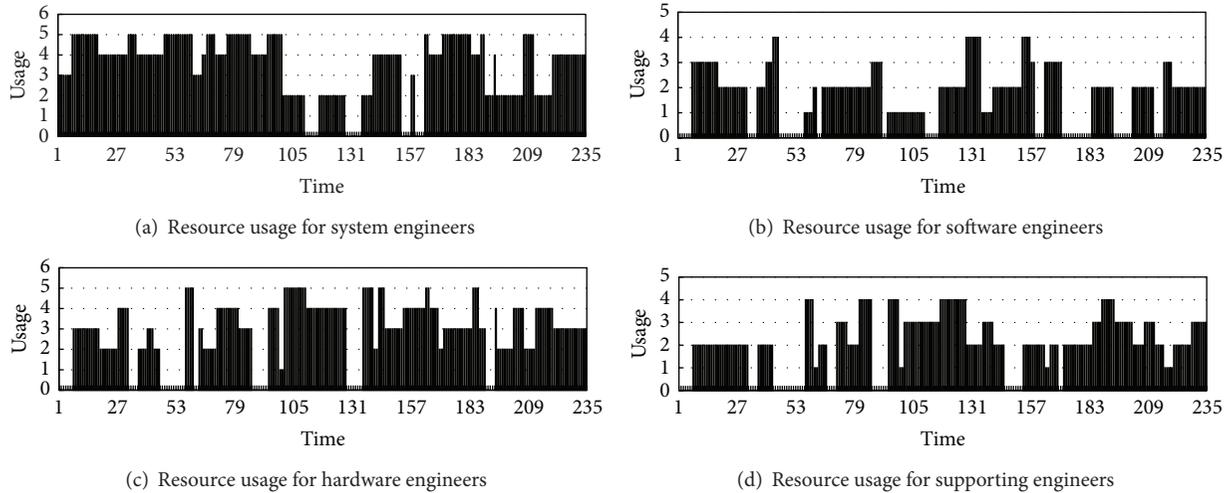


FIGURE 7: The resource usage profile for every resource type at $\alpha = 1$ in the optimistic case.

K : Types of renewable resources

R_k : The total number of available renewable resources of type k , $k = 1, 2, \dots, K$

r_{jk} : The number of renewable resources of type k required per unit time over the duration of the activity j , $j = 1, 2, \dots, n$, $k = 1, 2, \dots, K$

\tilde{b} : The fuzzy ready time of project

\tilde{s}_j : The fuzzy start time of activity j

\tilde{c}_j : The fuzzy finish time of activity j

$A_{\tilde{t}}$: The set of activities which are performed at current fuzzy time moment \tilde{t}

\tilde{C} : The fuzzy completion time of the product development project.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

Aids from the National Natural Science Foundation of China under Grant nos. 71102146, 71272084, and 61202296, and the Humanities and Social Science Foundation of Ministry of Education of the People's Republic of China under Grant no. 12YJC630278 are gratefully acknowledged. The authors thank the reviewers for their helpful advices to improve this paper.

References

- [1] J. Wang, "A fuzzy project scheduling approach to minimize schedule risk for product development," *Fuzzy Sets and Systems*, vol. 127, no. 2, pp. 99–116, 2002.
- [2] S. H. Nasution, "Fuzzy critical path method," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 1, pp. 48–57, 1994.
- [3] F. A. Lootsma, "Stochastic and fuzzy Pert," *European Journal of Operational Research*, vol. 43, no. 2, pp. 174–183, 1989.
- [4] M. Hapke and R. Slowinski, "Fuzzy priority heuristics for project scheduling," *Fuzzy Sets and Systems*, vol. 83, no. 3, pp. 291–299, 1996.
- [5] T. Bhaskar, M. N. Pal, and A. K. Pal, "A heuristic method for RCPSP with fuzzy activity times," *European Journal of Operational Research*, vol. 208, no. 1, pp. 57–66, 2011.
- [6] J. Wang, "A fuzzy robust scheduling approach for product development projects," *European Journal of Operational Research*, vol. 152, no. 1, pp. 180–194, 2004.
- [7] H. Ke and B. Liu, "Fuzzy project scheduling problem and its hybrid intelligent algorithm," *Applied Mathematical Modelling*, vol. 34, no. 2, pp. 301–308, 2010.
- [8] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, Frome, UK, 2008.
- [9] X. S. Yang, "Firefly algorithm for multimodal optimization," in *Stochastic Algorithms: Foundations and Applications*, vol. 5792 of *Lecture Notes in Computer Science*, pp. 169–178, Springer, Berlin, Germany, 2009.
- [10] X.-S. Yang, "Firefly algorithm, stochastic test functions and design optimization," *International Journal of Bio-Inspired Computation*, vol. 2, no. 2, pp. 78–84, 2010.
- [11] S. Lukasik and S. Zak, "Firefly algorithm for continuous constrained optimization tasks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5796, pp. 97–106, 2009.
- [12] M. K. Sayadi, R. Ramezani, and N. Ghaffari-Nasab, "A discrete firefly meta-heuristic with local search for makespan minimization in permutation flow shop scheduling problems," *International Journal of Industrial Engineering Computations*, vol. 1, no. 1, pp. 1–10, 2010.
- [13] P. Sanaei, R. Akbari, V. Zeighami, and S. Shams, "Using firefly algorithm to solve resource constrained project scheduling problem," *Advances in Intelligent Systems and Computing*, vol. 201, no. 1, pp. 417–428, 2013.
- [14] M. M. Huang, J. J. Yuan, and Y. Xue, "A hybrid firefly algorithm for resource-constrained project scheduling problem," in *Proceedings of International Conference on Computer and Management (CAMMAN '12)*, Wuhan, China, March 2012.
- [15] J.-J. Yuan, "Optimal design for scale-based product family based on multi-objective firefly algorithm," *Computer Integrated*

- Manufacturing Systems*, vol. 18, no. 8, pp. 1801–1809, 2012 (Chinese).
- [16] X. S. Yang, “Multiobjective firefly algorithm for continuous optimization,” *Engineering with Computers*, vol. 29, no. 2, pp. 175–184, 2013.
- [17] F. Luna, D. L. González-Álvarez, F. Chicano, and M. A. Vega-Rodríguez, “The software project scheduling problem: a scalability analysis of multi-objective metaheuristics,” *Applied Soft Computing Journal*, vol. 15, pp. 136–148, 2014.
- [18] D. Dubois and H. Prade, “Towards fuzzy differential calculus part 3: differentiation,” *Fuzzy Sets and Systems*, vol. 8, no. 3, pp. 225–233, 1982.
- [19] M. Ma, M. Friedman, and A. Kandel, “A new fuzzy arithmetic,” *Fuzzy Sets and Systems*, vol. 108, no. 1, pp. 83–90, 1999.
- [20] B. Asady and A. Zendehnam, “Ranking fuzzy numbers by distance minimization,” *Applied Mathematical Modelling*, vol. 31, no. 11, pp. 2589–2598, 2007.
- [21] S. Abbasbandy and T. Hajjari, “A new approach for ranking of trapezoidal fuzzy numbers,” *Computers & Mathematics with Applications*, vol. 57, no. 3, pp. 413–419, 2009.
- [22] R. Ezzati, T. Allahviranloo, S. Khezerloo, and M. Khezerloo, “An approach for ranking of fuzzy numbers,” *Expert Systems with Applications*, vol. 39, no. 1, pp. 690–695, 2012.
- [23] R. Kolisch and A. Sprecher, “PSPLIB—a project scheduling problem library,” *European Journal of Operational Research*, vol. 96, no. 1, pp. 205–216, 1997.

Research Article

Taxiing Route Scheduling between Taxiway and Runway in Hub Airport

Yu Jiang, Xinxing Xu, Honghai Zhang, and Yuxiao Luo

College of Civil Aviation, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, China

Correspondence should be addressed to Yu Jiang; jiangyu07@nuaa.edu.cn

Received 5 June 2014; Accepted 16 September 2014

Academic Editor: Bin Yu

Copyright © 2015 Yu Jiang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To guarantee the operation safety of airport, improve the efficiency of surface operation, and enhance the fairness of taxiing route scheduling, an optimizing model is established for the airport surface taxiing route scheduling. Reducing the total aircraft taxiing route length and reducing the waiting delay time are the goals of the model by controlling the initial taxiing time of aircraft and choosing the right taxiing route. The model can guarantee the continuous taxiing for all aircraft without conflicts. The runway scheduling is taken into consideration in the model to optimize the surface operation. The improved genetic algorithm is designed for simulation and validation. The simulation results show that compared with the ant colony optimization method, the improved genetic algorithm reduces the total extra taxiing distance by 47.8% and the total waiting delay time decreases by 21.5%. The optimization model and improved genetic algorithm are feasible. The optimization of taxiing route method can provide decision support for hub airports.

1. Introduction

With the rapid development of air transport and the sharp increase in the number of aircraft, airports have increasingly become a “bottleneck” of the air transportation network. This phenomenon leads to low operation efficiency of resources in the airport increasing aircraft delays and air pollution and other problems [1]. At present, there are two main ways to solve the airport “bottleneck” problem at home and abroad: one is to increase the infrastructure construction of airport surface resources and the other is to improve the operation efficiency of airport surface resources. However, the airport infrastructure construction in China is still far behind the rapid development abroad. Therefore, how to improve the operation efficiency of airport surface resources becomes the main method to resolve the airport “bottleneck” problem.

The operation of airport surface resources includes runway scheduling for landings and take-offs, gate assignment, and taxiway routing. Connecting runways and gates, taxiways are the key resource of the airport taxiing process [2]. Thus how to improve the operation efficiency of the taxiway is

the key to improve the operation efficiency of airport surface resources.

Many scholars have studied aircraft surface taxiing and scheduling and achieved some good achievements. These achievements can be mainly divided into two aspects: one is the preset route of aircraft taxiing and the other is dynamic selection route of aircraft taxiing. Among the preset route of aircraft taxiing researches, Smeltink and Soomer [3] presented the first approach to solve the surface movement problem using the MILP formulation, but the research did not consider the taxiing interval and taxiing time constraints, so there was a certain security risk. Based on the research of Smeltink and Soomer, Rathinam et al. [4] added a taxiing interval constraint to the model, but each aircraft still moved with a predetermined route. Gotteland et al. [5] presented a taxiing model based on the characteristics of aircraft taxiing conflicts and used genetic algorithms to simulate the model, but the research did not consider the waiting problem of aircraft caused by the potential taxiing conflicts. Landry et al. [6] used the theory of complex network, dynamically detected and resolved the conflicts on taxiways

and runways, improved the operation efficiency of the surface resources, and ensured the safety of the aircraft taxiing, but the simulation of model was relatively complicated, which could not meet the requirement of real-time scheduling of the taxiing. Anderson et al. [7] proposed two simple queuing models to indicate the taxiing-in and taxiing-out processes of aircraft. The model could be applied for not only predicting the surface congestion, but also evaluating the control strategy which could improve the operation efficiency of airport. But the model had little consideration on the interrelation between landings and take-offs. Pitfield et al. [8] used the Monte Carlo simulation to study the potential taxiing conflicts on congested taxiways, but taxiing optimization had not been performed in the simulation. The study resolved the taxiing conflicts by controlling aircraft to wait in certain nodes. Therefore, the study always belonged to the predetermined route research.

The predetermined taxiing route of aircraft is likely to lead to taxiing delays and increase the operation cost. Therefore, some scholars have carried out some studies on dynamic taxiing route of aircraft. Keith et al. [9] formulated a MILP model combining the runway scheduling with taxiway routing in a continuous time environment. The simulation results were significantly better than the results obtained from the runway scheduling and taxiway route routing problem separately. But the simulation data used was relatively simple, and it could not operate in real simulation environment. Balakrishnan and Yoon [10] presented the aircraft dynamic taxiing model. The results could reduce the total taxiing time by controlling the taxiing-out time and taxiing path rerouting. But each aircraft could only be allocated one of the limited routes in the set. Roling and Visser [11] presented an alternative MILP model for ground movement on a space-based network, in which optional routes were assigned for each aircraft in advance. This research only ensured that there were no conflicts in the planning period. Anderson and Milutinović [12] considered aircraft taxiing traffic flows on the taxiway by adjusting the speed of taxiing aircraft on each taxiway segment. Meanwhile the model was employed to consider the flows of aircraft instead of node occupation on the time dimension. The model also effectively incorporated the aircraft taxiing uncertainty into it. You and Han [13] introduced multiple agent technology to model the taxiing route of aircraft. The research considered the contract net protocol theory in the model and selected Dijkstra algorithm to optimize the taxiing route. A multiple agent route optimization algorithm was adopted and the simulation analysis was made. But the research did not take the dynamic nature of aircraft taxiing into account. Wang et al. [14] presented a dynamic taxiing route allocation algorithm to avoid taxiing conflicts, but the uncertainty of the aircraft taxiing speed was not taken into account. Marin et al. [15, 16] abstracted the taxiway into multiple commodity flow problem. The model took the aircraft routing and the queue scheduling problem into account. It belonged to dynamic route assignment scope. However, the presented algorithm could not handle separation constraints in an accurate way. In the dynamic route assignment, scholars have presented dynamic assignment models from different viewpoints, but

the accuracy and operation speed of algorithm need to be improved. Carr et al. [17] and Idris et al. [18] considered the interaction influences of arrival and departure taxiing process and presented a route assignment model for dynamic taxiing. But the algorithm used in the solution reduced the speed and accuracy of the model. Baik et al. [19] used graph theory and designed a time-dependent network assignment strategy to optimize the aircraft dynamic taxiing routes. To obtain the satisfactory acceptable solution, García et al. [20] and Gotteland and Durand [21] introduced the heuristic algorithm to improve the operation efficiency of airport surface resources and achieved good achievements.

In summary, both in the preset and nonset taxiing routes, most researches formulated a single goal model for aircraft taxiing, without consideration of synergistic operations of other surface resources. Meanwhile, most of the researches only considered unilateral interests of airports, airlines, or air traffic controllers separately, without considering their whole interests comprehensively. In the aircraft taxiing, the occupancy and operation mode of runway have a great influence on the aircraft taxiing route. How to model and simulate the synergistic scheduling of runway and taxiways operation under the premise of considering the interests of all parties is an urgent problem in the airport surface resource scheduling.

The needs of air traffic controllers, airlines, and airports are considered in the paper. A synergistic scheduling strategy based on safety, efficiency, and fairness is presented in the airport surface movement. A synergistic scheduling model is established on runway and taxiway scheduling and an improved genetic algorithm is used to simulate the verification of model. It can provide theoretical guidance for resource synergistic scheduling for managers in hub airports.

The remainder of this paper is structured as follows. Section 2 provides a brief description of the aircraft taxiing scheduling problem and the aspects this paper mainly considered. Section 3 gives the optimizing model of the airport synergistic scheduling problem. In Section 4, the improved genetic algorithm method is introduced first. The simulation data from the airport is then presented. Following this, the results are then shown together with the discussion. Section 5 contains the conclusions.

2. Problem Description

Aircraft taxiing scheduling is a complex route scheduling problem including landing and take-off taxiing scheduling. Landing taxiing scheduling means assigning appropriate landing time for landings, selecting the appropriate runway exit to taxiway, and then selecting the appropriate taxiing route to reach the assigned gate. Take-off taxiing scheduling means assigning appropriate push-out time and selecting appropriate taxiing route so that the aircraft can begin to taxi from the gate to the right runway threshold. Therefore, both the operations of landing and take-off are successive. Airport taxiing scheduling for aircraft on airport surface is to determine aircraft approach time, departure time, and taxiing route for each aircraft under the premise of ensuring safety. Therefore, the total taxiing distance (time) of both landings

and take-offs is made to be the shortest and the total delays to be the least.

In the airport surface taxiing scheduling, the different stakeholders have different requirements. The air traffic control authorities require ensuring the safety of the movement of aircraft on the airport surface. Airlines need to ensure that flights can take off and land on time to reduce delay and taxiing cost. Airport authorities want to improve the utilization efficiency of airport resources. Therefore, the airport surface scheduling, not only security but also efficiency and fairness of scheduling, should be taken into account. The so-called synergistic scheduling is to consider three aspects of comprehensive requirements, as well as the synergistic scheduling of taxiway and runway.

3. Model

3.1. Assumptions. (1) Aircraft's speed is constant during taxiing and the taxiing route will be continuous without any stop until to the destination.

(2) All aircraft have the same taxiing route unit cost and waiting delay cost.

(3) Gate zone has sufficient gates to meet all the demand of landing and take-off aircraft in planned period of time.

(4) Take-off aircraft starts taxiing from the gate zone and ends in the runway threshold; landing aircraft start taxiing from the runway exit and end in the gate zone. Each aircraft has a fixed gate and exit on the runway.

3.2. Define Variables

F : set of all aircraft within planned period, $F = \{f_1, f_2, \dots, f_k\}$;

F^d : set of all take-offs within planned period;

F^a : set of all landings within planned period;

A : set of airlines, $A = \{\alpha_1, \alpha_2, \dots, \alpha_l\}$;

N : set of airport surface nodes, any node $n_p, n_q \in N$;

N_r : set of runway entrance and exit nodes, $N_r \subset N$;

L_{pq} : the distance between nodes n_p and n_q (unit: km);

V_i^{\max} : the maximum taxiing speed of aircraft f_i , and any speed $V_i \in (0, V_i^{\max}]$;

$C_{pq} = 1$, if there is a directly connected and usable route from node n_p to node n_q ; 0, otherwise;

T_{ip} : the time of aircraft f_i arrives at the node n_p ;

T_i^0 : the reference scheduling time, namely, the earliest time that aircraft could come into the taxiway system;

$R_{ipq} = 1$, if aircraft f_i taxis from node n_p to node n_q ; 0, otherwise;

$Z_{ijp} = 1$, if aircraft f_i reaches node n_p before aircraft f_j ; 0, otherwise;

N^i : the taxiing route of aircraft f_i , consisting of a series of nodes $N^i = (n_1^i, n_2^i, \dots, n_{k_i}^i)$;

t_{ij}^e : the safe taxiing time interval between aircraft f_i and aircraft f_j ;

t_{ij}^w : the wake turbulence separation between aircraft f_i and aircraft f_j ;

Δ : the time of aircraft occupying runway;

$B_{ijr} = 1$, if aircraft f_i uses the runway r before aircraft f_j ; 0, otherwise;

ETA_i : the estimated starting taxiing time of landing aircraft f_i after landing;

ETP_i : the estimated push-out time of take-off aircraft f_i ;

ETD_i : the estimated take-off time of take-off aircraft f_i ;

TBT_i : the starting taxiing time of aircraft f_i ;

TL_i : the true taxiing route length of aircraft f_i ;

SL_i : the shortest taxiing route length of aircraft f_i from the beginning to end;

D_α : the total extra taxiing route length of airline α ;

p_α : the proportion that the number of airlines α accounts for the total airlines;

$(\sum p_\alpha)$: the proportion that the number of flights of airline α accounts for the total airlines accumulatively;

DT_α : the total waiting delay time of airline α ;

G_1 : the Gini coefficient of taxiing route;

G_2 : the Gini coefficient of waiting delay;

ϵ_1 : the parameter value of Gini coefficient G_1 ;

ϵ_2 : the parameter value of Gini coefficient G_2 .

3.3. Objective Functions. In hub airports, the taxiing route length of aircraft and waiting delay are the main factors affecting the efficiency of surface movement. Thus the objective functions of the paper are presented in two aspects: the taxiing route cost and the waiting delay time. Shortening taxiing route length and reducing waiting delay time of all aircraft in the planned period are the main goals in the aircraft scheduling.

As the speed of aircraft is constant during taxiing, the taxiing time and taxiing route length of aircraft are equivalent. In the paper, the taxiing time of aircraft refers to the difference between the start time and the end time of taxiing. The goal is to minimize the total taxiing time of all aircraft in planned period:

$$\min \sum_{f_i \in F} (T_{in_{k_i}^i} - T_{in_1^i}). \quad (1)$$

For landing, the waiting time before entering into the taxiway system is due to the busy use of surface resources, which can postpone the landing aircraft. The same reason can postpone the push-out time of the take-off aircraft from the gate zone. During the synergistic scheduling of taxiway and runway, with the premise of no aircraft affecting the taxiing safety, all aircraft should enter the taxiing system as soon as possible,

which can make the total waiting delay time of all aircraft shortest:

$$\min \sum_{f_i \in F} (T_{ini} - T_i^o). \quad (2)$$

3.4. Constraints

(1) *Taxiing Route Constraints.* If the runway exit, runway entrance, and gate of aircraft are known, the aircraft taxiing route constraints are to generate a feasible route from the starting point to the end for each aircraft:

$$R_{ipq} \leq C_{pq} \quad \forall f_i \in F, \forall n_p, n_q \in N. \quad (3)$$

Formula (3) ensures that each link of the taxiing route of any aircraft f_i needs to meet the capacity and the physical connectivity of taxiway in the airport:

$$\sum_{n_p \in N} R_{ipq} - \sum_{n_s \in N} R_{iqs} = \begin{cases} 1 & n_q = n_{k_i}^i \\ 0 & \text{other} \\ -1 & n_q = n_1^i \end{cases} \quad (4)$$

$$\forall f_i \in F, \quad \forall n_p, n_q, n_s \in N.$$

Formula (4) ensures that each aircraft is assigned a feasible taxiing route from the starting point to the end point and that all the surface nodes have a liquidity balance. That is to say, if an aircraft reaches a node of the airport surface, it must begin to taxi from this node to the other one:

$$\left(T_{ip} + \frac{L_{pq}}{V_i} - T_{iq} \right) \cdot R_{ipq} = 0, \quad \forall f_i \in F, \forall n_p, n_q \in N. \quad (5)$$

Formula (5) ensures that the taxiing route of aircraft is always continuous. It does not allow any aircraft to stop to wait during the course of taxiing. When the aircraft f_i is taxiing from node n_p to the next node n_q , formula (5) is to be $T_{iq} = T_{ip} + L_{pq}/V_i$:

$$\frac{L_{pq}}{V_i^{\max}} \cdot R_{ipq} - M(1 - R_{ipq}) \leq T_{iq} - T_{ip} \quad (6)$$

$$\forall f_i \in F, \quad \forall n_p, n_q \in N.$$

Formula (6) restricts the taxiing speed of all aircraft. The speed of each aircraft must be less than the maximal speed.

Formula (3)–(6) ensure that any aircraft f_i must have an effective taxiing route from its gate to its runway entrance or from its runway exit to its gate and have a continuous taxiing route. However, the route generated here is random, not

necessarily the optimal route. What is more, it may conflict with other aircraft during taxiing.

(2) *Taxiing Security Constraints.* Aircraft taxiing on the airport surface usually produces three types of taxiing conflicts: the node conflict, rear-end conflict, and head-on conflict.

Node Conflict. The interval time, at which the two aircraft go through the same node, does not meet the minimum safety requirements.

Rear-End Conflict. When two aircraft taxi in the same direction on a taxiway, the faster trailing aircraft may overtake the leading aircraft.

Head-On Conflict. When two aircraft taxi in the opposite direction on a taxiway, they have an encounter with each other.

The taxiing safety constraints of aircraft are to ensure the safety of aircraft taxiing and avoid taxiing conflicts above:

$$Z_{ijp} (T_{ip} + t_{ij}^e - T_{jpp}) \leq 0 \quad \forall f_i, f_j \in F, n_p \in N^i \cap N^j. \quad (7)$$

Formula (7) ensures the safety interval of two taxiing aircraft. When two aircraft go through the same node, there must be a safety time interval between them. Safety time interval is related to aircraft type, wake turbulence separation, taxiing speed, and so on:

$$Z_{ijp} - Z_{ijq} \leq 2 - (R_{ipq} + R_{jpp}) \quad \forall f_i, f_j \in F, \forall n_p, n_q \in N,$$

$$Z_{ijp} - Z_{ijq} \geq -2 + (R_{ipq} + R_{jpp}) \quad \forall f_i, f_j \in F, \forall n_p, n_q \in N. \quad (8)$$

Formula (8) can avoid rear-end conflicts during the taxiing of airport. The scene that one aircraft overtakes another must not occur during the taxiing:

$$Z_{ijp} - Z_{ijq} \leq 2 - (R_{ipq} + R_{jpp}) \quad \forall f_i, f_j \in F, \forall n_p, n_q \in N,$$

$$Z_{ijp} - Z_{ijq} \leq -2 + (R_{ipq} + R_{jpp}) \quad \forall f_i, f_j \in F, \forall n_p, n_q \in N. \quad (9)$$

Formula (9) can avoid head-on conflicts during aircraft taxiing. An aircraft must not have an encounter with other aircraft in a certain taxiway during taxiing.

(3) *Runway Operation Constraints.* The runway operation strategy has an important influence on the airport surface taxiing. The successive relationship between two aircraft using the same runway can be divided into four kinds: takeoff-takeoff, takeoff-landing, landing-takeoff, and landing-landing:

$$T_{ip} + t_{ij}^w - (1 - Z_{ijp})M \leq T_{jpp} \quad \forall f_i, f_j \in F^d, \forall n_p \in N_r. \quad (10)$$

Formula (10) ensures that in the case of takeoff-takeoff, two consecutive take-off aircraft must meet wake turbulence separation standards between them:

$$\begin{aligned} T_{in_{k_i}^i} + t_{ij}^w - (1 - B_{ijr})M &\leq T_{jn_{k_i}^i} - \Delta, \\ \forall f_i \in F^d, \quad \forall f_j \in F^a, \quad \forall n_{k_i}^i, n_{k_i}^j &\in N_r. \end{aligned} \quad (11)$$

Formula (11) ensures that in the takeoff-landing process, two aircraft must meet the take-off wake turbulence separation standards. The take-off aircraft must leave the runway before the trailing lands on the runway. The starting taxiing time of landing aircraft f_j equals the time difference of reaching the runway and runway occupancy time. If the starting taxiing time is $T_{jn_{k_i}^i}$, the time of landing on the runway is $T_{jn_{k_i}^i} - \Delta$. According to the wake turbulence separation standards, formula (11) is got above:

$$\begin{aligned} T_{in_{k_i}^i} - (1 - B_{ijr})M &\leq T_{jn_{k_j}^j}, \\ \forall f_i \in F^a, \quad \forall f_j \in F^d, \quad \forall n_{k_i}^i, n_{k_j}^j &\in N_r. \end{aligned} \quad (12)$$

Formula (12) ensures that, in the case of landing-takeoff, the take-off aircraft can enter the runway until the heading landing aircraft leaves the runway. The time of entrance to the runway for take-off aircraft f_j at least equals the time the landing aircraft f_i exiting the runway:

$$\begin{aligned} T_{in_{k_i}^i} - (1 - B_{ijr})M &\leq T_{jn_{k_j}^j}, \\ \forall f_i \in F^a, \quad \forall f_j \in F^d, \quad \forall n_{k_i}^i, n_{k_j}^j &\in N_r. \end{aligned} \quad (13)$$

Formula (13) assures that, in the case of landing-landing, the two aircraft must meet the wake turbulence safety interval standards. The trailing aircraft cannot land on runway until the heading aircraft taxis off the runway.

(4) *Taxiing Time Constraints.* The taxiing time constraints are used to ensure that the taxiing scheduling of aircraft should be in accordance with the requirements of flight schedule:

$$T_{in_{k_i}^i} \geq \text{ETA}_i \quad \forall f_i \in F^a. \quad (14)$$

Formula (14) ensures that the landing aircraft should begin to taxi after the estimated starting taxiing time:

$$T_{in_{k_i}^i} \geq \text{ETP}_i \quad \forall f_i \in F^d, \quad (15)$$

$$T_{in_{k_i}^i} \leq \text{ETD}_i \quad \forall f_i \in F^d. \quad (16)$$

Formula (15) ensures that a take-off aircraft must start taxiing after its push-out time. Formula (16) requires aircraft to complete the taxiing and reach the runway entrance before the estimated take-off time.

(5) *Fairness Constraints.* During the course of scheduling of aircraft, sometimes the route length of one airline's aircraft is always much longer than other airlines, or the total delay

is much larger than other airlines. Therefore, the fairness for various airlines needs to be taken into account during the taxiing route scheduling of aircraft. The Gini coefficient in the economics is introduced to restrain the fairness of taxiing scheduling and we use it to quantify the taxiing route fairness and delay fairness. The Gini coefficient can reflect the degree of unfairness for resources distribution. The smaller the value is, the more average the distribution is.

As the runway and the gate of an aircraft are known, the shortest path of each aircraft is different. If we want to weigh the fairness of the taxiing route, simply comparing the taxiing route length of each aircraft cannot meet the actual requirements and is not enough as well. Therefore we introduce the concept of extra taxiing distance.

Definition 1 (extra taxiing distance). When the runway and the gate of an aircraft are known, we can get a shortest path for it. Extra taxiing distance is the difference distance between actual route length obtained by planning and the shortest route length. The extra taxiing distance is produced in order to avoid conflicts in the surface taxiing scheduling.

The fairness constraints of aircraft taxiing route are in essence to distribute the extra taxiing distance produced by scheduling between airlines. The smaller the Gini coefficient for airlines is, the more means the taxiing scheduling is. The extra taxiing distance of airline α is as follows:

$$D_a = \sum_{i \in F_a} |\text{TL}_i - \text{SL}_i|. \quad (17)$$

TL_i is the actual length of taxiing route which is obtained by the planning schedule, so $\text{TL}_i = \sum_{k=1}^{k_i-1} L_{n_{k_i}^i, n_{k+1}^i}$. SL_i is the shortest route from the starting point to the end, which could be obtained from the Dijkstra algorithm.

The Gini coefficient is defined and calculated as follows:

$$G_1 = 1 + \sum_{a \in A} p_a D_a - 2 \sum_{a \in A} \overline{(\sum p_a)} D_a. \quad (18)$$

Letting G_1 be less than a small constraint parameter value ε_1 , the fairness constraint based on taxiing length could be established as follows:

$$1 + \sum_{a \in A} p_a D_a - 2 \sum_{a \in A} \overline{(\sum p_a)} D_a \leq \varepsilon_1. \quad (19)$$

Formula (19) ensures the fairness of taxiing route length:

$$G_2 = 1 + \sum_{a \in A} p_a \text{DT}_a - 2 \sum_{a \in A} \overline{(\sum p_a)} \text{DT}_a, \quad (20)$$

$$1 + \sum_{a \in A} p_a \text{DT}_a - 2 \sum_{a \in A} \overline{(\sum p_a)} \text{DT}_a \leq \varepsilon_2. \quad (21)$$

Similarly, formula (21) ensures the fairness of waiting delay of all the aircraft.

4. Simulation

At the beginning of simulation, we create the surface taxiing route sets for all landing and take-off aircraft. On this

basis, we encode for the characteristic of airport surface taxiing scheduling, design an improved genetic algorithm, use the actual data of an airport for simulation and analysis, and compare the results with the results of the ant colony optimization.

4.1. Establish Taxiing Route Sets. Establishing surface taxiing route sets is to get the shortest feasible routes for each aircraft from the origin to destination. Aircraft with different origins or destinations need different route sets. The aircraft surface taxiing route problem can be abstracted into point-to-point problem, and Dijkstra algorithm can generate the shortest route for any single aircraft. However, the dynamic scheduling of surface taxiing requires multiple different taxiing routes to meet their demands. The K shortest path method [22] is used to generate the first K shortest taxiing routes to establish the set of taxiing routes in the paper.

4.2. The Improved Genetic Algorithm. The improved genetic algorithm (IGA) for the surface taxiing scheduling is introduced in this section.

4.2.1. Gene Encoding. Selecting gene encoding has an important impact on the design and solution of the genetic algorithm. As aircraft surface taxiing scheduling is extremely complex, simply using the aircraft taxiing route nodes as a chromosome coding accepted by most general genetic algorithm will likely cause a single chromosome to be huge. In addition, the crossover and mutation manipulation will destroy the existing sequence of nodes, which can result in that the adjacent nodes in the sequence are not reachable in real network. Therefore, the paper designs the improved encoding as follows.

In the improved encoding, dual-chromosome is adopted. The length of the chromosome is equal to the number of aircraft needed to be scheduled. A chromosome consists of two rows. One is the aircraft taxiing route sequence number row and the other is the aircraft waiting delay time row. We can use only a chromosome to express the taxiing route and the scheduling for all aircraft. Letting the number of aircraft be i , then the chromosome can be expressed as $P = (\vec{R}, \vec{T})$, where $\vec{R} = (R_1, R_2, \dots, R_i)$ and $\vec{T} = (T_1, T_2, \dots, T_i)$. For example, R_1 is the R_1 st route for the 1st aircraft in the corresponding route set. T_1 represents waiting delay time for the 1st aircraft. It cannot exceed the maximum waiting delay time, so $T_i \in [0, \text{Max Delay}]$. This encoding mode not only is fit for the characteristics of aircraft surface taxiing scheduling, but also is easier for the achievement of crossover and mutation manipulation.

4.2.2. The Fitness Function. Population initialization can be generated by the random method, and then we can calculate the individual fitness. According to the characteristics of aircraft taxiing route scheduling, the fitness function designed should take account of the cost of aircraft taxiing (the total taxiing time), the aircraft average waiting delay time, and the number of conflict points:

$$\text{fitness} = 2 * f + t + 800 * c. \quad (22)$$

According to information of taxiing route line in the chromosome, each aircraft's taxiing time can be calculated, as well as the total taxiing time f of all aircraft.

The average waiting delay time of an aircraft t can be obtained according to waiting delay information in the second row of the chromosome. Each aircraft waiting delay should be as less as possible.

Aircraft taxiing with no conflict is the key of aircraft taxiing scheduling. The c indicates the number of taxiing conflict points. According to chromosomes information, the landing time of each aircraft at each node can be calculated first, and then the number of taxiing conflict points can be obtained based on formula (7) to formula (13). When two aircraft taxi in violation of any safety spatial or temporal requirement, a conflict is recorded. Because in the final aircraft taxiing route scheduling there must be no conflict, this parameter should be set to a larger weight in the simulation.

Two methods are adopted to resolve aircraft taxiing route conflicts in taxiing: one is to control the taxiing time of aircraft in the system, and the other is to select one taxiing route dynamically from the predetermined route set. In the paper, the IGA simultaneously uses these two methods to solve the taxiing conflicts. Under the premise of no taxiing conflict in the whole airport surface, we minimize the taxiing cost and waiting delay time for all aircraft in the planning period.

4.2.3. Genetic Manipulation. Based on the above chromosome encoding and fitness calculation, the genetic manipulations can be done. Genetic manipulations mainly include selection, crossover, and mutation. The reinsertion process is added to select the new generation. The selection manipulation takes the random competing selection method. The crossover manipulation takes one-point crossover. According to the needs of the aircraft taxiing route scheduling, the improvements of the genetic manipulation are mainly reflected in two aspects: the reinsertion manipulation and the mutation manipulation.

Mutation Manipulation. In the paper, two mutation methods are used. One is a standard mutation based on aircraft taxiing route and waiting delay time. Namely, regular mutation is done on the first row of taxiing route and the second row of waiting delay time of the chromosome. The other is biased mutation based on waiting delay time. A random waiting time α is imposed to the chromosome delay row, $\alpha \in \{[-a, b] \mid a > b > 0\}$, such that $\alpha \in [-6, 2]$. As α is a random number in $[-6, 2]$, it is always biased negative. After adding the number to the chromosome, the aircraft waiting delay time can bias the direction of decreasing. This method can speed up the time that the individual with a small delay appears.

Reinsertion Manipulation. Reinsertion manipulation means replacing the most unsuited individuals in the previous generation of populations by the new suited individuals based on the fitness value. Because the aircraft surface taxiing route is quite complex, easily resulting in conflict points, and most individuals of the resulting generation are worse than

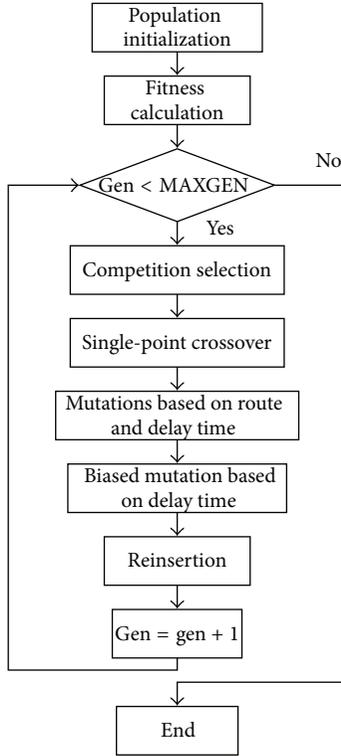


FIGURE 1: The main flowchart of the improved genetic algorithm.

the parent generation, the paper introduces the reinsertion manipulation. By reinsertion manipulation, we can use the small number of new individuals with the best fitness value to replace the corresponding number of individuals in the previous generation of population which could maximize the retention of superior individuals in the parent generation.

The main flowchart of the improved genetic algorithm used in the paper is shown in Figure 1.

4.3. Simulation Data. In the paper, some part of airfield configuration diagram of a large airport is used here (Figure 2), which contains 37 nodes and 48 edges, two independently running runways and three gate zones (T1, T2, and T3).

During the simulation, the paper first established a set with a certain number (set 15) between each gate and each runway threshold or between each runway exit and each gate zone. For example, according to Figure 2, we can establish a set with 15 taxiing routes for exit of runway 1 (node 32) and gate zone T1 (node 35). By permutations of the runways and gate zones known, we can get 12 sets and total 180 taxiing routes.

Assume that the taxiing speed of aircraft in taxiing way is 10 m/s; the minimum safety distance is 200 m; aircraft runway occupancy time is 30 s; the capacity of two taxiing way direct connected nodes $C_{pq} = 1$; wake turbulence separation criterion is 2 minutes.

Genetic Parameters. The generation gap $GGAP = 0.8$, crossover probability $XOVR = 0.6$, mutation probability routine $PM1 = 0.1$, biased mutation probability $PM2 = 0.5$,

TABLE 1: Information of landing and take-off aircraft.

Aircraft number	Airline	Arrival/departure	Gate zone	Runway
1	A_1	Departure	T1	1
2	A_2	Departure	T1	2
3	A_3	Departure	T1	2
4	A_1	Departure	T2	1
5	A_2	Departure	T2	1
6	A_3	Departure	T2	2
7	A_2	Departure	T3	1
8	A_1	Departure	T3	1
9	A_2	Departure	T3	2
10	A_3	Departure	T3	2
11	A_1	Arrival	T1	1
12	A_2	Arrival	T1	1
13	A_3	Arrival	T1	2
14	A_2	Arrival	T2	1
15	A_2	Arrival	T2	1
16	A_1	Arrival	T2	2
17	A_3	Arrival	T2	2
18	A_2	Arrival	T3	1
19	A_1	Arrival	T3	2
20	A_2	Arrival	T3	2

biased mutation random waiting time $\alpha \in [-20, 10]$, genetic generations $MAXGEN = 200$, population size $NIND = 40$, the maximum allowable waiting time $DETIME = 30$ min, and the probability of reinsertion $RX = 0.2$.

Fairness Parameters. Consider $\epsilon_1 = 0.35$, $\epsilon_2 = 0.35$.

In the aircraft scheduling simulation, assume that there are 20 aircraft needed to be scheduled. The estimated landing time of arrivals and the estimated time of push-out for departures are the time when $t = 0$ s. The estimated time for departures must meet the maximum allowable waiting time. Information about the arrivals and departures is shown in Table 1.

4.4. Simulation Results

4.4.1. Analysis of Algorithm Performance. Using MATLAB to code the IGA and based on the surface data of one of our country's large airports, we simulate the model established above. The evolution of the genetic algorithm is shown in Figure 3.

The evolutionary process of the IGA shows that in the first 20 generations the evolution of the population converges rapidly, and after 70 generations, the evolution levels off. After 140 iterations, the average fitness value of the population and the fitness value of the best individual start stabilizing. It takes 7 minutes and 55 seconds to run the program, which could meet the timeliness requirements of resources operating and scheduling on airport surface.

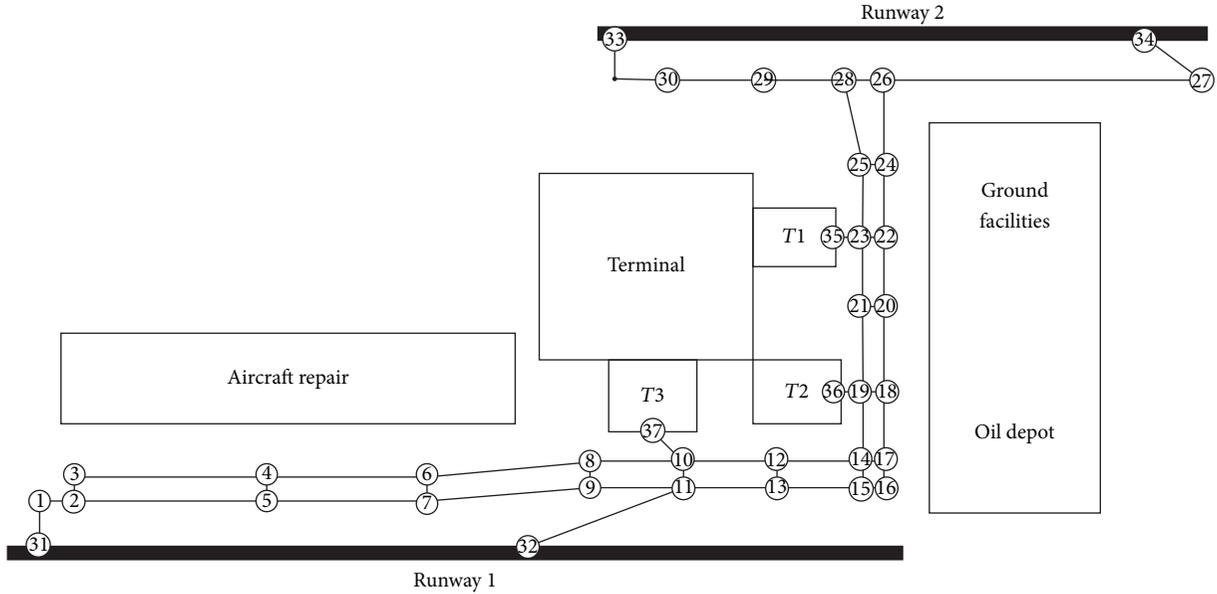


FIGURE 2: Part of airfield configuration of a large airport's taxiways and runways.

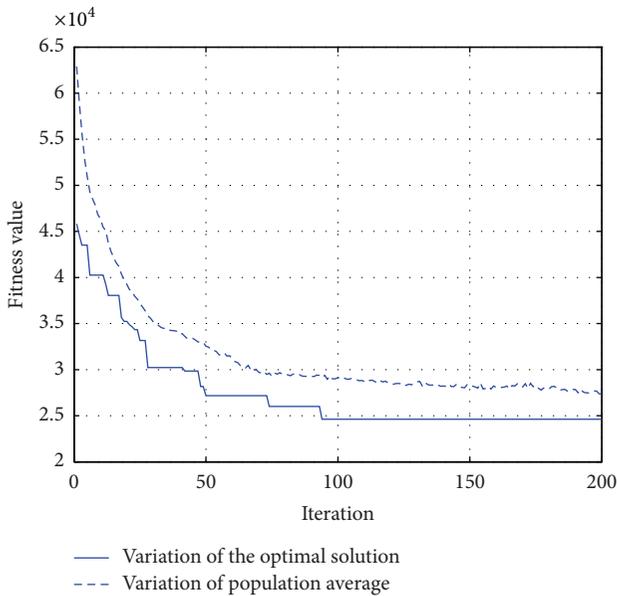


FIGURE 3: The evolutionary process of the IGA.

4.4.2. *Analysis of Simulation Results.* Through the simulation, we can get the best individual in the last generation population. Meanwhile, the paper selects a scheduling result through the ant colony optimization (ACO) to contrast the simulation results, which is shown in Table 2.

IGA and ACO are both conflict-free scheduling methods. In the two methods, each aircraft has a taxiing route and waiting delay time. We can see from Table 2, in the IGA, that 4 aircraft start taxiing with no waiting delay, waiting delay time of 9 aircraft is within 500 s, and waiting delay of 2 aircraft is longer than 1500 s. In the ACO, 4 aircraft have no waiting delays, but only waiting delay of 7 aircraft is within 500 s and

TABLE 2: The comparison of results between IGA and ACO.

Aircraft number	IGA		ACO	
	Route number	Waiting delay time/s	Route number	Waiting delay time/s
1	12	1773	11	545
2	26	0	23	0
3	26	681	25	1657
4	33	588	31	1520
5	39	0	40	135
6	50	464	53	298
7	71	1041	73	1297
8	62	1299	73	1059
9	85	897	78	0
10	83	1200	89	1761
11	92	1305	103	1285
12	101	189	105	743
13	120	1771	119	1400
14	134	909	121	927
15	134	0	130	0
16	148	0	144	975
17	137	569	139	0
18	157	378	159	355
19	175	410	177	1764
20	178	133	171	1608

Note. Due to the paper length, the path 12 sets with 180 routes cannot be fully demonstrated here.

waiting delay of 5 aircraft is longer than 1500 s. Therefore, the simulation results of IGA are better than the results of ACO in the taxiing route scheduling.

TABLE 3: The specific and detailed scheduling results for aircraft taxiing in IGA.

Aircraft number	The nodes and the arrival time																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																						
1	35	23	22	20	18	17	14	12	10	8	6	4	3	2	1	31		1773	1780	1788	1841	1846	1872	1880	1912	1947	1982	2043	2103	2175	2184	2197	2214	2							35	23	25	28	29	30	33											0	7	34	67	97	133	160				3							35	23	25	28	29	30	33											681	688	715	784	778	814	841				4				36	19	14	12	10	8	6	7	5	2	1	31						588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527
	1773	1780	1788	1841	1846	1872	1880	1912	1947	1982	2043	2103	2175	2184	2197	2214	2							35	23	25	28	29	30	33											0	7	34	67	97	133	160				3							35	23	25	28	29	30	33											681	688	715	784	778	814	841				4				36	19	14	12	10	8	6	7	5	2	1	31						588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																	
2							35	23	25	28	29	30	33											0	7	34	67	97	133	160				3							35	23	25	28	29	30	33											681	688	715	784	778	814	841				4				36	19	14	12	10	8	6	7	5	2	1	31						588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																		
							0	7	34	67	97	133	160				3							35	23	25	28	29	30	33											681	688	715	784	778	814	841				4				36	19	14	12	10	8	6	7	5	2	1	31						588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																			
3							35	23	25	28	29	30	33											681	688	715	784	778	814	841				4				36	19	14	12	10	8	6	7	5	2	1	31						588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																				
							681	688	715	784	778	814	841				4				36	19	14	12	10	8	6	7	5	2	1	31						588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																					
4				36	19	14	12	10	8	6	7	5	2	1	31						588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																						
				588	594	620	652	687	722	783	792	852	924	937	954		5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																							
5			36	19	18	17	14	12	10	8	6	4	5	2	1	31				0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																								
			0	6	14	40	48	80	115	150	211	271	280	352	365	382	6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																									
6				36	19	21	20	22	24	25	28	29	30	33							464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																										
				464	470	502	510	536	563	571	604	634	670	697			7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																											
7			37	10	8	6	7	5	4	3	2	1	31							1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																												
			1041	1054	1089	1150	1159	1219	1228	1300	1309	1322	1339				8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																													
8				7	10	8	6	4	5	2	1	31									1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																														
				1299	1312	1347	1408	1468	1477	1549	1562	1579					9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																															
9		37	10	12	14	19	18	20	22	23	25	28	29	30	33				897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																
		897	910	945	977	1003	1011	1043	1069	1077	1104	1137	1167	1203	1230		10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																	
10		37	10	12	14	19	18	20	22	24	25	28	29	30	33				1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																		
		1200	1213	1248	1280	1306	1314	1346	1372	1399	1407	1440	1470	1506	1533		11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																			
11			32	11	10	12	14	17	18	20	22	23	35							1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																				
			1305	1367	1377	1412	1444	1452	1478	1510	1536	1544	1551				12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																					
12			32	11	13	15	14	17	18	20	22	23	35							189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																						
			189	251	286	318	328	336	362	394	420	428	435				13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																							
13				34	27	26	28	25	23	35											1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																								
				1771	1795	1914	1929	1962	1989	1996							14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																									
14			32	11	13	15	16	17	14	19	36									909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																										
			909	971	1006	1038	1046	1056	1064	1090	1096						15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																											
15			32	11	13	15	16	17	14	19	36									0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																												
			0	62	97	129	137	147	155	181	187						16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																													
16			34	27	26	24	25	23	22	20	21	19	36							0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																														
			0	24	143	175	183	210	218	244	252	284	290				17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																															
17				34	27	26	24	22	20	21	19	36									569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																
				569	593	712	744	771	797	805	837	843					18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
18							32	11	10	37														378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																		
							378	440	450	463							19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																			
19			34	27	26	24	22	20	18	19	14	15	13	11	10	37				410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																				
			410	434	553	585	612	638	670	678	704	714	746	781	791	804	20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																					
20			34	27	26	24	22	23	21	19	14	15	13	11	10	37				133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																						
			133	157	276	308	335	343	369	401	427	437	469	504	514	527																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																							

Note. In the table, the up row indicates the upstream route nodes for each aircraft going through; the down row indicates the time reaching each node.

According to the aircraft taxiing scheduling results in Table 2 and the route sets, we can get the specific and detailed scheduling for aircraft's taxiing in the IGA, including the node which each aircraft passes through and the time of arriving at each node in Table 3.

The comparisons and statistics of taxiing route and waiting delay time for each aircraft are shown in Table 4. The shortest route (SHR), length of taxiing route (LTR), the extra

taxiing distance (EXTD), and the waiting delay time (WDT) information are shown in Table 4.

From Table 4, we can see that, in the IGA, the aircraft 2, 3, 4, 8, 17, and 18 all select the shortest route, greatly reducing the cost of taxiing. And to avoid conflicts, the other aircraft do not select the shortest route which leads to extra taxiing costs. In the ACO, only five aircraft choose the shortest route.

TABLE 4: The comparisons and statistics of aircraft taxiing.

Aircraft number	Airline	SHR/m	IGA			ACO		
			LTR/m	EXTD/m	WDT/s	LTR/m	EXTD/m	WDT/s
1	1	4250	4410	160	1773	4420	170	545
2	2	1600	1600	0	0	1760	160	0
3	3	1600	1600	0	681	2340	740	1657
4	1	3660	3660	0	588	3660	0	1520
5	2	3660	3820	160	0	3820	160	135
6	3	2170	2330	160	464	2330	160	298
7	2	2800	2980	180	1041	2990	190	1297
8	1	2800	2800	0	1299	2990	190	1059
9	2	3170	3330	160	897	3390	220	0
10	3	3170	3330	160	1200	3330	160	1761
11	1	2300	2460	160	1305	2460	160	1285
12	2	2300	2460	160	189	2460	160	743
13	3	2170	2250	80	1771	2170	0	1400
14	2	1710	1870	160	909	1710	0	927
15	2	1710	1870	160	0	1710	0	0
16	1	2740	2900	160	0	2980	240	975
17	3	2740	2740	0	569	2740	0	0
18	2	850	850	0	378	2190	1340	355
19	1	3740	3940	200	410	3940	200	1764
20	2	3740	3940	200	133	3820	80	1608
Total		52880	55140	2260	13607	57210	4330	17329

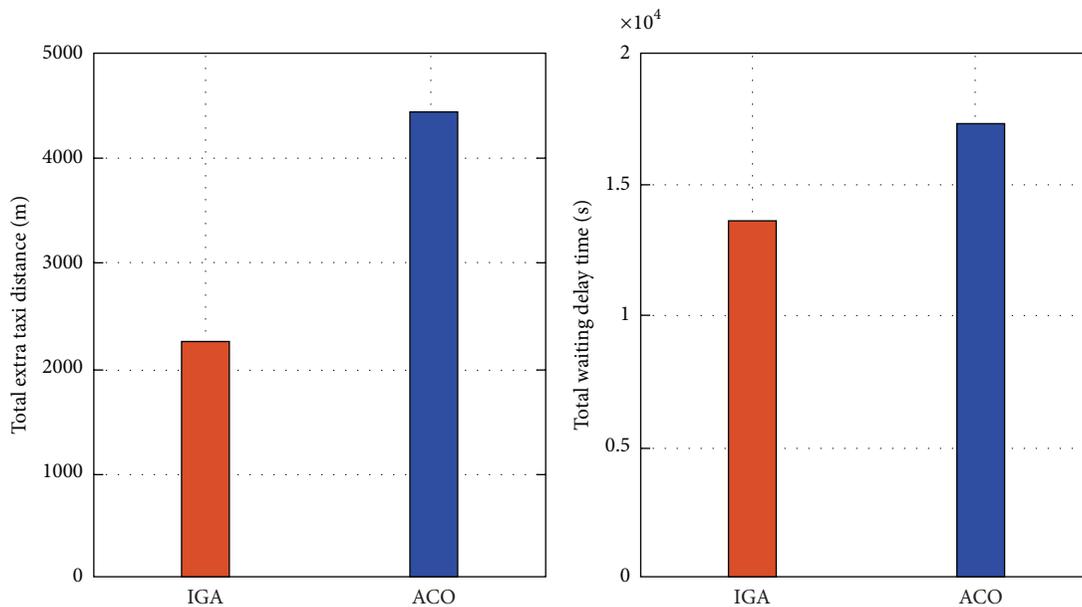


FIGURE 4: Comparison of aircraft's taxiing scheduling in IGA and ACO.

From Table 4, in the ACO, the total taxiing distance is 57210 m, the total extra taxiing route is 4330 m, and the total waiting delay time is 17329 s. In the IGA of this paper, the total taxiing distance is 55140 m; the total extra taxiing route is 2260 m, which is 47.8% lower than the ACO; the total delay time is 13607 s, which is reduced by 21.5% compared with

the ACO, as seen in Figure 4. The advantages of the IGA have been shown. In the specially designed genetic algorithm for surface taxiing scheduling, enough diverse individuals are generated through crossover and mutation manipulation, and excellent variation individuals will enter into the next generation with greater proportion through the selection and

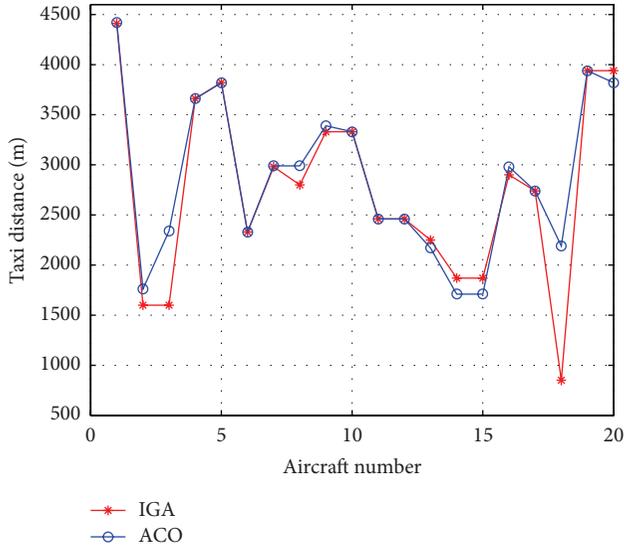


FIGURE 5: Comparison of taxiing route of aircraft in IGA and ACO.

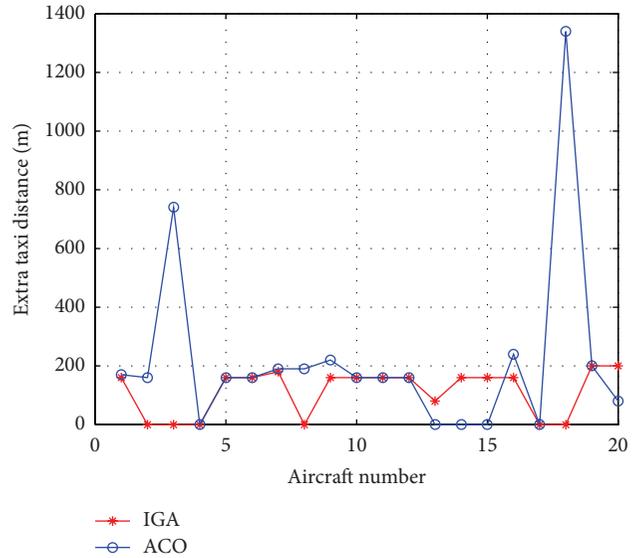


FIGURE 6: Comparison of the extra taxiing distance of aircraft in IGA and ACO.

reinsertion manipulation. The evolution of the IGA would go in the direction of no conflict, less waiting delay time, and less extra taxiing distance.

According to the simulation results in Table 4, we can draw the contrastive analysis diagram of the aircraft taxiing distance and the extra taxiing route in the IGA and ACO, which is shown in Figures 5 and 6.

From Figures 5 and 6, compared with the ACO, the results of the IGA are that each aircraft selects a shorter taxiing route as possible, and the taxiing distance is significantly reduced, especially for the extra taxiing route distance. From Figures 5 and 6, it can be seen that, in the IGA, the taxiing distance of 6 aircraft is reduced, 10 aircraft are constant, and 4 aircraft are increased. However, due to a big margin reduced in aircraft 3 and aircraft 18, the total extra taxiing distance reduces by 47.8% compared with the ACO. The IGA reduces the overall taxiing route distance and operating cost effectively. The 47.8% decrease is mainly caused by the concept of the extra taxiing distance. According to the concept, the extra taxiing distance could be reduced to zero with 100% decrease, if all aircraft taxi with their shortest taxiing route. That is to say, not all aircraft have extra taxiing distance. If some (not too many) aircraft's routes are improved, obvious reflection will be shown in this value.

According to Table 4, the contrastive analysis diagram of the waiting delay time can also be obtained in Figure 7.

Waiting delay time refers to the time postponed off the scheduling. The delay time is the difference between the time an aircraft actually begins to enter the surface taxiway system and the earliest possible time it can enter the taxiing system. Table 2 and Figure 7 show that, in the IGA, at the time $t = 0$ s, landing aircraft 2 and aircraft 5 leave gate 35 (T_1) and gate 36 (T_2), respectively, and taking-off aircraft 15 and aircraft 16 leave the runway 1 (exit node 32) and runway 2 (exit node 34), respectively. Therefore, 4 aircraft start to taxi simultaneously. Aircraft 1 is the last aircraft to begin taxiing at the time

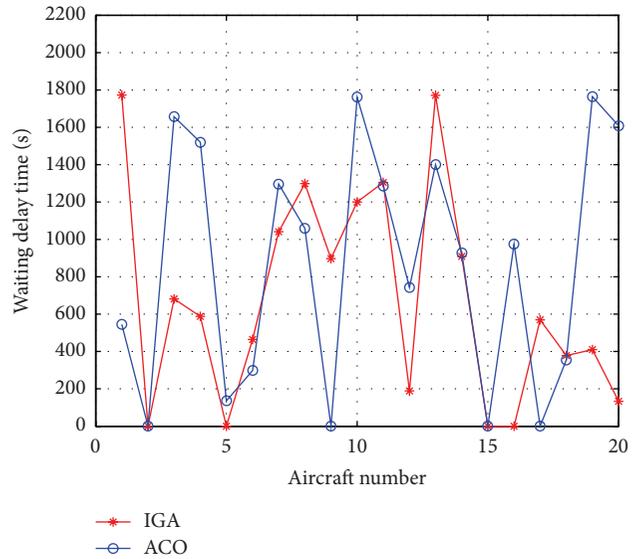


FIGURE 7: Comparison of the waiting delay time of aircraft in IGA and ACO.

1773 s. But it meets the condition that all aircraft can finish the taxiing or begin taxiing in the interval of 30 minutes. Compared with the ACO, 9 aircraft are shortened on waiting delay time, 5 aircraft are unchanged, and 6 aircraft are longer. On the whole, the IGA reduces the aircraft waiting delay time and improves the efficiency of airport surface.

The optimizing model provides a seamless connection between taxiway and runway. Runway queuing and scheduling are integrated into the model. The model considers different stakeholders' requirements, ensuring the safety of airport surface movement, reducing delay and taxiing cost, and improving the utilization efficiency of airport resources. The genetic algorithm used here is greatly improved which

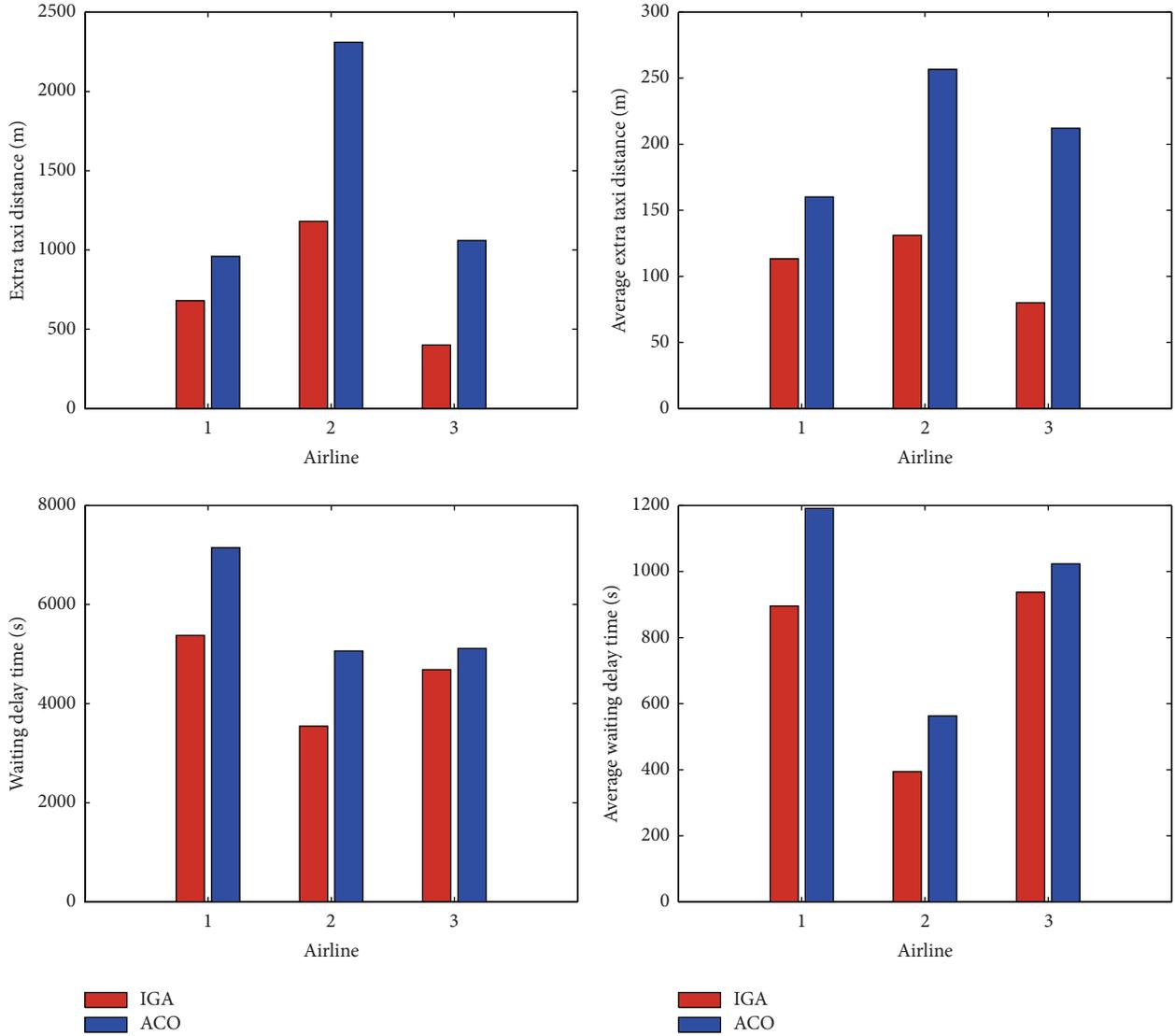


FIGURE 8: Analysis on taxiing scheduling of airlines.

is more suitable for the airport surface taxiing scheduling problem and the optimizing model. In the IGA, aircraft could change the initial taxiing time and the taxiing route to optimize the taxiing and scheduling in large data search space rapidly as shown in the results.

4.4.3. The Gini Coefficient and Fairness Analysis for Airlines

(1) Analysis of Scheduling Fairness. According to Table 4, we can calculate the extra taxiing distance (EXTD), the average extra taxiing distance (AETD), the waiting delay time (WDT), and the average waiting delay time (AWDT) for each airline in Table 5 and Figure 8.

It can be seen in Table 5 and Figure 8, in the IGA, that the average extra taxiing distances of each airline are 113.3 m, 131.1 m, and 80 m, respectively. The Gini coefficient of extra taxiing distance is 0.3412. The average delay time of each airline is 895.8 s, 394.1 s, and 937 s, respectively. The Gini

coefficient of the average waiting delay time is 0.3422. Both Gini coefficients meet the requirement within a reasonable range. In the ACO, the Gini coefficient of extra taxiing distance is 0.2886; it is fair but the overall extra distance is too long. The Gini coefficient of the average waiting delay time is 0.4351, which exceeds the required value 0.35. The results indicate that the distribution of the delay between airlines is not fair enough.

The optimizing model considers the fairness of scheduling, and the IGA could well finish the solving. The extra taxiing distance and waiting delay time are distributed between airlines fairly. Thus, the interests of various airlines are protected.

(2) Impact Analysis on Different Fairness Parameters. The paper mainly analyzes the impact of the two fairness constraint parameters on optimization results. In order to control variables, we set ϵ_1 or ϵ_2 to be 1 and the other to be 0.25,

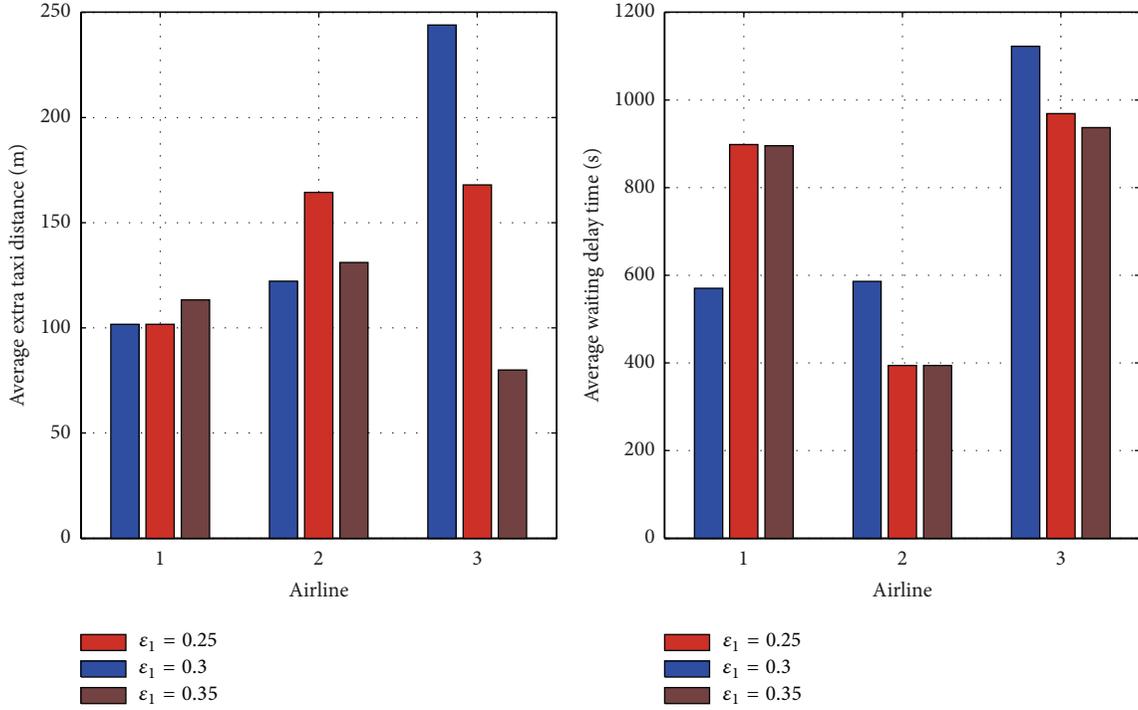


FIGURE 9: Optimization results of taxiing scheduling for each airline with different parameters.

TABLE 5: Comparison of the scheduling taxiing route between IGA and ACO.

Airline	Number of aircraft	IGA				ACO			
		EXTD/m	AETD/m	AWDT/s	AWDT/s	EXTD/m	AETD/m	WDT/s	AWDT/s
1	6	680	113.3	5375	895.8	960	160	7148	1191.3
2	9	1180	131.1	3547	394.1	2310	256.7	5065	562.8
3	5	400	80	4685	937.0	1060	212	5116	1023.2
Gini coefficient		0.3412		0.3422		0.2886		0.4351	

TABLE 6: Optimization results of taxiing route with different parameters.

$\epsilon_2 = 1, \epsilon_1$	Total extra taxiing distance/m	Total waiting delay time/s
0.25	2930	14312
0.3	2930	13782
0.35	1569	12468

0.3, and 0.35, respectively. For example, let $\epsilon_2 = 1$ in the optimization, and the optimization results and properties are shown in Table 6. The average extra taxiing distance and the average waiting delay time of each airline are shown in Table 7 and Figure 9.

From Table 6, we can see that, with the increase of Gini coefficient (ϵ_2) controlling the delay fairness (namely, fairness reduced), the total extra taxiing distance and the total waiting delay time are falling (namely, efficiency increased). Table 7 and Figure 9 also show that the average extra taxiing distance and the average waiting delay time for each airline are decreasing. This illustrates that the efficiency and fairness

TABLE 7: Optimization results of each airline with different parameters.

Airline	$\epsilon_2 = 1$ Average taxiing distance						Average waiting delay time		
	ϵ_1			ϵ_1			ϵ_1		
	0.25	0.3	0.35	0.25	0.3	0.35	0.25	0.3	0.35
1	101.7	101.7	113.3	570.5	898.5	895.8			
2	122.2	164.4	131.1	586.2	394.1	394.1			
3	244	168	80	1122.6	968.8	937			

of scheduling are mutually affected and restricted. So the selection of fairness parameter values has an important impact on the entire scheduling optimization results.

5. Conclusions

The airport surface taxiing route scheduling between taxiway and runway in hub airport problem is studied here. An optimization model based on safety, efficiency, and fairness is established for the aircraft surface taxiing problem. The

paper also combines the needs and interests of the air traffic controllers, airlines, and airport authorities. The occupancy of the runway has an important impact on the surface taxiing and is integrated into the airport surface scheduling. The IGA adapted to the problem and model is designed to optimize and solve the problem. Simulation data comes from a large airport. Simulation results show that compared with the ACO, the paper proposed the total extra taxiing distance which is reduced by 47.8% and the total waiting delay time which is decreased by 21.5%. The efficiency of the airport surface taxiing is effectively improved and the surface taxiing fairness interairline is enhanced. The optimal solution got here is in line with expectations.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China and Civil Aviation Administration of China (no. U1333117); The National Natural Science Foundation of China (NSFC) (no. 61104159); China Postdoctoral Science Foundation (no. 2012M511275); and The Fundamental Research Funds for the Central Universities (no. NS2013067).

References

- [1] SESAR, "Milestone deliverable D1: air transport framework: the current situation," Tech. Rep. DLM-0602-001-03-00, Eurocontrol, 2006.
- [2] Transport Canada, "National Civil Aviation Safety Subcommittee on Runway Incursions: final report," Tech. Rep. TP13795E, Transport Canada, Ottawa, Canada, 2000.
- [3] J. W. Smeltink and M. J. Soomer, "An optimisation model for airport taxi scheduling," in *Proceedings of the INFORMS Annual Meeting*, Denver, Colo, USA, 2004.
- [4] S. Rathinam, J. Montoya, and Y. Jung, "An optimization model for reducing aircraft taxi times at the Dallas Fort Worth International Airport," in *Proceedings of the 26th Congress of International Council of the Aeronautical Sciences (ICAS '08)*, pp. 14–19, September 2008.
- [5] J. B. Gotteland, N. Durand, J. M. Alliot et al., "Aircraft ground traffic optimization," in *Proceedings of the 4th International USA/Europe Air Traffic Management Research and Development Seminar (ATM '01)*, 2001.
- [6] S. J. Landry, X. W. Chen, and S. Y. Nof, "A decision support methodology for dynamic taxiway and runway conflict prevention," *Decision Support Systems*, vol. 55, no. 1, pp. 165–174, 2013.
- [7] K. Anderson, F. Carr, E. Feron et al., "Analysis and modeling of ground operations at hub airports," in *Proceedings of the 3rd USA/Europe Air Traffic Management R&D Seminar*, pp. 13–167, Napoli, Italy, 2000.
- [8] D. E. Pitfield, A. S. Brooke, and E. A. Jerrard, "A Monte-Carlo simulation of potentially conflicting ground movements at a new international airport," *Journal of Air Transport Management*, vol. 4, no. 1, pp. 3–9, 1998.
- [9] G. Keith, A. Richards, and S. Sharma, "Optimization of taxiway routing and runway scheduling," in *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Honolulu, Hawaii, USA, August 2008.
- [10] H. Balakrishnan and J. Yoon, "A framework for coordinated surface operations planning at dallas-fort worth international airport," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, vol. 3, pp. 2382–2400, August 2007.
- [11] P. C. Ruling and H. G. Visser, "Optimal airport surface traffic planning using mixed-integer linear programming," *International Journal of Aerospace Engineering*, vol. 2008, no. 1, pp. 1–11, 2008.
- [12] R. Anderson and D. Milutinović, "An approach to optimization of airport taxiway scheduling and traversal under uncertainty," *Proceedings of the Institution of Mechanical Engineers Part G: Journal of Aerospace Engineering*, vol. 227, no. 2, pp. 273–284, 2013.
- [13] J. You and S.-C. Han, "Taxi route optimization algorithm of airport surface based on multi-agent," *Journal of Traffic and Transportation Engineering*, vol. 9, no. 1, pp. 109–112, 2009.
- [14] Y. Wang, M. Hu, and W. Su, "Dynamic taxiway routing algorithm based on conflict avoidance," *Journal of Southwest Jiaotong University*, vol. 44, no. 6, pp. 933–939, 2009.
- [15] Á. G. Marín, "Airport management: taxi planning," *Annals of Operations Research*, vol. 143, no. 1, pp. 191–202, 2006.
- [16] A. Marín and E. Codina, "Network design: taxi planning," *Annals of Operations Research*, vol. 157, no. 1, pp. 135–151, 2008.
- [17] F. Carr, A. Evans, J.-P. Clarke, and E. Feron, "Modeling and control of airport queueing dynamics under severe flow restrictions," in *Proceedings of the American Control Conference*, vol. 2, pp. 1314–1319, IEEE, May 2002.
- [18] H. Idris, P. J. Clarke, R. Bhuvu, and L. Kang, "Queueing model for taxi-out time estimation," *Air Traffic Control Quarterly*, vol. 10, no. 1, pp. 1–22, 2002.
- [19] H. Baik, H. D. Sherali, and A. A. Trani, "Time-dependent network assignment strategy for taxiway routing at airports," *Transportation Research Record*, vol. 1788, no. 1, pp. 70–75, 2002.
- [20] J. García, A. Berlanga, J. M. Molina, and J. R. Casar, "Optimization of airport ground operations integrating genetic and dynamic flow management algorithms," *AI Communications*, vol. 18, no. 2, pp. 143–164, 2005.
- [21] J.-B. Gotteland and N. Durand, "Genetic algorithms applied to airport ground traffic optimization," in *Proceedings of the Congress on Evolutionary Computation (CEC '03)*, pp. 544–551, IEEE, December 2003.
- [22] D. Eppstein, "Finding the k shortest paths," *SIAM Journal on Computing*, vol. 28, no. 2, pp. 652–673, 1998.

Research Article

Study of On-Ramp PI Controller Based on Dual Group QPSO with Different Well Centers Algorithm

Tao Wu,¹ Xi Chen,² and Yusong Yan³

¹ School of Computer Science, Chengdu University of Information Technology, Chengdu 610225, China

² School of Computer Science & Technology, Southwest University for Nationalities, Chengdu 610041, China

³ School of Computer Science & Technology, Southwest Jiaotong University, Chengdu 610031, China

Correspondence should be addressed to Tao Wu; dorawu840720@gmail.com

Received 31 May 2014; Revised 11 November 2014; Accepted 12 November 2014

Academic Editor: Baozhen Yao

Copyright © 2015 Tao Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A novel quantum-behaved particle swarm optimization (QPSO) algorithm, dual-group QPSO with different well centers (DWC-QPSO) algorithm, is proposed by constructing the master-slave subswarms. The new algorithm was applied in the parameter optimization of on-ramp traffic PI controller combining with nonlinear feedback theory. With the critical information contained in the searching space and results of the basic QPSO algorithm, this algorithm avoids the rapid disappearance of swarm diversity and enhances the global searching ability through collaboration between subswarms. Experiment results on an on-ramp traffic control simulation show that DWC-QPSO can be well applied in the study of on-ramp traffic PI controller and the comparison results illustrate that DWC-QPSO outperforms other evolutionary algorithms with enhancement in both adaptability and stability.

1. Introduction

Interference caused by intersection and convergence between different traffic flows at the traffic bottlenecks such as on-ramp not only seriously reduces the efficiency of freeway but also leads to a reduction of the entire traffic network efficiency. On-ramp control is the most effective way to improve traffic condition of freeway. Therefore, it is necessary to study the traffic flow and develop effective traffic management and control measures on the basis of existing facilities at the on-ramp. With the development of computer technology, computational intelligence provides an effective method which can select the optimal control parameters for ramp traffic control strategy according to different conditions to solve the on-ramp traffic flow control problem. Now, the ant colony optimization (ACO) [1], particle swarm optimization (PSO) [2], artificial neural network (ANN) [3], cellular automata (CA) [4], and genetic algorithms (GA) [5, 6] have been used in on-ramp control.

Although the computational intelligence methods above have made some achievements on on-ramp traffic control,

they still have the following limitations: firstly, the convergence rate of PSO, ACO, and GA is so slow as to result in premature convergence and low precision. Secondly, due to the fact that traffic flows have real-time and nonlinear characteristics, the existing models are not fast enough to respond to real-time information, thus having limited applications. To solve these problems, a novel quantum-behaved particle swarm optimization (QPSO) algorithm, the dual-group QPSO with different well centers (DWC-QPSO) algorithm, is proposed in this paper based on quantum-behaved particle swarm optimization (QPSO) algorithm. With the critical information contained in the searching space, DWC-QPSO avoids the rapid disappearance of swarm diversity and enhances the global searching ability through collaboration between subswarms. Then the on-ramp traffic PI controller is designed based on DWC-QPSO algorithm by optimizing the values of parameters K_p and K_i . Experiment results show that DWC-QPSO algorithm can be applicable in the ramp traffic PI controller and the comparison results illustrate that DWC-QPSO outperforms other evolutionary algorithms with enhancement in both adaptability and stability.

This paper is organized as follows. Section 1 gave an introduction of on-ramp control problem; Section 2 introduced the PSO algorithm and QPSO algorithm. Then, a new algorithm, DWC-QPSO algorithm, is proposed in Section 3, and the convergence performance of the novel method is analysed by test functions experiment results; Section 4 explained the on-ramp traffic model and illustrated the computation process of DWC-QPSO algorithm optimizing the parameters of the on-ramp PI controller. In the end, Section 5 focuses on the simulation results analysis and Section 6 is the conclusion of this work.

2. PSO Algorithm and QPSO Algorithm

2.1. PSO Algorithm. Particle swarm optimization (PSO) algorithm is an evolutionary computation method proposed by Kennedy and Eberhart in 1995 [7–9]. It is one pattern of the latest swarm intelligence (SI) optimization algorithms. The basic idea of PSO algorithm derived from the study of bird behavior, through simulating birds' prey behavior to achieve the purpose of problem optimization.

In the PSO algorithm, each potential solution of the problem is abstracted as a particle with no weight and no volume. These particles fly at a certain speed in the D -dimensional search space [10]. Assuming the particle swarm population size is N , the particle i 's position and velocity at time step t are, respectively, expressed as follows.

The position of particle i at time step t :

$$X_i(t) = \{x_{i1}(t), x_{i2}(t), \dots, x_{iD}(t)\}. \quad (1)$$

The velocity of particle i at time step t :

$$V_i(t) = \{V_{i1}(t), V_{i2}(t), \dots, V_{iD}(t)\}. \quad (2)$$

p_{best} is used to describe the particle i 's personal best solution and g_{best} is used to describe the best position found by the neighborhood of particle i , at time step t :

$$\begin{aligned} p_{\text{best}} : P_i(t) &= \{p_{i1}(t), p_{i2}(t), \dots, p_{iD}(t)\}; \\ g_{\text{best}} : G(t) &= \{p_{g1}(t), p_{g2}(t), \dots, p_{gD}(t)\}. \end{aligned} \quad (3)$$

In the standard PSO model, the updated formula of particles' velocity and position on each dimension is as follows:

$$\begin{aligned} V_{i,j}(t+1) &= wV_{i,j}(t) + c_1r1_{i,j}(t)(P_{i,j}(t) - X_{i,j}(t)) \\ &\quad + c_2r2_{i,j}(t)(G_j(t) - X_{i,j}(t)), \end{aligned} \quad (4)$$

$$X_{i,j}(t+1) = X_{i,j}(t) + V_{i,j}(t+1)$$

for $i = 1, \dots, N$ and $j = 1, \dots, D$, where

- (i) N is the total number of particles in the swarm,
- (ii) D is the dimension of the problem, that is, the number of parameters of the function being optimized,
- (iii) w is the inertia weight,
- (iv) c_1 and c_2 are acceleration coefficients,
- (v) $r1_{i,j}(t), r2_{i,j}(t) \sim U(0, 1)$.

As PSO has simple concept, a small number of parameters, and desirable performance, PSO has become a very promising optimization tool and attracted extensive attention and has been successfully applied in many areas, such as function optimization, power systems, data mining, and wireless sensor network. However, some problems of this novel algorithm remain to be solved. One of these problems is that particle swarm cannot converge on the global optimal solution with one hundred percent probability, which has been proven by Bergh [11].

2.2. Quantum-Behaved Particle Swarm Optimization Algorithm. In order to solve this problem, Sun [12] proposed the quantum-behaved particle swarm optimization algorithm in the quantum space. According to the results Clerc and Kennedy [13] analyzed the particles' orbit in the PSO algorithm, and QPSO algorithm establishes a δ potential well to impact particles' convergence at the local attraction points $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$. The position of p_i is calculated by the following equations:

$$p_{i,j}(t) = \frac{c_1r1_{i,j}(t)P_{i,j}(t) + c_2r2_{i,j}(t)G_j(t)}{c_1r1_{i,j}(t) + c_2r2_{i,j}(t)}, \quad 1 \leq j \leq D, \quad (5)$$

where $P_{i,j}$ and G_j denote the j -dimensional components of particle i 's past best solution (p_{best}) and global best solution (g_{best}), respectively; $r1$ and $r2$ are random factors which are independently distributed random variables; c_1 is the individual cognitive acceleration factor, and c_2 is the group cognitive acceleration factor:

$$\begin{aligned} L_{i,j}(t) &= 2\alpha \cdot |C_j(t) - X_{i,j}(t)| X_{i,j}(t+1) \\ &= p_{i,j}(t) \pm \alpha \cdot |C_j(t) - X_{i,j}(t)| \ln \left[\frac{1}{u_{i,j}(t)} \right], \end{aligned} \quad (6)$$

where $u_{i,j}(t) \sim U(0, 1)$, $L_{i,j}(t)$ is the length of the δ potential well, which is evaluated by formula (6).

$C(t)$ is defined as the population gravity center which is the average of all the best individual locations. It is calculated as follows:

$$\begin{aligned} C(t) &= (C_1(t), C_2(t), \dots, C_n(t)) = \frac{1}{N} \sum_{i=1}^N P_i(t) \\ &= \left(\frac{1}{N} \sum_{i=1}^N P_{i,1}(t), \frac{1}{N} \sum_{i=1}^N P_{i,2}(t), \dots, \frac{1}{N} \sum_{i=1}^N P_{i,N}(t) \right), \end{aligned} \quad (7)$$

where α is contraction-expansion coefficient. It is the only additional control parameter in QPSO besides the population size and the number of iterations. α can be set by fixed values or decreases linearly.

3. DWC-QPSO Algorithm

3.1. DWC-QPSO Algorithm. QPSO ensures global convergence; however, this algorithm still has the local optimum problem. The main reason for that is that QPSO algorithm establishes a single δ potential well at the point p_i , and the population of particles gradually approaches to p_i under the gravity function. While this information exchange method is fast, the transmission is unidirectional, leading to the population of particles quickly gather in an increasingly smaller search area and resulting in low population diversity. In order to improve the performance of QPSO algorithm, this paper proposes the DWC-QPSO algorithm, which enhances the QPSO's convergence properties by increasing wait effects between particles to avoid population excessive accumulation.

The specific idea of DWC-QPSO algorithm is as follows: dividing the particle swarm into two separate subgroups after randomizing the individual's position in the solution space. One subgroup establishes a δ potential well at the position of p_i in accordance with the rules of standards QPSO algorithm. The particles of the second subgroup are iteratively searching in the solution space under the influence of the other δ potential well which is established with p'_i as the center. Given the entire population S , the first subgroup is called master subgroup, represented by S_1 , and the second subgroup is called secondary subgroup, represented by S_2 ; thus, $S_1 \cup S_2 = S$. The position of p'_i is calculated as in formula (8).

Evolutionary formula of particles' positions in DWC-QPSO algorithm is as follows:

$$p_{i,j}(t)' = [1 - \varphi_{i,j}(t)] P_{i,j}(t) + \varphi_{i,j}(t) G_j(t), \quad (8)$$

$$\begin{aligned} X_{i,j}(t+1) &= p_{i,j}(t) \pm \alpha \cdot |C_j(t) - X_{i,j}(t)| \\ &\quad \times \ln \left[\frac{1}{u_{i,j}(t)} \right] \quad i \in N_{S_1}, \\ X_{i,j}(t+1) &= p_{i,j}(t)' \pm \alpha \cdot |C_j(t) - X_{i,j}(t)| \\ &\quad \times \ln \left[\frac{1}{u_{i,j}(t)} \right] \quad i \in N_{S_2}. \end{aligned} \quad (9)$$

These two subgroups exchange information by means of the best individual during the searching period in DWC-QPSO algorithm that is comparing the best fitness value of S_1 and S_2 at the end of each iteration. Given that g_{best1} is S_1 's global best solution and g_{best2} is S_2 's global best solution, we compare the fitness values corresponding to g_{best1} and g_{best2} , respectively. If the latter one is superior than the former one, g_{best2} is assigned to g_{best1} ; otherwise g_{best1} remains unchanged. The essence of the above operation is an update of the whole swarm's global best solution g_{best} to help subgroups escape from local optima. By analyzing the displacement formula, the result shows that p_i and p'_i are symmetry in point of the center of p_{best} and g_{best} . In the convergence process, all particles' p_{best} will gradually move closer to the g_{best} . Due to the gravity of δ potential well, p_i of master subgroup and p'_i of secondary subgroup will overlap at the same place in the solution space eventually. Therefore, the convergence of the DWC-QPSO algorithm can be achieved.

3.2. Experiment and Simulation. In order to verify the novel algorithm, the optimization results of DWC-QPSO algorithm are compared with the performance of standard PSO algorithm and QPSO algorithm using 6 standard test functions. Since many practical projects, including single-peak and multi-peak problems, choose two single peak functions (Sphere function and Rosenbrock function) and five multi-peak functions (Rastrigin function, Griewank function, Ackley function, Schaffer function, and Schwefel function) as the experimental test functions, we will give some information about these test functions, including expressions, the ranges of variables, dimensions, the optimal solutions, and optimal values of test functions as follows:

(1) Sphere function:

$$\begin{aligned} f_1(X) &= \sum_{i=1}^D x_i^2, \\ x_i &\in [-100, 100], \quad D = 20, \quad \min f_1(0, 0, \dots, 0) = 0, \end{aligned} \quad (10)$$

(2) Rosenbrock function:

$$\begin{aligned} f_2(X) &= \left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right], \\ x_i &\in [-10, 10], \quad D = 20, \quad \min f_2(1, 1, \dots, 1) = 0, \end{aligned} \quad (11)$$

(3) Rastrigin function:

$$\begin{aligned} f_3(X) &= \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10), \\ x_i &\in [-5.12, 5.12], \quad D = 20, \quad \min f_3(0, 0, \dots, 0) = 0, \end{aligned} \quad (12)$$

(4) Griewank function:

$$\begin{aligned} f_4(X) &= \sum_{i=1}^D \frac{x_i^2}{4000} - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, \\ x_i &\in [-600, 600], \quad D = 20, \quad \min f_4(0, 0, \dots, 0) = 0, \end{aligned} \quad (13)$$

(5) Ackley function:

$$\begin{aligned} f_5(X) &= -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) \\ &\quad - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i)\right) + 20 + e, \\ x_i &\in [-32.786, 32.786], \quad D = 20, \quad \min f_5(0, 0, \dots, 0) = 0, \end{aligned} \quad (14)$$

TABLE 1: The results of optimization algorithms ($D = 20, N = 30$).

Algorithm		f_1	f_2	f_3	f_4	f_5	f_6
PSO	FV	1.3782E - 286	1.5113E001	1.1550E001	4.1815E - 002	1.6991E - 011	4.2418E003
	T (s)	0.8379	0.1753	0.1965	0.1992	0.2147	0.1797
QPSO	FV	1.3872E - 287	1.5928E001	1.0945E001	4.1815E - 002	1.7319E - 011	4.2317E003
	T (s)	0.8297	0.1766	0.2031	0.1984	0.2188	0.1797
DWC-QPSO	FV	1.0461E - 313	1.5672	5.9996	1.2431E - 002	1.8385E - 013	1.4243E003
	T (s)	0.7563	0.1203	0.1328	0.1216	0.1641	0.1547

TABLE 2: The results of optimization algorithms ($D = 20, s = 50$).

Algorithm		f_1	f_2	f_3	f_4	f_5	f_6
PSO	FV	2.1469E - 289	1.3941E001	10.1020	3.7234E - 002	2.1537E - 014	4.1132E003
	T (s)	1.2889	0.2987	0.2966	0.3289	0.3517	0.2655
QPSO	FV	2.0941E - 289	1.4351E001	9.6164	3.6314E - 002	2.0428E - 014	4.0053E003
	T (s)	1.2984	0.3047	0.2938	0.3391	0.3422	0.2641
DWC-QPSO	FV	7.1978E - 313	9.8871	5.9698	4.0719E - 006	1.3323E - 014	9.4751E002
	T (s)	1.2906	0.2062	0.2062	0.2625	0.2641	0.2484

(6) Schwefel function:

$$f_6(X) = 418.9829D - \sum_{i=1}^D x_i \sin(x_i^{1/2}),$$

$$x_i \in [-500, 500], \quad D = 20,$$

$$\min f_7(420.98, 420.98, \dots, 420.98) = 0.$$

In the experiment simulation, the parameter α linearly decreases from 1 to 0.5. The maximum number of iterations *iterMax* is set to 1000. The population sizes are set to 30 and 50, respectively. The size of the master cluster is N_{S_1} and the size of the secondary cluster is N_{S_2} , using the average distribution strategy, which is $N_{S_1} = N_{S_2}$. Each test case runs 30 times independently, then calculate the average results of the test function. Experimental environment is MATLAB 7.9.0, Intel Core (TM) 2, 1.80 GHz, 2.50 GB RAM.

Mean fitness value and CPU time per round of PSO, QPSO, and DWC-QPSO algorithm to optimize the various functions are as shown in Tables 1 and 2.

Tables 1 and 2 illustrate that the DWC-QPSO has the best performance compared with the PSO algorithm and the QPSO algorithm. This is because the DWC-QPSO algorithm has two subgroups, which are master subgroup S_1 and secondary subgroup S_2 , respectively. S_1 and S_2 complement each other and coevolve by exchanging the optimal location information constantly.

This pattern of two subgroups searching in solving space at the same time and learning from each other increases the probability of finding the optimal solution and improves the convergence performance of the new algorithm.

Figures 1(a) and 1(b) show the convergence curves of the PSO, QPSO, and DWC-QPSO algorithm solving 20 dimensions of unimodal functions with 30 particles, respectively. Sphere function is relatively simple, which can be used to test the accuracy of optimization algorithms to

study the implementation of the algorithm performance. It is learned from Figure 1(a) that DWC-QPSO has the similar performance. Rastrigin's function is difficult to optimize as it has a large number of local optimums. The novel one gets the best results of these three algorithms for solving this function.

Figures 1(c)~1(e) illustrate the convergence curves of the PSO, QPSO, and DWC-QPSO algorithm solving 20 dimensions of multimodal functions with 30 particles, respectively. Rastrigin function is easy to fall into local minima. The pattern of dural subgroups interaction allows DWC-QPSO algorithm to have a better ability to escape from local optima to get the better optimization. From Figures 1(d) to 1(f) are the curves of Griewank function, Ackley function, and Schwefel function. There are correlations between the variables of these three functions. The experiment results show that the DWC-QPSO algorithm has better performance in this type of problem.

In summary, the DWC-QPSO algorithm proposed by this study obtained outstanding global search capability in limited iterative steps because it could get more useful information.

4. The On-Ramp PI Controller Based on DWC-QPSO Algorithm

4.1. On-Ramp Traffic Model. Consider that the on-ramp has one lane, and the main lane of freeway is divided into M sections with length L_i . The traffic model is shown in Figure 2, where $i = 1, 2, \dots, M$.

The traffic flow model in the i th section of freeway is discrete in both space and time domains. It includes the following variables:

$\sigma_i(k)$: the traffic flow density of i th section during the k th period;

$v_i(k)$: the average vehicle speed in i th section during the k th period;

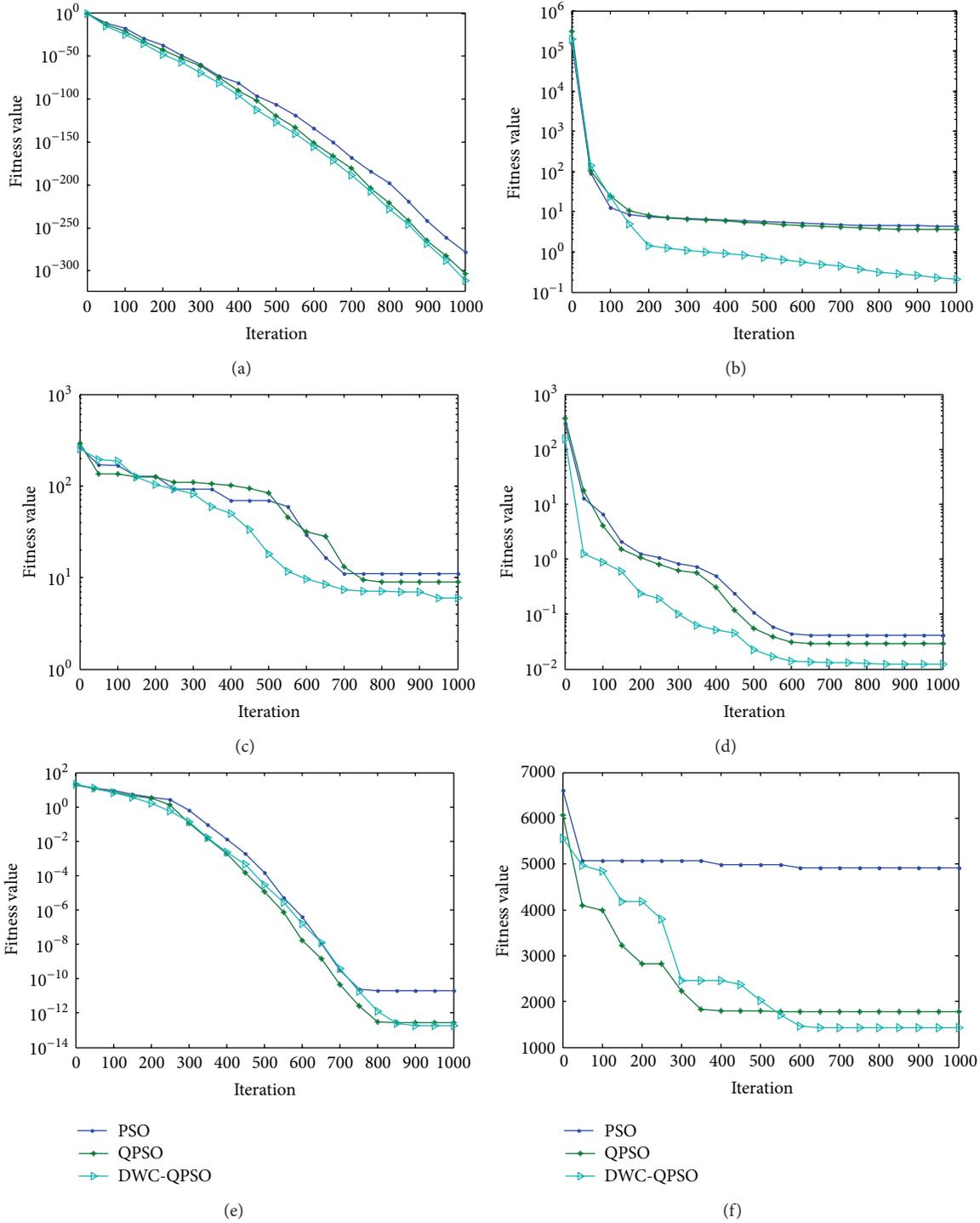


FIGURE 1: The convergence curves of test functions: (a) f_1 , (b) f_2 , (c) f_3 , (d) f_4 , (e) f_5 , and (f) f_6 .

$q_i(k)$: the increased traffic flow of $i + 1$ th section from i th segment during the k th period;

$r_i(k)$: the on-ramp metering rate of i th section during the k th period;

$N_i(k)$: the total number of vehicles in i th section during the k th period;

λ : the number of main lanes of freeway;

L_i : the length of the i th section;

T : the discrete time steps.

According to the law of vehicles conservation, the total number of vehicles in i th section during the $(k + 1)$ th period is as follows:

$$N_i(k + 1) = N_i(k) + T [\lambda q_{i-1}(k) - \lambda q_i(k) + r_i(k)]. \quad (16)$$

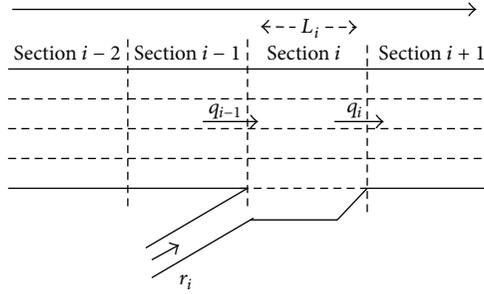


FIGURE 2: On-Ramp traffic model of freeway.

The traffic density of i th section can be calculated through (9) in line with the definition of the traffic density:

$$\sigma_i(k) = \frac{N_i(k)}{\lambda L_i}. \quad (17)$$

Then,

$$\sigma_i(k+1) = \sigma_i(k) + \frac{T}{\lambda L_i} \left[q_{i-1}(k) - q_i(k) + \frac{r_i(k)}{\lambda} \right]. \quad (18)$$

The parabolic flow-density model can be deduced:

$$q_i(k) = v_f \sigma_i(k) \left(1 - \frac{\sigma_i(k)}{\sigma_{\text{jam}}} \right) = v_f \sigma_i(k) - \frac{v_f \sigma_i(k)^2}{\sigma_{\text{jam}}}. \quad (19)$$

After the combination of (16) and (17), the complete traffic flow of the on-ramp of freeway is shown as formula (18):

$$\begin{aligned} \sigma_i(k+1) &= \sigma_i(k) + \frac{T}{\lambda L_i} \\ &\times \left[q_{i-1}(k) + \frac{r_i(k)}{\lambda} - v_f \sigma_i(k) - \frac{v_f \sigma_i(k)^2}{\sigma_{\text{jam}}} \right], \end{aligned} \quad (20)$$

where v_f is the free-flow speed and σ_{jam} is the jamming density of the freeway. In fact, the flow-density model also creates a critical density σ_c corresponding to the maximum flow rate q_m . $q_i(k)$ increases with the increase of $\sigma_i(k)$ value in $[0, \sigma_c]$ interval, while $q_i(k)$ decreases with the increase of $\sigma_i(k)$ value in $[\sigma_c, \sigma_{\text{jam}}]$ interval. When $\sigma_i(k)$ is equal to σ_{jam} , the $q_i(k)$ is 0; therefore, the traffic jam occurs.

4.2. The On-Ramp PI Controller Based on DWC-QPSO. As a feedback control system, PI controller becomes one of the main methods of engineering controls because of its simple structure, good stability, reliability, and easy adjustment. Performance of PI controller depends on the reasonable values of parameters K_p and K_i . At present, PI controller parameters rely mainly on manual adjustment. However, this traditional approach is not only time consuming but also unable to guarantee the best performance. In this paper, DWC-QPSO algorithm is used to optimize the value of K_p and K_i , which is shown in Figure 3.

PI controller system in Figure 3 includes the following variables:

$$\text{error value: } e_i(k) = \sigma_{di}(k) - \sigma_i(k);$$

$$\text{error variation: } \Delta e_i(k) = e_i(k) - e_i(k-1);$$

$$\text{the output of the PI controller: } \Delta r_i(k) = K_p \Delta e_i(k) + K_i e_i(k);$$

$$\text{mediation rate of the on-ramp: } r_i(k) = r_i(k-1) + \Delta r_i(k);$$

$$\text{expected traffic density: } \sigma_{di}(k) = \sigma_c - \varepsilon, \varepsilon \text{ is an appropriate small positive number.}$$

This system's objective is to control the mediation rate of the on-ramp $r_i(k)$ and avoid the traffic congestion through maintaining the traffic density of main lane $\sigma_{di}(k)$ at a negative neighborhood of critical density σ_c . Therefore, the square sum of the differences between actual traffic flow density $\sigma_i(k)$ and expected traffic density $\sigma_{di}(k)$ can be used as the system objective function:

$$J = \sum_{k=1}^n (e_i(k))^2 = \sum_{k=1}^n (\sigma_i(k) - \sigma_{di}(k))^2. \quad (21)$$

Each particle of the swarm corresponds to a pair of values of parameters K_p and K_i when using the DWC-QPSO algorithm optimizing the PI controller. Due to the fact that the square sum of the differences between actual traffic flow density $\sigma_i(k)$ and expected traffic density $\sigma_{di}(k)$ is larger, the corresponding adjustment of on-ramp rate $r_i(k)$ value should be smaller and the DWC-QPSO adaptation algorithm corresponding value should also be smaller; the reciprocal of the sum of squared deviations can be selected as the fitness function of the DWC-QPSO algorithm:

$$\min F = \frac{1}{\sum_{k=1}^n (e_i(k))^2} = \frac{1}{\sum_{k=1}^n (\sigma_i(k) - \sigma_{di}(k))^2}. \quad (22)$$

Figure 4 illustrates the process of using DWC-QPSO algorithm optimizing the PI controller parameters.

The specific process is as follows.

Step 1 (setting parameters). This includes the acceleration factors c_1 and c_2 , the contraction-expansion factor α , the swarm population N , the maximum allowed times of iterations $iterMax$ or error accuracy, and the population of subgroups, where N_{S1} and N_{S2} are the population of master subgroup S_1 and secondary subgroup S_2 , respectively.

Step 2 (population initialization). For S_1 , randomly generate the particles' initial positions $X_{i,j}(0)$ in the solution space, and let $P_{i,j}(0) = X_{i,j}(0)$, where $i = \{1, \dots, N_{S1}\}$, $j = \{1 \dots D\}$; for S_2 , do the same operation with S_1 , generate the particles' random initial positions $X_{i,j}(0)$, and let $P_{i,j}(0) = X_{i,j}(0)$, where $i \in \{N_{S1} + 1, \dots, N\}$, $j = \{1 \dots D\}$.

Step 3. Calculate the fitness value of all particles in S_1 and S_2 according to (22). The particle location of the smallest fitness value is assigned to the global best solution; that is, $\min(f(X_i)), i \in \{1, \dots, N\} \rightarrow g_{\text{best}}$.

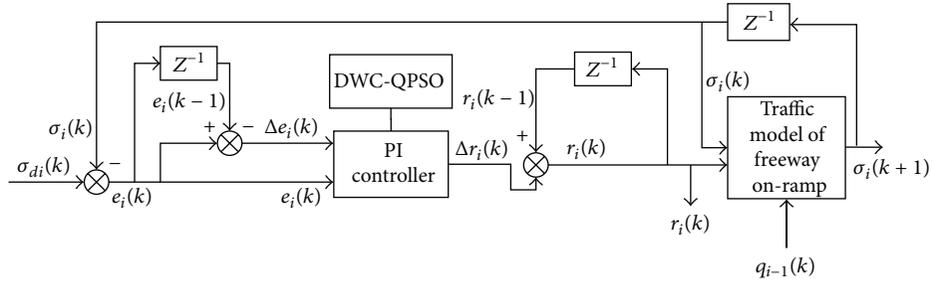


FIGURE 3: On-ramp PI controller based on DWC-QPSO.

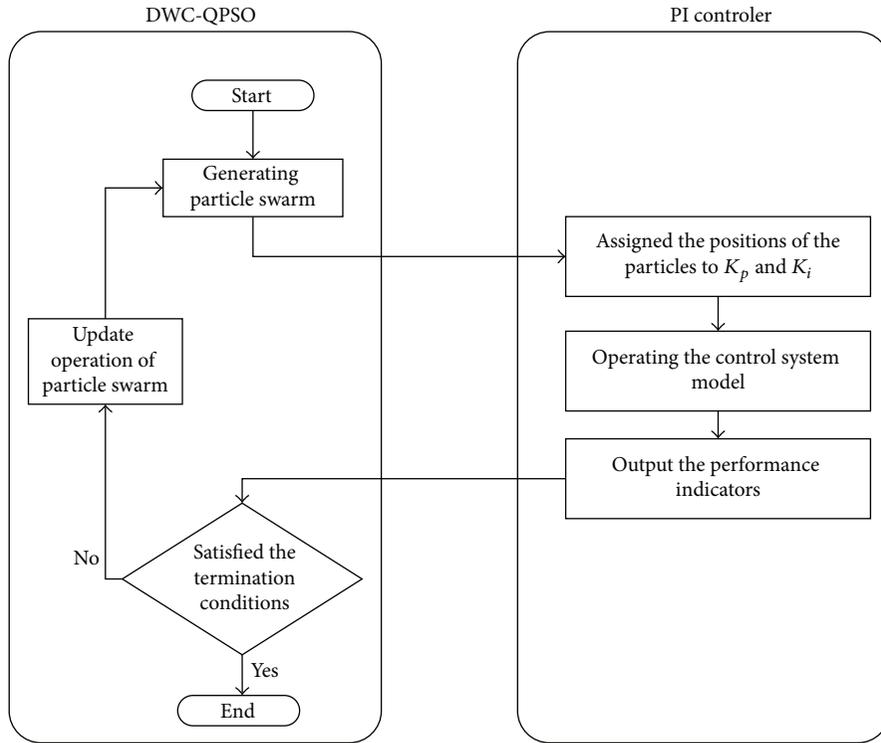


FIGURE 4: The process of optimizing on-ramp PI controller using DWC-QPSO algorithm.

Step 4. Calculate the mean best position $C(t)$ of the particle swarm based on (7), using it to evaluate $L_{i,j}(t)$, where i is the length of δ potential well.

Step 5. According to (9), update the location of particle i ($1 \leq i \leq N$).

Step 6. Recalculate the fitness value of the particle i 's new position in the solution space according to the objective function.

Step 7. Update the locations of all particles' $P_i(t)$. The specific operation is comparing the fitness value of $X_i(t)$ and $P_i(t-1)$; if $f(X_i(t)) < f(P_i(t-1))$, do $P_i(t) = X_i(t)$; otherwise do $P_i(t) = P_i(t-1)$.

Step 8. Compare all particles' fitness value with $P_g(t-1)$'s fitness value. If $i \in S_1$ and $f(P_i(t)) < f(P_g(t-1))$, save i 's

position as the global best position of subgroup S_1 , denoted by $P_{gS1}(t)$; otherwise do the operation of $P_g(t-1) \rightarrow P_{gS1}(t)$. If $i \in S_2$ and $f(P_i(t)) < f(P_g(t-1))$, save i 's position as the global best position of subgroup S_2 ; otherwise do the operation of $P_g(t-1) \rightarrow P_{gS2}(t)$.

Step 9. Compare the fitness value of $P_{gS1}(t)$ with the fitness value of $P_{gS2}(t)$. If $f(P_{gS1}(t)) > f(P_{gS2}(t))$, $P_{gS2}(t)$ would be the global best position of the entire population $P_g(t)$; that is, $P_{gS2}(t) \rightarrow P_g(t)$; otherwise, do the operation of $P_{gS1}(t) \rightarrow P_g(t)$.

Step 10. Determine the termination conditions of the algorithm. If the end conditions are not reached (usually the end condition is the expected fitness value or the maximum number of iterations), do $t = t + 1$ and return to Step 3.

Step 11. Output the optimal solution.

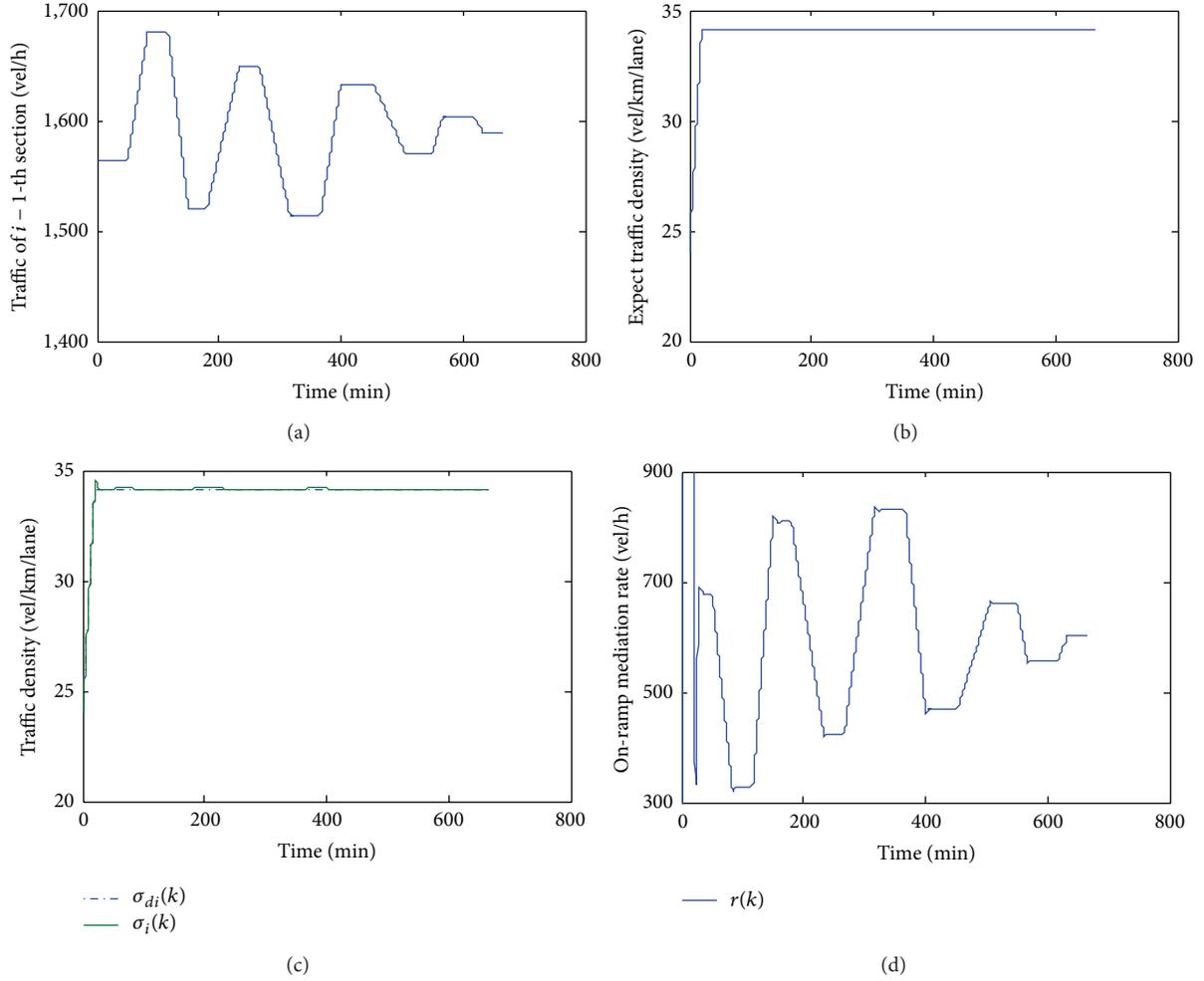


FIGURE 5: Comparison of actual and expected traffic density: (a) traffic flow of $i-1$ section; (b) expected traffic density; (c) traffic density comparison; (d) on-ramp mediation rate.

5. On-Ramp PI Controller Simulation and Results Analysis

In order to demonstrate the effectiveness of the new algorithm, an experiment for on-ramp PI controller based on DWC-QPSO algorithm is conducted. In the simulation, the acceleration coefficients of DWC-QPSO c_1 and c_2 are set to be 2, and the contraction-expansion factor α decreases linearly from 1 to 0.5. The maximum number of iteration steps $itemMax$ is 150 and the entire population size N is 30. All particles are evenly distributed to two subgroups, which means that the population size of master subgroup N_{s1} is equal to the population size of secondary subgroup N_{s2} . The system prototype is implemented in MATLAB 7.9.0. Typical parameters of on-ramp traffic flow model are shown in Table 3.

Assume the curve of the traffic $q_{i-1}(k)$ is shown in Figure 5(a). And Figure 5(b) illustrates that the expected traffic density of i th section $\sigma_{di}(k)$ linearly increases from the initial 24.06 vel/km·lane⁻¹ to 34.16 vel/km·lane⁻¹, and since then it has been maintained at 34.16 vel/km·lane⁻¹. The values of parameters of the on-ramp traffic PI controller optimized by

TABLE 3: Parameters of on-ramp traffic flow model.

v_f	σ_{jam}	σ_c	λ	T
97.3 km/h	74 vel/km·lane ⁻¹	37 vel/km·lane ⁻¹	3	20 s

DWC-PSO algorithm are $K_p = 186.6008$ and $K_i = 330.000$, which corresponds to the minimum fitness value $J = 2.054$. After using on-ramp traffic PI controller under the above parameter settings, the curves of actual traffic flow density of lane's i th section and the on-ramp mediation rate are shown in Figures 5(c) and 5(d), respectively.

It is learned from Figure 5(c) that the error between $\sigma_i(k)$ and $\sigma_{di}(k)$ is very small, which means that the expected traffic density can be well tracked by the actual traffic density in the system using DWC-QPSO algorithm optimization. As can be seen from Figure 5(d), the on-ramp mediation rate increases with time at the beginning, and there is a surge during this period. Since the traffic of $(i-1)$ th section of lane remains unchanged in the initial stage, while the expected traffic density is linearly growing from 24.06 vel/km·lane⁻¹ to

TABLE 4: Parameters of PI controller and the objective function values.

Algorithms	K_p	K_i	J
DWC-QPSO	186.6008	330.000	0.2054
PSO [2]	184.4559	330.000	0.25083
ACO [1]	183.2129	329.0433	0.25198.

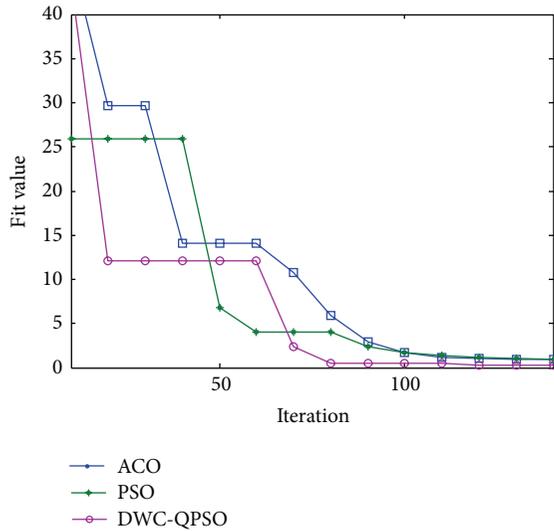


FIGURE 6: Convergence curve of algorithms.

34.16 vel/km-lane⁻¹, then traffic increment is mainly coming from the on-ramp. In the following stage, the expected traffic density remains at vel/km-lane⁻¹, while the on-ramp mediation rate changes with the traffic variation of ($i-1$)th section. The on-ramp mediation rate would be decreasing and the vehicles entering to the main lane from on-ramp would be reducing when the traffic of $i-1$ section is increasing; however, when the traffic of $i-1$ section is reducing, the on-ramp mediation rate would be increasing and the vehicles entering to the main lane from on-ramp would be increasing as well. The on-ramp control strategies above could avoid traffic congestion, keeping the traffic smooth and maximizing the utilization of the main lane.

In order to verify the effectiveness of the algorithm, the optimization performances of ant colony optimization (ACO) algorithm and particle swarm optimization (PSO) algorithm are compared with the performance of DWC-QPSO. The PI controller parameters and the optimal value of objective function corresponding to these three algorithms are shown in Table 4, and the curve of the fitness value is shown in Figure 6.

The simulation illustrates that the minimum fitness value of on-ramp PI controller system can be obtained by DWC-QPSO in these three algorithms. The results reveal that the actual traffic flow density $\sigma_i(k)$ is closest to the expected traffic density $\sigma_{di}(k)$ and the system has the highest stabilities under this condition. As a real-time traffic control system, if the reaction is too slow, the input information would be useless for the system and the outputs would lose effectiveness for inducing actual traffic. Therefore, the timeliness is an

important evaluation standard for optimization algorithms in real-time applications. It can be learned that the DWC-QPSO algorithm has the fastest convergence speed comparing with the ant colony algorithm and the standard particle swarm optimization from Figure 6. Because the novel algorithm has high-speed response capabilities in avoiding cumbersome and complex human intervention, it is able to quickly find the optimal value of parameters K_p and K_i . In summary, because the system is capable of adjusting the on-ramp traffic dynamically in real-time according to the traffic condition of upper section, the DWC-QPSO algorithm is quite applicable in on-ramp traffic PI controller and outperforms other evolutionary algorithms with enhancement in both adaptability and stability.

6. Conclusion

On-ramp control is a significant way to control the traffic density of freeway. Furthermore, it is also an important foundation for the traffic guidance systems and other decision-making systems in ITS. In this paper, the on-ramp traffic PI controller based on the DWC-QPSO algorithm is proposed, which is designed on the assumption that all sections of lanes have the abilities of measuring the average vehicle speed and the traffic density. The DWC-QPSO algorithm is applied in optimizing the PI controller parameters K_p and K_i in order to achieve the traffic density control at the on-ramp. Experimental results show that the on-ramp traffic PI controller based on DWC-QPSO algorithm has better adaptability and stability compared to the other evolutionary algorithms.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This paper is supported by the National Nature Science Foundation of China (no. 61104175) and Fundamental Research Funds for the Central Universities (no. 2014NZYQN24). Tao Wu, Yusong Yan, and Xi Chen would like to thank everyone who contributed towards the completion of this paper.

References

- [1] Q. X. Lv and X. R. Liang, "Freeway on-ramp control based on ant colony algorithm," in *Measurement & Control Technology*, vol. 10, pp. 61–69, 2011.
- [2] Y. K. Fan and X. R. Liang, "Particle swarm optimization based PI controller for freeway ramp metering," in *Proceedings of the 27th Chinese Control Conference*, pp. 503–506, Kunming, China, July 2008.
- [3] J. X. Xu, S. H. Yu, and L. J. Xu, "Adaptive coordinated control based on FNN for freeway entering ramp," *Journal of Southeast University (Natural Science Edition)*, vol. S1, pp. 266–271, 2009.
- [4] L. Lei, L.-Y. Dong, and H.-X. Ge, "Study of gear-alternating control regulation in the interfluent location of on-ramps with cellular automaton model," *Acta Physica Sinica*, vol. 56, no. 12, pp. 6874–6880, 2007.

- [5] B. L. Li and S. R. Qu, "Adaptive control of on-ramp metering for highway based on genetic-fuzzy approach," *Measurement & Control Technology*, vol. 5, pp. 73–80, 2011.
- [6] W.-S. Su, Y. Li, J.-G. Sun, and W. Zhou, "Optimization methodology of PI controller parameters for aeroengines based on genetic algorithm," *Journal of Aerospace Power*, vol. 20, no. 6, pp. 1078–1082, 2005.
- [7] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, December 1995.
- [8] R. Eberhart and J. Kennedy, "New optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, pp. 39–43, IEEE Press, Nagoya, Japan, October 1995.
- [9] Q.-J. Ni, Z.-Z. Zhang, Z.-Z. Wang, and H.-C. Xing, "Dynamic probabilistic particle swarm optimization based on varying multi-cluster structure," *Journal of Software*, vol. 20, no. 2, pp. 339–349, 2009.
- [10] W. Jiao and G.-B. Liu, "Two subpopulation swarm PSO algorithm in dynamic environment," *Control and Decision*, vol. 24, no. 7, pp. 1083–1091, 2009.
- [11] F. V. D. Bergh, *An analysis of particle swarm optimizers [Ph.D. thesis]*, University of Pretoria, 2002.
- [12] J. Sun, *Particle Swarm Optimization with Particles Having Quantum Behavior*, Jiangnan University.
- [13] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

Research Article

The Inertia Weight Updating Strategies in Particle Swarm Optimisation Based on the Beta Distribution

Petr Maca and Pavel Pech

Department of Water Resources and Environmental Modeling, Faculty of Environmental Sciences, Czech University of Life Sciences Prague, Kamycka 1176, Praha 6, 165 21 Suchbát, Czech Republic

Correspondence should be addressed to Petr Maca; maca@fzp.czu.cz

Received 6 June 2014; Revised 21 August 2014; Accepted 8 September 2014

Academic Editor: Fang Zong

Copyright © 2015 P. Maca and P. Pech. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The presented paper deals with the comparison of selected random updating strategies of inertia weight in particle swarm optimisation. Six versions of particle swarm optimization were analysed on 28 benchmark functions, prepared for the Special Session on Real-Parameter Single Objective Optimisation at CEC2013. The random components of tested inertia weight were generated from Beta distribution with different values of shape parameters. The best analysed PSO version is the multiswarm PSO, which combines two strategies of updating the inertia weight. The first is driven by the temporally varying shape parameters, while the second is based on random control of shape parameters of Beta distribution.

1. Introduction

The particle swarm optimisation—PSO—is a popular heuristic optimisation algorithm developed by Kennedy and Eberhart [1]. It is a nature inspired heuristic, which mimics the behaviour of flocks of birds or schools of fish. The recent survey of variants of PSO can be found in [2]. It is a population based evolutionary technique [3, 4], its introductory description is provided in [5]. The PSO has been successfully applied to many real life optimisation problems [6, 7].

Recently the PSO oriented research focuses on the development of new adaptation strategies, which avoid the premature convergence of particle population, or being trapped in local optima. For example the periodic changes of number of particles in population enhance the PSO performance [8]. The adaptive tuning of velocity particle estimated by the average velocity information accelerates the PSO ability to jump out of local optima [9]. Hu et al. [10] developed adaptive variant of PSO called PSO-MAM, which adopts the subgradient method for adjusting the PSO parameters. Liu et al. [11] applied in CPSO—chaotic particle swarm optimisation—the logistic equation for adjusting the new location of particles.

The improvement the estimation of particle's velocity is an essential task in PSO research. It was shown that the

inertia weight—IW—helps to increase the overall PSO search performance [4, 12]. Nickabadi et al. [13] provide the overview of 15 different strategies for the inertia weights adaptation.

The random adaptations of inertia weight play an important role in improving the PSO performance [4, 11, 12, 14, 15]. Mostly they support the exploratory search in the beginning of optimisation process. They increase the population diversity during the search process. Bansal et al. [14] compared 15 different IW strategies on 5 optimisation problems. The linear decreasing inertia term with logistic mapping was the best IW strategy in terms of average error. The logistic mapping of form $x[i + 1] = 4x[i](1 - x[i])$ is random number generator related to the symmetric Beta distribution with parameters $a = 0.5$ and $b = 0.5$ [16].

Besides the adaptation strategies of PSO parameters the special attention has to be put on development the multiswarm PSO [17–19]. The multiswarm PSO based on exclusion and anticonvergence was tested in dynamic environments [20]. The master slave multiswarm models with competitive and collaborative versions, in which the slave swarm provides the master swarm with the best particle, were studied in [17]. The cooperative multiswarm PSO of four swarms with cooperative search and diversity strategy performed better than single PSO on complex multimodal functions [21]. The five swarms with constant period of migration and

constant migration rate outperformed single PSO on eight optimisation problems [18].

The comparison study of 12 different migration strategies 6 on 36 optimisation problems is provided in [19]. Two migration strategies BW and BWM—the BWM applied the mutation on migrating particles—based on migrating the selected number of best particles from subswarm and substituting with them the worst particles outperformed remaining migration models. The parallel PSO with three communication strategies is compared in work of Chang et al. [22]. All three migration strategies are applied sequentially in one optimisation run and periodically exchange the particles between subswarms.

The aim of the presented paper is to compare selected version of PSO. The tested single and multiswarm versions of particle swarm optimisation are based on modifications of inertia weight, which are related to the random component controlled by the Beta distribution.

The remaining part of paper is arranged as follows. The description of PSO provides details on standard PSO, the proposed random inertia weight strategies, and the description of tested multiswarm PSO. Results comment on the finding based on extensive 10 dimensional computational experiments. The article summarizes the main findings in Conclusions.

2. The Description of PSO

2.1. The Standard PSO. The standard PSO (sPSO) modifies the location of particle $X^j = \{x_1^j, \dots, x_{\dim}^j\}$ with dimension \dim using the velocity $V^j = \{v_1^j, \dots, v_{\dim}^j\}$ updated in generation t as

$$\begin{aligned} v_i^j[t] = & w[t] v_i^j[t-1] + c_1 r_1 (g_i - x_i^j[t-1]) \\ & + c_2 r_2 (p_i^j[t-1] - x_i^j[t-1]) \end{aligned} \quad (1)$$

for $i = 1, \dots, \dim$,

where the $r_1 \sim U(0, 1)$ and $r_2 \sim U(0, 1)$ are random numbers with uniform distribution, c_1, c_2 denote the acceleration coefficients of social and cognitive learning, and the $w[t]$ is the inertia weight.

The new location of particle is computed as

$$x_i^j[t] = x_i^j[t-1] + v_i^j[t] \quad \text{for } i = 1, \dots, \dim, \quad (2)$$

and the social component is controlled by the location of the global best particle denoted as $G = \{g_1, \dots, g_{\dim}\}$. For solving the minimization problem based on fitness function $f(X)$ is the $f(G) \leq f(X^j)$ for all X^j in population. The cognitive learning component is represented by the personal best location of particle $P^j = \{p_1^j, \dots, p_{\dim}^j\}$, which is $f(P^j) \leq f(X^j)$ for all actually known locations of particle $X^j[t]$. Equations (1) and (2) are applied on all X^j for $j = 1, \dots, n_{\text{pop}}$ with n_{pop} is the number of particles in swarm population [1, 3–5].

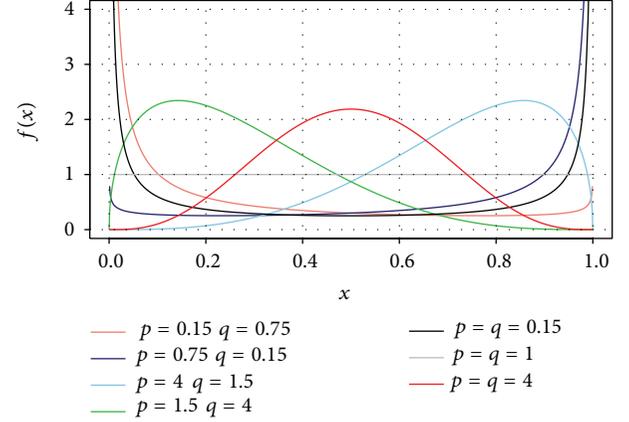


FIGURE 1: The selected densities of Beta distribution.

The sPSO is based on the velocity update with the linear decreasing inertia term $w[t]$, calculated with the formula

$$w[t] = (w_{\max} - w_{\min}) \frac{t_{\max} - t}{t_{\max}} + w_{\min}, \quad (3)$$

where the w_{\max} was set to 0.9, w_{\min} is equal to 0.4, and t_{\max} is the maximum number of generations.

The velocity update formula is restricted by v_{\max} , and it is applied as velocity control on the cases, when $|v_i^j| > v_{\max}$. Then the value of velocity is bounded on

$$v_i^j = \begin{cases} v_{\max} & \text{for } v_i^j > v_{\max} \\ -v_{\max} & \text{for } v_i^j < -v_{\max} \end{cases} \quad (4)$$

Note that this type of velocity control only enables limiting the maximum distance in which particle may move during one iteration [1, 23, 24].

2.2. The Proposed Inertia Weight Modifications. The proposed inertia weight modifications are based on random numbers generated using the Beta distribution. The density of Beta distribution $f(x)$ is defined as

$$f(x) = \begin{cases} \frac{1}{B(p, q)} x^{p-1} (1-x)^{q-1} & \text{for } 0 \leq x \leq 1 \\ 0 & \text{for } (0 > x) \wedge (x > 1). \end{cases} \quad (5)$$

Figure 1 shows selected densities of Beta distribution with different values of shape parameters p and q . The Beta distribution allows simulation from symmetric densities ($p = q$) and asymmetric densities with shape parameters $p \neq q$. Note that the uniform distribution is a special case of Beta distribution $p = q = 1$, and it has the maximum entropy from all Beta distributions.

One of the main advantages of Beta distribution is that it describes probability densities with various shapes on the interval $(0, 1)$. For equal shape parameters $p = q > 1$ the density is bell shaped, for $p = q > 1$ is U shaped. The U shaped densities allow simulating the extremes on interval

TABLE 1: Tested inertia weight updates based on Beta distribution.

PSO version	The weight update formula	Random component
RBld	$w[t] = \left((w_{\max} - w_{\min}) \frac{t_{\max} - t}{t_{\max}} + w_{\min} \right) - \frac{rb}{2}$	$rb \sim \text{Beta}(bs[t], bs[t])$
RBrr	$w[t] = \left((w_{\max} - w_{\min}) \frac{t_{\max} - t}{t_{\max}} + w_{\min} \right) - \frac{rb}{2}$	$rb \sim \text{Beta}(r_1, r_2)$
RBRa	$w[t] = \left((w_{\max} - w_{\min}) \frac{t_{\max} - t}{t_{\max}} + w_{\min} \right) + rb \cdot w_{\min}$	$rb \sim \text{Beta}(r_1, r_2)$

$\langle 0, 1 \rangle$, while the bell shaped ones are focused on center of interval. This property supports the balanced exploratory and exploitative search process and avoids the premature convergence.

Table 1 shows definitions of three tested inertia weight strategies based on the Beta distribution. The RBld represents the linearly decreasing inertia weight with random component based on symmetric Beta distribution with linearly varied shape parameters $bs[t]$. The $bs[t]$ are controlled by the iteration t and are expressed as

$$bs[t] = (bs_{\max} - bs_{\min}) \frac{t - 1}{t_{\max} - 1} + bs_{\min}, \quad (6)$$

where the $bs[t] = p = q$ represents the shape parameters for symmetrical Beta distribution, which are applied on random number generation in time t .

The RBrr inertia weight version applies randomly selected shape parameters $r_1 \sim U(0, 1)$ and $r_2 \sim U(0, 1)$. The simulated random component for one generation consists mainly of random numbers generated from different asymmetrical Beta distributions. Note that the probability that $r_1 = r_2$ is smaller than the probability that $r_1 \neq r_2$.

The RBRa is modification of original of logistic mapping [11, 12]. The noise generated by the Beta distribution random generator is added to linearly varied inertia weight. The randomly varied shape parameters enable generation from both symmetrical and asymmetrical Beta distributions.

2.3. The Multiswarm PSO. The new proposed multiswarm PSO combines the search of four subswarms. This PSO version is marked as BrBl. The algorithm follows the principles of multiswarm algorithms [17–19], and it is completed by migration principle. The subswarms are divided into the two groups: the cooperative subswarms and elitistic subswarm. The subswarms use different inertia weight Beta distribution strategies. They share the information about global best particle only through the migration process.

The migration period is controlled by the simple rule, which increases the number of generations between two successive migrations. The migration iteration $t_m[i]$ is controlled by the previous migration $t_m[i - 1]$ and is calculated as follows:

$$t_m[i] = 2t_m[i - 1]. \quad (7)$$

This mechanism supports in the beginning of search process the exploration of search space through the intensive

migration of particles. The increase of $t_m[t]$ supports the exploitive search. The migration of cooperative and of elitistic subswarms is performed in the same generation.

The cooperative subswarms are formed of the three subswarms. Their cooperation is based on migration with migration rate m_r . Each subswarm selects n_m the number of its best particles in generation T_m and replaces the n_m randomly selected particles of swarm. The two cooperative swarms use the RBrr inertia weight update; the third cooperative swarm applies the RBld updating formula. The elitistic swarm uses the RBld inertia weight control.

The selection of subswarm for emigration is controlled randomly. Note that with the probability $p = 1/6$ all three subswarms will substitute their own worst or randomly selected particles with their own best particles, with probability $p = 2/3$ at least one of subswarm interchanges its worst or random particles with its bests, and with probability $p = 1/3$ the subswarm obtains best particles from other cooperative subswarms.

The second group of subswarms is formed from one elitistic swarm. This subswarm searches over the search space and receives the all best particles from cooperative swarms. The best particles substitute the randomly selected particles from elitistic subswarm. The elitistic swarm does not share the knowledge of global best particle with cooperative subswarms.

3. Results

The proposed modifications of inertia weight strategies were applied on 28 CEC2013 benchmark minimization problems [25]. Only 10 dimensional problems were analysed in the presented study. The set of CEC2013 benchmark problem consists of five unimodal functions f1–f5, fifteen multimodal problems f6–f20, and eight composition functions f21–f28.

The search space for all CEC2013 benchmark functions was $\langle -100, 100 \rangle$. Each PSO run was repeated 51 times per one optimisation problem. The maximum number of function evaluations was 100000, as recommended by the CEC2013 benchmark optimisation experiment [25].

The computations were made using the R statistical environment 3.0.2 [26] on 64-bit GNU/Linux operative system, and benchmark functions were used through the implementation of CEC2013 R package v0.1-4 [27]. The R package serves as a wrapper of original C code of 28 benchmark functions [25]. The random number generator was based on the work of Matsumoto and Nishimura [28].

TABLE 2: The minimum values achieved at $10E + 05$ iteration and Min. is the problem solution.

CEC problem	Min.	RBRa	RBRr	AMPSO2	RBlD	sPSO	BrBl
f1	-1400	-1400.00	-1400.00	-1400.00	-1400.00	-1400.00	-1400.00
f2	-1300	31802.55	10719.58	204814.76	20573.61	51187.06	18609.16
f3	-1200	-1199.74	-1199.95	17054.84	-1199.82	-1199.96	-1199.95
f4	-1100	-964.04	-947.26	576.38	-981.92	-913.96	-388.49
f5	-1000	-1000.00	-1000.00	-1000.00	-1000.00	-1000.00	-1000.00
f6	-900	-899.93	-899.99	-899.51	-899.81	-899.76	-900.00
f7	-800	-799.96	-799.32	-798.57	-799.92	-799.95	-798.90
f8	-700	-679.83	-679.80	-679.79	-679.78	-679.84	-679.94
f9	-600	-599.25	-599.23	-599.25	-598.43	-599.43	-599.37
f10	-500	-499.93	-499.95	-499.82	-499.91	-499.91	-499.94
f11	-400	-400.00	-400.00	-397.56	-399.01	-400.00	-400.00
f12	-300	-295.03	-294.03	-284.82	-293.04	-295.03	-295.03
f13	-200	-193.73	-193.73	-182.54	-193.04	-197.16	-192.22
f14	-100	-96.40	-96.40	-40.45	-93.11	-93.05	-93.12
f15	100	243.94	382.90	624.16	343.59	463.56	295.58
f16	200	200.70	200.24	200.38	200.47	200.58	200.10
f17	300	303.29	301.60	324.08	302.45	304.34	301.11
f18	400	414.78	413.57	433.38	414.77	415.52	414.11
f19	500	500.04	500.07	500.89	500.19	500.31	500.14
f20	600	601.96	601.94	602.44	602.03	601.81	602.11
f21	700	800.00	800.00	800.03	800.00	700.00	800.00
f22	800	829.38	823.36	925.66	829.06	830.92	821.09
f23	900	1149.35	1080.55	1613.80	1127.72	1181.06	1338.13
f24	1000	1109.13	1114.66	1125.87	1112.24	1136.31	1109.72
f25	1100	1301.93	1303.37	1302.83	1302.10	1301.53	1213.63
f26	1200	1307.96	1305.97	1321.06	1306.97	1308.95	1304.98
f27	1300	1615.16	1622.81	1658.37	1636.13	1607.02	1603.81
f28	1400	1500.00	1500.00	1500.01	1500.00	1500.00	1500.00

The single PSO parameter settings were based on [3, 24]. The size of population was $n_{\text{pop}} = 40$, the $c_1 = c_2 = 2$, and $v_{\text{max}} = 95$. The populations were randomly initialized within the search space using the uniform distribution and the values of parameters controlling inertia weight were $w_{\text{max}} = 0.9$ and $w_{\text{min}} = 0.4$. The linearly increasing values of shape parameters were $bs_{\text{min}} = 0.1$, $bs_{\text{max}} = 4$, and $t_{\text{max}} = 100000$.

The PSO with proposed inertia weight strategies was compared with standard PSO (sPSO) and AMPSO2. The AMPSO2 uses the Beta distribution on adaptive mutation of the personal best particles and global best particle [29]. The RBRr and RBRa use the shape parameters r_1 and r_2 randomly generated from interval (0, 1).

The parameter settings particle initializations of BrBl subswarms were those used in single PSO. The BrBl migration rate $m_r = 0.2$, the first migration started in the second generation $t_m[1] = 2$, and the total number of realized migrations was 9.

3.1. The Exploration and Exploitation of Proposed PSO Versions. We relate the description of balance between the exploration and exploitation to the evolution of the variances and fitness values of global best particles generated by the all 51 optimisation runs.

The variance of tested PSO versions was described using the standard deviations of differences between fitness values and median, which was obtained from 51 runs in given iteration. The results for the first 4000 iterations on 12 selected benchmark problems are shown in Figures 2, 3, and 4.

On unimodal problems f1–f5 and multimodal problem f17 the AMPSO2 and sPSO show clearly different patterns in the evolution of standard deviations than PSO versions with Beta distribution. The PSO versions with Beta distribution show the decrease of the variance of swarm particles, while the AMPSO2 and sPSO show the stagnation. These similar patterns of decrease and stagnation are apparent on the fitness values of global best particles.

Those patterns are connected to the convergence of tested PSO versions. For example on f1 problem all PSO versions based on Beta distribution found earlier the optimum than AMPSO2 and sPSO (see the results of Table 3).

The BrBl shows the highest variances in the beginning of iteration search. These are connected to the intensive migrations, performed during the early stages of optimisation search. The main benefit is shown in later rapid decrease of fitness value (e.g., see the results in Figure 2). Similar patterns are shown in [17, 21, 30].

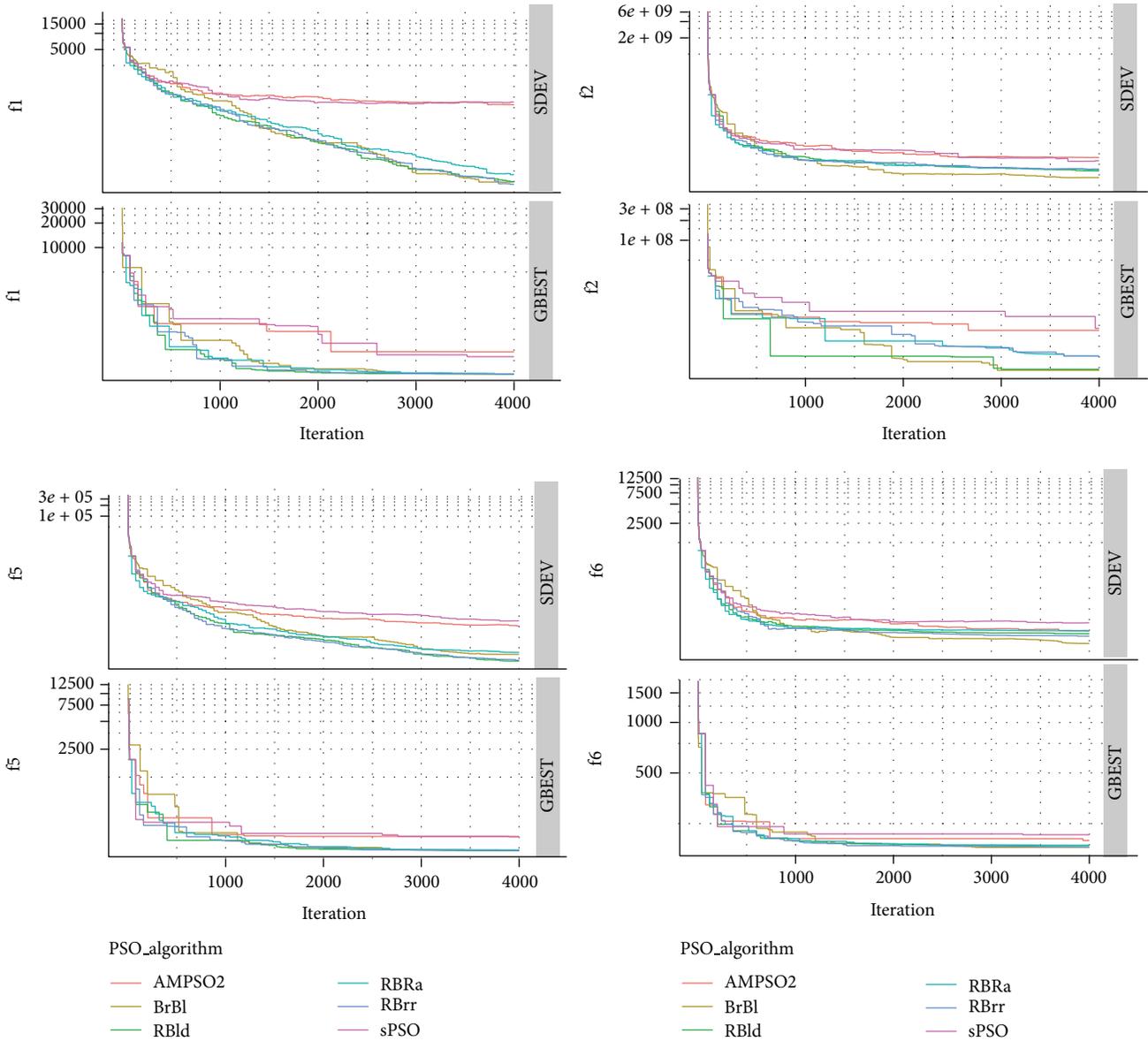


FIGURE 2: The normalized standard deviation and global best model of 51 runs for f1, f2, f5, and f6; SDEV is the standard deviation, GBEST fitness of global best particle. Note: all values are shifted due to the logarithmic transformation of y-axis.

The BrBl version also shows the increase of variance during the search process on f15 and f23. This fact is again connected to the finding of better solutions in terms of values of global best particle (see Table 3). On the other hand the BrBl was on f23 the second worst PSO version (see Table 2).

3.2. The Comparison of PSO Versions. Following the recommendation of CEC2013 the maximum function evaluation (FES) was set as $10E + 05$ [24, 25, 31]. The overall results of fitness values of global best particles are shown Table 3 for $FES = 10E + 04$ and Table 2 for $FES = 10E + 05$.

The PSO versions with Beta distribution components show the best convergence properties on all benchmark problems for $FES 10E + 04$ (see Table 3). The best fitness

values recorded on $FES 10E + 04$ show the RBld for unimodal problems. Three PSO algorithms achieved best fitness values on 15 multimodal problems, f6–f21. They are sPSO on f8, RBld on f13 and f21, and RBrr on f7, f10, f12, and f20. The remaining 8 multimodal problems were described by the BrBl. The RBld and BrBl were superior for composition functions, f21–f28.

The results of $FES 10E + 05$ show that tested versions of PSO solved the following benchmark problems: f1—all PSO versions, f5—all PSO versions, f6—the BrBl version, f11—all PSO versions except the AMPSO2, and RBld, f21—sPSO. These results are comparable with findings of Zambrano-Bigiarini et al. [24] and El-Abd [31]. The BrBl achieved the best values on 13 optimisation problems.

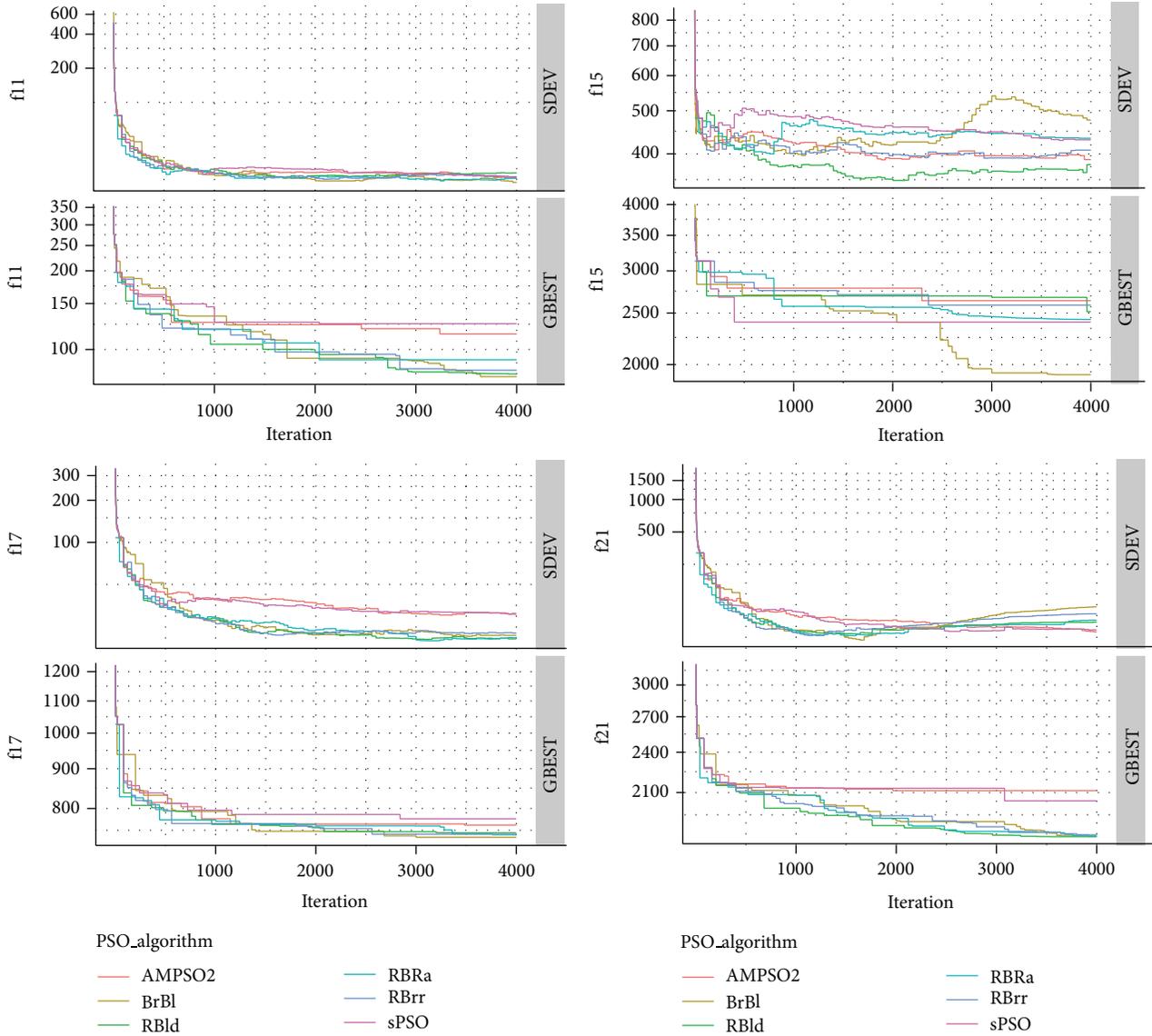


FIGURE 3: The normalized standard deviation and global best model of 51 runs for f11, f15, f17, and f21; SDEV is the standard deviation, GBEST fitness of global best particle. Note: all values are shifted due to the logarithmic transformation of y -axis.

The comparison of mean performance of all 51 runs for FES $10E + 04$ is shown in Table 4. Those results are based on the contrast test of unadjusted median test (for detailed explanation see [32]). The final ranking shows that the BrBl PSO version is superior to the remaining tested versions. Similar results showed the contrast test values obtained for FES $10E + 05$.

These findings were confirmed by the results of paired Wilcoxon test. The P values for Wilcoxon test of BrBl and other PSO versions were statistically significant for FES $10E + 04$ (see Table 5) and also for FES $10E + 05$.

4. Conclusions

The presented analysis evaluates the 6 different versions of PSO algorithm on 28 CEC2013 benchmark functions. The goal was to experimentally compare the different inertia

weight updating strategies related to the random component generated by the Beta distribution.

The computational experiment consists of approximately $8.57E + 08$ function evaluations (28 benchmark functions \times 51 repetitions \times 100000 function evaluations \times 6 versions of particle swarm optimization algorithm). We compared 5 single swarm PSO versions and 1 multiswarm PSO version.

The results of comparison of selected single swarm PSO versions indicate that the Beta distribution applied on inertia weight strategy provides important source of modifications of original PSO. It supports the balanced exploratory and exploitive search. The best single swarm strategies according to the results of contrast test based on unadjusted median are RBld and RBrr.

Our results highlight that the best version from 6 tested PSO modifications is the multiswarm algorithm BrBl. The BrBl combines the swarms with modifications of inertia

TABLE 3: The minimum values achieved at $10E + 04$ iteration and Min. is the problem solution.

CEC problem	Min.	RBRa	RBrr	AMPSO2	RBld	sPSO	BrBl
f1	-1400	-1400.00	-1400.00	-1147.93	-1400.00	-1265.72	-1400.00
f2	-1300	222040.73	322300.27	1498763.45	130380.94	3959345.31	236444.51
f3	-1200	2136685.60	392088.96	1193329345.61	369651.03	331615655.13	6232942.40
f4	-1100	5620.73	5057.65	6101.20	4597.78	7088.07	5451.37
f5	-1000	-999.98	-1000.00	-943.64	-1000.00	-961.61	-999.97
f6	-900	-899.70	-899.84	-878.23	-899.57	-881.82	-899.94
f7	-800	-795.95	-797.12	-752.51	-794.56	-771.83	-788.15
f8	-700	-679.73	-679.75	-679.75	-679.78	-679.84	-679.73
f9	-600	-598.53	-597.77	-593.72	-597.54	-593.42	-599.29
f10	-500	-498.94	-499.44	-464.60	-499.36	-468.55	-498.81
f11	-400	-394.23	-395.61	-368.60	-395.74	-362.84	-395.99
f12	-300	-283.02	-292.15	-252.46	-292.11	-258.93	-286.85
f13	-200	-182.06	-183.29	-147.10	-188.58	-148.35	-179.85
f14	-100	-43.56	116.03	199.00	73.64	1221.41	25.31
f15	100	1366.48	859.64	1416.16	834.71	1331.53	505.67
f16	200	201.00	200.58	200.73	200.87	200.89	200.40
f17	300	327.16	322.55	366.99	323.63	365.88	318.69
f18	400	446.89	438.03	473.89	435.77	482.47	433.20
f19	500	501.16	501.02	508.15	500.75	505.62	500.55
f20	600	602.94	602.78	603.07	602.86	603.40	602.98
f21	700	803.41	800.72	1080.72	800.49	1053.29	801.40
f22	800	1107.86	1009.20	1318.77	870.95	2022.70	1003.78
f23	900	1580.41	1796.86	2286.08	1740.24	2546.45	1514.33
f24	1000	1137.37	1144.17	1175.66	1130.14	1172.10	1135.56
f25	1100	1305.98	1305.48	1315.70	1303.40	1319.22	1257.89
f26	1200	1321.41	1321.35	1347.29	1311.11	1357.60	1308.38
f27	1300	1646.86	1641.18	1799.34	1659.35	1776.92	1637.37
f28	1400	1503.26	1500.31	1692.99	1500.23	1667.17	1500.63

TABLE 4: The contrast test on best values achieved on $10E + 04$ iteration.

	RBRa	RBrr	AMPSO2	RBld	sPSO	BrBl	Ranking
RBRa	—	$2.67E + 00$	$-4.14E + 01$	$2.56E + 00$	$-4.74E + 01$	$7.66E + 00$	4
RBrr	$-2.67E + 00$	—	$-4.41E + 01$	$-1.09E - 01$	$-5.01E + 01$	$4.99E + 00$	2
AMPSO2	$4.14E + 01$	$4.41E + 01$	—	$4.40E + 01$	$-6.00E + 00$	$4.91E + 01$	5
RBld	$-2.56E + 00$	$1.09E - 01$	$-4.40E + 01$	—	$-5.00E + 01$	$5.10E + 00$	3
sPSO	$4.74E + 01$	$5.01E + 01$	$6.00E + 00$	$5.00E + 01$	—	$5.51E + 01$	6
BrBl	$-7.66E + 00$	$-4.99E + 00$	$-4.91E + 01$	$-5.10E + 00$	$-5.51E + 01$	—	1

TABLE 5: The Wilcoxon test on best values achieved on $10E + 04$ iteration.

	RBRa	RBrr	AMPSO2	RBld	sPSO	BrBl	Ranking
RBRa	—	$1.00E + 00$	$1.51E - 110$	$1.00E + 00$	$1.90E - 176$	$1.00E + 00$	4
RBrr	$4.38E - 18$	—	$7.75E - 141$	$4.57E - 01$	$2.22E - 184$	$1.00E + 00$	2-3
AMPSO2	$1.00E + 00$	$1.00E + 00$	—	$1.00E + 00$	$1.38E - 09$	$1.00E + 00$	5
RBld	$3.02E - 18$	$5.43E - 01$	$7.89E - 140$	—	$2.79E - 181$	$1.00E + 00$	2-3
sPSO	$1.00E + 00$	$1.00E + 00$	$1.00E + 00$	$1.00E + 00$	—	$1.00E + 00$	6
BrBl	$5.57E - 48$	$6.00E - 13$	$5.01E - 201$	$4.60E - 18$	$6.51E - 211$	—	1

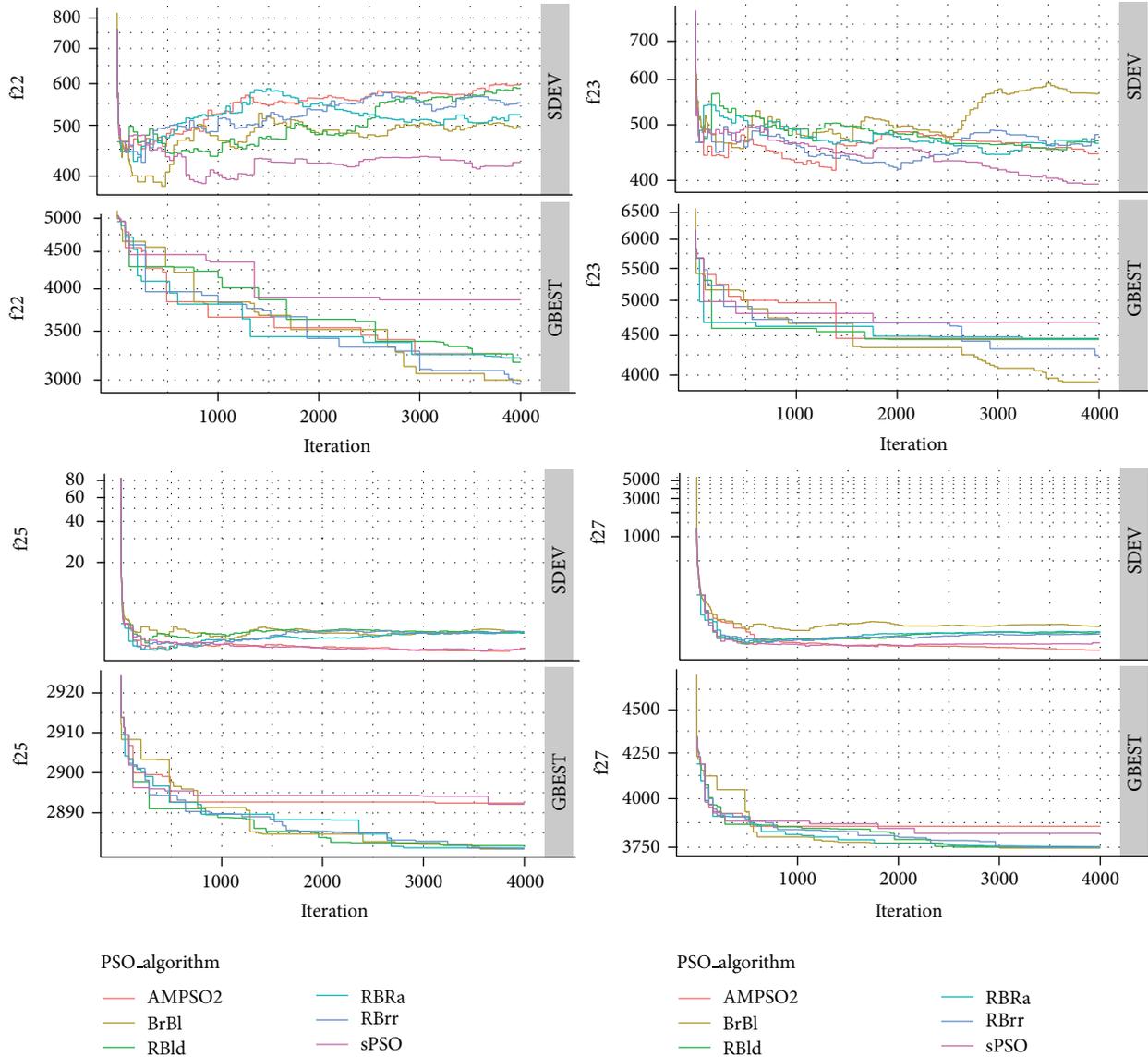


FIGURE 4: The normalized standard deviation and global best model of 51 runs for f_{22} , f_{23} , f_{25} , and f_{27} ; SDEV is the standard deviation, GBEST fitness of global best particle. Note: all values are shifted due to the logarithmic transformation of y -axis.

weight by the random component controlled by the time varied constant shape parameters and randomly varied shape parameters of Beta distributions.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, The University of Western Australia, Perth, Australia, December 1995.
- [2] K. Kameyama, "Particle swarm optimization—a survey," *IEICE Transactions on Information and Systems*, vol. 92, no. 7, pp. 1354–1361, 2009.
- [3] Y. Shi and R. Eberhart, "Modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 69–73, IEEE Computer Society, Washington, DC, USA, May 1998.
- [4] R. Eberhart and Y. Shi, "Tracking and optimizing dynamic systems with particle swarms," in *Proceedings of the Congress on Evolutionary Computation*, vol. 1, pp. 94–100, 2001.
- [5] A. P. Engelbrecht, *Computational Intelligence: An Introduction*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2007.
- [6] J.-B. Park, Y.-W. Jeong, J.-R. Shin, and K. Y. Lee, "An improved particle swarm optimization for nonconvex economic dispatch

- problems," *IEEE Transactions on Power Systems*, vol. 25, no. 1, pp. 156–166, 2010.
- [7] M. Zambrano-Bigiarini and R. Rojas, "A model-independent particle swarm optimisation software for model calibration," *Environmental Modelling & Software*, vol. 43, pp. 5–25, 2013.
- [8] D. Chen and C. Zhao, "Particle swarm optimization with adaptive population size and its application," *Applied Soft Computing Journal*, vol. 9, no. 1, pp. 39–48, 2009.
- [9] G. Xu, "An adaptive parameter tuning of particle swarm optimization algorithm," *Applied Mathematics and Computation*, vol. 219, no. 9, pp. 4560–4569, 2013.
- [10] M. Hu, T. Wu, and J. D. Weir, "An adaptive particle swarm optimization with multiple adaptive methods," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 705–720, 2013.
- [11] B. Liu, L. Wang, Y.-H. Jin, F. Tang, and D.-X. Huang, "Improved particle swarm optimization combined with chaos," *Chaos, Solitons and Fractals*, vol. 25, no. 5, pp. 1261–1271, 2005.
- [12] Y. Feng, G.-F. Teng, A.-X. Wang, and Y.-M. Yao, "Chaotic inertia weight in particle swarm optimization," in *Proceedings of the 2nd International Conference on Innovative Computing, Information and Control (ICICIC '07)*, p. 475, Kumamoto, Japan, September 2007.
- [13] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A novel particle swarm optimization algorithm with adaptive inertia weight," *Applied Soft Computing Journal*, vol. 11, no. 4, pp. 3658–3670, 2011.
- [14] J. C. Bansal, P. K. Singh, M. Saraswat, A. Verma, S. S. Jadon, and A. Abraham, "Inertia weight strategies in particle swarm optimization," in *Proceedings of the 3rd World Congress on Nature and Biologically Inspired Computing (NaBIC '11)*, pp. 633–640, October 2011.
- [15] M. Jakubcová, P. Máca, and P. Pech, "A comparison of selected modifications of the particle swarm optimization algorithm," *Journal of Applied Mathematics*, vol. 2014, Article ID 293087, 10 pages, 2014.
- [16] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, no. 5560, pp. 459–467, 1976.
- [17] B. Niu, Y. Zhu, X. He, and H. Wu, "MCPSCO: a multi-swarm cooperative particle swarm optimizer," *Applied Mathematics and Computation*, vol. 185, no. 2, pp. 1050–1062, 2007.
- [18] M. Gang, Z. Wei, and X. Chang, "A novel particle swarm optimization algorithm based on particle migration," *Applied Mathematics and Computation*, vol. 218, no. 11, pp. 6620–6626, 2012.
- [19] X. Lai and G. Tan, "Studies on migration strategies of multiple population parallel particle swarm optimization," in *Proceedings of the 8th International Conference on Natural Computation (ICNC '12)*, pp. 798–802, IEEE, May 2012.
- [20] T. Blackwell and J. Branke, "Multiswarms, exclusion, and anti-convergence in dynamic environments," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 4, pp. 459–472, 2006.
- [21] J. Zhang and X. Ding, "A multi-swarm self-adaptive and cooperative particle swarm optimization," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 6, pp. 958–967, 2011.
- [22] J.-F. Chang, S.-C. Chu, J. F. Roddick, and J.-S. Pan, "A parallel particle swarm optimization algorithm with communication strategies," *Journal of Information Science and Engineering*, vol. 21, no. 4, pp. 809–818, 2005.
- [23] F. van den Bergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, 2004.
- [24] M. Zambrano-Bigiarini, M. Clerc, and R. Rojas, "Standard particle swarm optimisation 2011 at CEC-2013: a baseline for future pso improvements," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '13)*, pp. 2337–2344, Cancun, Mexico, June 2013.
- [25] J. J. Liang, B. Y. Qu, P. N. Suganthan, and A. G. Hernandez-Diaz, "Problem definitions and evaluation criteria for the CEC 2013 special session and competition on real-parameter optimization," Tech. Rep. 201212, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2013.
- [26] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2013.
- [27] M. Zambrano-Bigiarini and Y. Gonzalez Fernandez, "cec2013: benchmark functions for the special session and competition on real-parameter single objective optimization at CEC-2013," R Package Version 0.1-4, 2013.
- [28] M. Matsumoto and T. Nishimura, "Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Transactions on Modeling and Computer Simulation*, vol. 8, no. 1, pp. 3–30, 1998.
- [29] M. Pant, R. Thangaraj, and A. Abraham, "Particle Swarm Optimization using adaptive mutation," in *Proceedings of the 19th International Conference on Database and Expert Systems Applications (DEXA '08)*, pp. 519–523, IEEE Computer Society, September 2008.
- [30] H. Wang, H. Sun, C. Li, S. Rahnamayan, and J. Pan, "Diversity enhanced particle swarm optimization with neighborhood search," *Information Sciences*, vol. 223, pp. 119–135, 2013.
- [31] M. El-Abd, "Testing a particle swarm optimization and artificial bee colony hybrid algorithm on the CEC13 benchmarks," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '13)*, pp. 2215–2220, IEEE Computational Intelligence Society, Cancun, Mexico, June 2013.
- [32] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

Research Article

Artificial Intelligence Mechanisms on Interactive Modified Simplex Method with Desirability Function for Optimising Surface Lapping Process

Pongchanun Luangpaiboon and Sitthikorn Duangkaew

*Industrial Statistics and Operational Research Unit (ISO-RU), Department of Industrial Engineering,
Faculty of Engineering, Thammasat University, Pathum Thani 12120, Thailand*

Correspondence should be addressed to Pongchanun Luangpaiboon; lpongch@engr.tu.ac.th

Received 6 June 2014; Revised 20 August 2014; Accepted 16 September 2014; Published 2 October 2014

Academic Editor: Rui Mu

Copyright © 2014 P. Luangpaiboon and S. Duangkaew. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A study has been made to optimise the influential parameters of surface lapping process. Lapping time, lapping speed, downward pressure, and charging pressure were chosen from the preliminary studies as parameters to determine process performances in terms of material removal, lap width, and clamp force. The desirability functions of the-nominal-the-best were used to compromise multiple responses into the overall desirability function level or D response. The conventional modified simplex or Nelder-Mead simplex method and the interactive desirability function are performed to optimise online the parameter levels in order to maximise the D response. In order to determine the lapping process parameters effectively, this research then applies two powerful artificial intelligence optimisation mechanisms from harmony search and firefly algorithms. The recommended condition of (lapping time, lapping speed, downward pressure, and charging pressure) at (33, 35, 6.0, and 5.0) has been verified by performing confirmation experiments. It showed that the D response level increased to 0.96. When compared with the current operating condition, there is a decrease of the material removal and lap width with the improved process performance indices of 2.01 and 1.14, respectively. Similarly, there is an increase of the clamp force with the improved process performance index of 1.58.

1. Introduction

The hard disk drive (HDD) precision components mainly consist of magnetic, mechanical, electromechanical, and electronic components. A built-in hard drive of a disk clamp on the spindle motor hub assembly is one of important mechanical components in the HDD. The surface quality of the disk clamp in contact with the disk needs to concentrate via a surface lapping process. It is the precision finishing process with the different mechanical arrangement where a material is precisely removed from a work piece or a specimen. It aims to produce a desired dimensional accuracy and flatness, very fine surface finish, or shape with minimal level of subsurface damage [1]. Lapping is an operation of slow material removal [2]. It aims to decrease the original surface roughness with preset levels of removed material and modified shape. A process of lapping has been applied to

a wide range of materials and applications, ranging between metals, glass, optics, semiconductors, and ceramics [3]. In lapping operations the machining methods can be practically categorised into cylindrical lapping between laps, lapping with bonded abrasives and single and double side flat lapping types.

With the rapid development of the HDD industries there is consequently a need for enhancing all related performance measures of lapping operations. It is more challenging to achieve better surface quality of disk clamps with high manufacturing efficiency. In order to successfully control the process, it is important to analyse all the influential parameters and determine their proper levels in manufacturing processes. Many studies have been performed on various methods of automatic optimisation based on surface lapping process parameters. Response surface methodology (RSM) was widely used to determine and predict the optimal

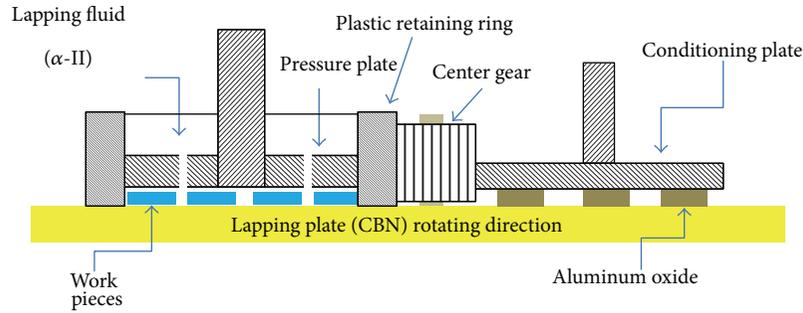


FIGURE 1: Schematic of the section view of surface lapping process.

parameter levels for the maximal improvement of the surface roughness for the cylindrical lapping of fine ceramics [4]. This cylindrical lapping was characterised and optimised because it was time consuming and very complicated. Other powerful tools for designing high quality systems were based on Taguchi designs and analyses [5, 6]. When optimising performance, quality and cost Taguchi's strategy was simple, efficient and systematic. The smaller-the-better quality characteristic was used to determine minimal surface roughness when lapping ceramic blocks. The influential lapping parameter was the abrasive size [7]. The experimental design and analysis including the regression were conducted on a study on surface roughness and material removal rate of D2 steel lapping with a polymer-coated plate. From an investigation the effects and optimal levels of process parameters were lapping time, lapping rotation speed, applied load, and abrasive particle size [8].

2. Surface Lapping Process (SLP)

One of the best known and most widely used machining systems for manufacturing the desired dimension of engineering parts is a single side lapping process. There are some advantages for this type of the lapping process. During machining many work pieces can be simultaneously carried out. The system for holding work is not complicated. It provides consistent cut rates and close accuracies. All machines used in the single side lapping process have a rotating annular-sharped lap plate. Work pieces or disk clamps are positioned on the flat rotating wheel as shown in Figure 1. Lapping mainly includes lapping plate and fluid including the conditioning ring [3]. Operational steps of the lapping process are followed. Firstly, parts or disk clamps are loaded into carriers. The plastic retaining ring and neoprene are then placed to control the parts at the right controlled area. A flat aluminum plate is placed on top of neoprene to protect the work pieces from the pressure plate via a suction cup. Retaining ring slides onto a lapping plate to actuate the pressure plate down. Parts are lapped and coolants are used as lubricants. The process simultaneously applies the downward pressure on a disk, charging pressure of Al_2O_3 , speed up the lapping plate and lapping time. The parts are finally unloaded to clean and dry.

Some important functions and features of SLP components are briefly explained as follows. Lapping plates are made

of cubic boron nitride or CBN with the size of $20\ \mu m$ and hexagonal tiles of 92% minimal coverage. The types of the lapping fluid and the conditioning plate are the cutting fluid Alpha-2 and Al_2O_3 with the grit size number 220, respectively. A human-made synthetic abrasive of CBN is commonly known as borazon TM CBN. It performs well to ferrous metal without carbonisation when interacting with Fe (iron) during a lapping operation of 52100 bearing steel, cast iron, die steel, tool steel, super alloys, and some cases of ceramic materials [3]. Lapping fluid provides an important carrier characteristic in forms of an oil or aqueous medium in various viscosities when abrasive grains are transported to the lapping operation zone. An aim is to achieve a continuous distribution across the lapping plate. The liquid carrier is applied to lubricate two surfaces to achieve the friction reduction between the abrasive and the work piece [3]. It helps to distribute the abrasive product across the lapping plate and finally to remove the abraded debris from the lapping operation zone. The structure of a conditioning plate of a fused crystalline abrasive of Mohs 9 under silicon carbide has very hard crystal to fracture [3]. On lapping operations it is best suited for applying the pressure to break down the crystals.

In this research, the surface lapping process is used to produce dimensionally accurate specimens or the disk clamps to high tolerances. The lapping plate will rotate at the low levels of speed, approximately less than 80 rpm, and the middle range of abrasive particles of $5\text{--}20\ \mu m$. With the preferable levels of reliability and produced work pieces lifetime, the material removal of work pieces including a lap width and a desired clamp force are SLP responses to compromise process and customer specifications within their lower and upper levels. Under the current operating condition, the mean, standard deviation and an overall process performance index (Ppk) of all responses of material removal, surface lap width, and clamp force have unsatisfactory levels when compared to the target (Table 1).

In order to satisfy the customers and get the higher levels of benefit, the investigation and improvement for this process is carried out. The sequential experiments of completely randomised design and the two-level factorial design were applied in a preliminary study to determine the influential parameters via all actual responses. Lapping processes have a large number of parameters that can be varied in order to obtain the desired process responses. The lapping process

TABLE 1: Quality characteristics at the current operating condition.

Response	Current statistics			Specification		
	Mean	Standard deviation	Ppk	Lower	Target	Upper
Material removal	1.383	0.134	0.91	0.25	1	1.75
Lap width	0.767	0.028	0.95	0.45	0.65	0.85
Clamp force	26.54	0.585	0.88	25	28	31

is influenced by lapping pressure, lapping speed, material of the lapping plate, lapping time, grain size of the abrasive, type of lapping fluid, flow rate, and the number of work pieces. It is almost impossible to vary all the parameters that influence the lapping process; one solution is to limit the number of variables. The desirability function approach in the case of the-nominal-the-best was used to compromise the multiple responses of material removal, lap width, and clamp force into single response in forms of the composite desirability function level or D . The modified simplex method was implemented online to drive the process achieving the optimal condition. When there was no improvement from the modified simplex method, an interactive desirability function model was also applied to obtain the satisfactory compromise of the conflict responses instead of the original massive contraction. Artificial intelligence mechanisms of the harmony search and firefly algorithms were generated in each design point of the simplex to search for the better representative. Its aim was to converge to the optimum quickly based on the maximal D levels.

3. Interactive Modified Simplex Method (IMSM) with Desirability Function and Artificial Intelligence Mechanisms

Design and analysis of experiments (DOE) is determined as a sequential, structured, and efficient framework for planning experiments and then describing the relationship between influential parameters affecting a process or product and its output or response(s) in any experimental context. Practically, one or several parameters at two or more levels will be varied by experimenters and the aim is to observe the effect of the changes on one or multiple responses. A frequent phenomenon of this interaction between parameters is taken into account. There is a theoretical argument that the minimal number of trials for finding this relationship is just one more than the number of parameters, that is, a simplex design. On both single and multiple responses, the sequential procedures based on the simplex designs involve changing all parameters from one experiment to the next. The preferable level for one of them can depend on other parameter levels. However, some trials deteriorate the process yield or the composite desirability function level for the multiple response surface optimisation. Various methods have been proposed to solve a large number of optimisation problems and some are considered as NP-complete problems or noisy problems. Conventional optimisation methods are useful tools to obtain the global optimum though some methods are unable to solve within satisfactory execute time. These conditions and

difficulties have forced to develop alternative methods of artificial intelligence for solving such problems via various rules of randomness or natural phenomena such as natural biological evolutionary process or the social behavior of species. Artificial intelligence methods came to the fore and show great efficiency when solving problems. Artificial intelligence methods are commonly categorised into two types of a local and population based methods when searching for the global optimal solution [9, 10]. These mechanisms including an interactive desirability function model have been proposed to prevent the mature convergence of the process parameters.

3.1. Desirability Function Approach (DFA). In robust designed experiments there is a use of measurements on a set of design points with various responses [11]. Instead of optimising each response separately, settings for the influential parameters sought to satisfy all of the responses at once. Multiple response surface optimisation is then used to determine optimal levels of process or product parameters with low variation of responses [12]. There are some transformation scenarios to deal with multiple responses. One among the most frequently used multiple response optimisation strategies in practice is a desirability function approach or DFA which was originally developed by Harrington and modified to be more flexible by Derringer and Suich. It is a scale invariant index or a single compromise function from multiple responses to enable to compare quality characteristics with various units. The DFA basic idea involves transformation of a multiple response problem into a single response problem, by using proper mathematical transformations, to simultaneously optimise those multiple responses. The DFA introduces, for each estimated response determined by the X vector of process parameters, a desirability function of $d_i(\hat{y}_i)$ with feasible levels between zero and one where the desirability level of one represents the closeness of a response to its most desirable level. If the i th response falls within the unacceptable levels the desirability level turns to zero. When a response lies within the tolerance levels but not the ideal levels, the desirability level lies between 0 and 1. There are two cases of one-sided and two-sided desirability transformations when each estimated response of \hat{y}_i is transformed to $d_i(\hat{y}_i)$.

According to the m responses' characteristics there are three forms of the desirability function which consist of the maximisation or the larger-the-best or the minimisation or the smaller-the-best and the optimisation around a target or the-nominal-the-best. For the larger-the-best the response level is expected to the larger the best. When the response

TABLE 2: Transformation parameters of the DFA.

Parameter	Definition
\hat{y}_i	i th estimated response model
Y_i^{MIN}	Minimal acceptable values of the i th response
Y_i^{MAX}	Maximal acceptable values of the i th response
T_i^{MIN}	Lower targets of the i th response
T_i^{MAX}	Upper targets of the i th response

level exceeds a preset criteria level or its requirement the desirability function level is equal to 1. The desirability function of the larger-the-best can be written as follows:

$$d_i(\hat{y}_i) = \begin{cases} 0, & \text{if } \hat{y}_i \leq Y_i^{\text{MIN}}, \\ \left[\frac{\hat{y}_i - Y_i^{\text{MIN}}}{Y_i^{\text{MAX}} - Y_i^{\text{MIN}}} \right]^{\text{PM}_i}, & \text{if } Y_i^{\text{MIN}} \leq \hat{y}_i \leq Y_i^{\text{MAX}}, \\ 1, & \text{if } \hat{y}_i \geq Y_i^{\text{MAX}}. \end{cases} \quad (1)$$

For the smaller-the-best the response level is expected to be the smaller-the-best. When the response level is less than a preset criteria level, the desirability function level is equal to 1. In contrast if the response level exceeds a preset criteria level, the desirability function level equals 0, where the smaller-the-best is equivalent to the maximisation of $-\hat{y}_i$. For the-nominal-the-best the response level is required to achieve a particular target. When the response level equals the target, the desirability function level equals 1. When there is a departure of the response level over a particular target range, the desirability function level equals 0. The desirability function of the-nominal-the-best can be written as follows and the associated effects of the weights are also included in Table 2:

$$d_i(\hat{y}_i) = \begin{cases} 0, & \text{if } \hat{y}_i \leq Y_i^{\text{MIN}} \text{ or } \hat{y}_i > Y_i^{\text{MAX}}, \\ \left[\frac{\hat{y}_i - Y_i^{\text{MIN}}}{T_i^{\text{MIN}} - Y_i^{\text{MIN}}} \right]^{\text{PT1}_i}, & \text{if } Y_i^{\text{MIN}} < \hat{y}_i \leq T_i^{\text{MIN}}, \\ \left[\frac{Y_i^{\text{MAX}} - \hat{y}_i}{Y_i^{\text{MAX}} - T_i^{\text{MAX}}} \right]^{\text{PT2}_i}, & \text{if } T_i^{\text{MAX}} < \hat{y}_i \leq Y_i^{\text{MAX}}, \\ 1, & \text{if } T_i^{\text{MIN}} < \hat{y}_i \leq T_i^{\text{MAX}}. \end{cases} \quad (2)$$

According to the requirement of the user the shape of $d_i(\hat{y}_i)$ is determined via the power coefficients of PM_i , PT1_i , and PT2_i . With the linear shape PT1_i or PT2_i equals one. With a convex PT1_i or PT2_i is larger than 1 and with a concave PT1_i or PT2_i is less than 1. For the-nominal-the-best, if T_i^{MIN} equals T_i^{MAX} on (2), a triangular desirability function will be applied instead of the trapezoidal function [13]. When each of the responses of interest is defined as the desirability function levels, an overall assessment of all the desirability functions

in forms of the geometric mean of the desirability function levels or D is followed:

$$D = \left(\prod_i^m d_i(\hat{y}_i) \right)^{1/m}. \quad (3)$$

D or the composite desirability function level provides a mean level less than or equal to the lowest individual optimisation desirability level. D will increase as the balance of the properties is more favourable regardless of the values taken on by other responses [14]. The objective is to maximise D . When the D level is different from zero it is implied that all responses are simultaneously in a desirable level. Consequently, for the D level close to 1, it is implied that the combination of the different criteria is globally optima or the response levels are simultaneously near the target values.

3.2. Interactive Desirability Function Approach (IDFA). A decision maker (DM) determines this combined dimensionless desirability function for constructing the sequential optimisation procedures. With this potential effectiveness for compromising all responses various works attempt to solve multiple response surface optimisation problems in an interactive manner via a DM's preference parameters such as the shape, bound, and target of a desirability function in both single and integrated frameworks. In the progressive preference or interactive method the DM can progressively articulate the preference information while solving the problem in forms of interactive desirability function approach (IDFA). For sequential procedures of the IDFA it starts from an initialisation phase. The calculation phase is then constituted and followed by the decision-making phase via an optimisation model. During the decision-making phase, the numerical results of the calculation phase are evaluated and articulated the preference information by the DM [13]. The DM can adjust the shape, bound, or target of a desirability function on either tightening or relaxation to repeat the calculation procedure until there is the DM satisfaction via the intersection of desirability functions corresponding to actual responses. The modes of tightening and relaxation are applied when the DM makes the unsatisfactory requirements more stringent and the satisfactory requirements less stringent, respectively. In the IDFA the preference parameters such as shape, bound, and target need to be initialised to construct the desirability function in the first iteration. The initial shape is practically set to be linear whereas the initial bound and target depend on the DM's subject judgments or the feasible ranges of the responses. The iteration starts at zero and the response indices of the tightening or relaxation set is set at null. However, for simplicity it is assumed that merely one response can be tightened or relaxed at a time. With an iteration increase, the desirability functions are then constructed and the optimisation model is solved and subject to process parameters (X) within their lower (LB) and upper (UB) bounds. An objective is to maximise the overall desirability among various aggregation schemes to be employed. For three actual responses, an interactive

desirability function model (IDFM) is then formally defined as

$$\begin{aligned}
 & \text{Maximise } D \\
 & \text{Subject to } d_1(\hat{y}_1) \geq D, \\
 & \quad d_2(\hat{y}_2) \geq D, \\
 & \quad d_3(\hat{y}_3) \geq D, \\
 & \quad D \geq 0 \\
 & \quad \text{LB} < X < \text{UB}.
 \end{aligned} \tag{4}$$

If the current solution is evaluated to reach the optimal compromise solution the IDFA successfully ends. Otherwise, the algorithm moves to adjust the preference parameter in order to improve the unsatisfactory responses. On the DM selection, one of the unsatisfactory responses can be chosen for tightening or one of the satisfactory responses for relaxation. Both tightening and relaxation modes of the selected response can be implemented in the IDFM by adjusting the preference parameters. For the-nominal-the-best the tightening is implemented by making the shape more convex or less concave, by increasing the lower target and/or decreasing the upper target or by increasing the lower bound and/or decreasing the upper bound. The IDFA restarts until the DM is unwilling to tighten or relax any response or no further tradeoff can be made.

3.3. Harmony Search Algorithm (HSA). The HSA, firstly proposed by Lee and Geem, in 2001, is a population based or socially-based inspiration algorithm. It is derived from the behaviour of musicians when they all improvise their various musical instruments to achieve a perfect state of harmony [15]. In the musical improvisation, the aesthetic quality is conceptually determined via the pitch played by each instrument [16]. Moreover, after practice of musicians the harmony quality can be enhanced for all types of musical instruments. Each musician stores all pitches from a good experience of harmony and for the next time it is possible to have a better level. In all the musicians' memories each musician brings any preferable pitch in the possible range to improvise each other to form a new musically harmony vector. If this leads to the better one when compared to existing harmonies in their memories a new harmony is replaced in their memories. With a controlled number of pitches the worst one is also excluded from their memories. There are three possible rules to improvise for each musician [10]. They are rules of playing any pitch from a musician's memory, playing an adjacent pitch of one pitch from a musician's memory, or randomly playing any pitch from the feasible sound range. This process is repeated until there is a fantastic harmony or the termination criterion is met.

Similarly, the improvisation is analogous to the global search scheme in the HSA. For the optimisation the global optimal solution is determined via the levels of a set of parameters. Each musician can be replaced with each parameter and the harmony is analogous to the response of the problem. With an experience to seek for the optimum, any

good solution is stored in each parameter's memory. For the next iteration there is an increase of a possibility to make a good solution. In order to choose one parameter level to improve the D response the HSA follows three rules as above. They consist of rules for choosing any one level from the harmony memory (HM) within the specific harmony memory size (HMS), choosing an adjacent level of any one in the HM which is called the pitch adjustment with an arbitrary distance bandwidth (BW), or randomly choosing any level from the possible range which is defined as randomisation. The optimisation via improvisation is associated with a harmony memory considering rate (P_{HMCR}) and a pitch adjusting rate (P_{PAR}) parameters. The movement of the i th parameter is adjusted to improvise more harmony by the equation of $x_{i+1} = x_i \pm (\text{rand})\text{BW}(\text{ITE})$, where is $\text{BW}(\text{ITE}) = \text{BW}_{\text{MAX}} \text{EXP}(I(\ln(\text{BW}_{\text{MAX}}/\text{BW}_{\text{MIN}})/\text{ITE}))$. In the literature for the HSA parameter levels, the BW_{MAX} , BW_{MIN} , HMS, P_{HMCR} and P_{PAR} are recommended to be within the range of 1, [0.001, 0.1], [20, 50], [0.7, 0.95] and [0.3, 0.7], respectively.

3.4. Firefly Algorithm (FA). Fireflies naturally generate light from a chemical process in light-emitting organs. The light in adult fireflies is evolved to communicate for a sexual selection and to search for potential preys via varied flashing patterns, the flashing rate, and the amount of time of the signal system. At a specific distance from the light source its intensity follows the inverse square law or when there is an increase of the distance the light intensity decreases. Moreover, the light becomes weaker from the air effect when there is a distance increase. These combined factors make most fireflies visible to communicate with each other within a limited distance. Similarly, the flashing light of the successful communication is analogous to the global search scheme in the FA. For the optimisation the global optimal solution is determined via the levels of a set of parameters. The flashing behaviours of fireflies in nature consequently develop the firefly artificial intelligence algorithm (FA) for the global search scheme, firstly developed by Yang [17, 18]. There are some important characteristics in describing the FA as follows. Other fireflies with weaker flashes are attracted by a strong flashing firefly regardless of their sex. That attractiveness is proportional to the brightness and is reversely proportional to the distances. As a result for any two flashing fireflies, the less bright one will move towards the brighter one. If there is no brightness difference, the firefly will move randomly. In the FA, the brightness of a firefly is proportionally determined by the D response.

In the FA there are two important issues which are the formulations of brightness and attractiveness. For simplicity, the brightness of a firefly at a particular location or $I(x)$ is analogous to the response of a solution at a particular parameter level or $f(x)$. The attractiveness or β by the brightness or the higher D response level of a firefly or a specific solution is relative and judged by the other fireflies or other solutions. So the attractiveness function or $\beta(r_{ij})$, whose value decreases with increasing distance between two fireflies, will vary with the distance between i th and j th fireflies (r_{ij}) by any monotonically decreasing functions

such as $\beta(r) = \beta_0 e^{-\gamma r_{ij}^2}$ or $\beta_0/1 + \gamma r_{ij}^2$. The parameters of β_0 and γ are the attractiveness at $r_{ij} = 0$ and a light absorption coefficient or the variation of the attractiveness in the environment, respectively. The γ value is crucially important in determining the convergence speed and the FA behaviour. Over d dimensions, the distance of r_{ij} between i th and j th fireflies at x_i and x_j can be the Cartesian distance function of $r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^d (x_{ik} - x_{jk})^2}$, where x_{ik} is the k th component of the spatial coordinate of x_i for the i th firefly. The movement of the i th firefly which is attracted to another more attractive or the j th firefly is determined by the equation of $x_{i+1} = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha(\text{rand} - 1/2)$, where the second term is from the attraction and the third term is the randomisation of movement with α being the randomisation parameter and rand is a random number with uniform distribution in the $[0, 1]$ interval or a normal distribution $N(0, 1)$. In the d dimensions it is possible to use the randomisation term to modify the parameters with significantly different scales. This term is useful to explore the search space with an increased diversity of the solutions. This movement equation is then a fine balance between local intensive exploitation and global exploration.

3.5. Sequential Procedures of the Proposed IMSM. The sequential optimisation method based on simplex designs firstly introduced by Spendley et al. expresses contrary to fundamental factorial designs in some characteristics [19]. This rigid simplex method (RS) is described as an evolutionary method for multiple parameter and interactive optimisation systems. It can find the global optimum with fewer trials via empirical, self-improving, and efficient strategies. The RS is based on geometry of the convex hull with an initial $k + 1$ design points, where k is the number of parameters that correspond with dimensions of searching space. Thus, in case of two-parameter optimisation, the simplex design takes the form of a triangle and in case of three parameters, it will be a tetrahedron. After evaluating and ranking the responses from all initial design points, the RS procedure is sequential with a removal of the least favourable response design point from an initial simplex. Continually forming a new simplex, one opposite new design point is determined in the hyperplane of the remaining design points. The RS evolves toward the optimum by simultaneously improving all parameters in every vertex. In order to improve the performance of the RS, many modifications have been proposed. One among them, the Nelder-Mead or modified simplex method (MSM), offers the effectiveness of the rapid convergence by employing an expansion or a contraction operator of the reflection according to quality of the responses including the massive contraction toward the best vertex. For possible stopping criterions Nelder and Mead included the standard deviation of the estimated responses at all the vertices of the simplex and the parameter levels and compared with their specified tolerances [20]. However, in this research there are some modifications applied to the original MSM.

In the IMSM, when focusing on the two-parameter maximisation of the composite desirability function level (D)

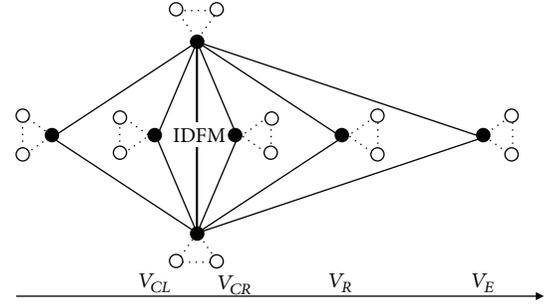


FIGURE 2: Different simplex moves of V_R , V_E , V_{CR} , V_{CL} , and the IDFM from the rejected trial vertex (V_L).

or D response with a triangle simplex, sequential procedures are given as follows. The simplex design is first applied at either arbitrary design points or previous operating design points within the safe region of operation including the neighbourhood design points from artificial intelligence mechanisms from the HSA and FA (Figure 2). All three design points evaluated their own D responses and the best will be selected to form a simplex. All selected design points evaluated their composite desirability function levels and are categorised as the vertices corresponding to the highest (V_H), second lowest (V_{SL}), and lowest (V_L) of D responses. The lowest D response vertex is identified and reflected in the opposite hyperface to obtain the reflected vertex (V_R) via the centroid (\bar{P}) obtained by all remaining vertices in the simplex. Hence the reflected vertex (V_R) may be given by

$$V_R = \bar{P} + \alpha(\bar{P} - V_L), \quad (5)$$

where \bar{P} is the centroid of the design points on the reflecting line. In general, the reflection coefficient parameter of α is greater than 0 and it is the ratio of the distance between $\bar{P} - V_L$ and $\bar{P} - V_R$ and is normally defined to be 1. If the D response of V_R lies between the lowest and highest levels, a new simplex is restarted by replacing the vertex V_L with V_R . If the response of V_R produces the highest level from the new simplex, an expansion operation is applied so as to further enhance the D response level by expanding the reflected vertex V_R along the direction of $\bar{P} - V_R$ to a new expansion vertex of V_E via the following function:

$$V_E = \bar{P} + \gamma(\bar{P} - V_L), \quad (6)$$

where the expansion coefficient parameter of γ is the ratio of the distance between $\bar{P} - V_E$ and $\bar{P} - V_R$ and is normally defined to be larger than 1. If the expansion vertex of V_E produces the highest level of D response when compared to D responses from others, the new simplex is restarted by replacing V_H with V_E . Otherwise, the simplex is restarted by replacing V_H with V_R . In some situation of the IMSM there is a use of two contraction processes. Firstly, if the reflected vertex of V_R produces the second highest level of the D response only with the exception of V_L , a positive contraction process may

Procedure of IMSM ()**While** (termination criterion not satisfied)—(line 1)**Schedule activities**

Generate and determine D responses on the simplex design vertices and their additional design points from artificial intelligence mechanisms

Select the best in each vertex to form the simplex

Reflection of the least D response is processed

Compute and determine D responses on V_R or $f(V_R)$ and its additional design points from artificial intelligence mechanisms

Compare D responses for the new simplex

If $f(V_R)$ is the highest **then**

Extension of V_E and determination of D responses on V_E or $f(V_E)$ and its additional design points from artificial intelligence mechanisms will be processed

Else

If V_R and $f(V_R)$ continue to be the least **then**

Contraction of V_{CL} and determination of D responses on V_{CL} or $f(V_{CL})$ and its additional design points from artificial intelligence mechanisms will be processed

or

Contraction of V_{CR} and determination of D responses on V_{CR} or $f(V_{CR})$ and its additional design points from artificial intelligence mechanisms will be processed

or

Formulation of the IDFM will be processed

Else go to line 3.

End if

End if

End schedule activities

End while

End procedure

PSEUDOCODE 1: Pseudocode of IMSM.

be made to generate the contracted vertex of V_{CR} using the following equation:

$$V_{CR} = \bar{P} + \beta^+ (\bar{P} - V_L), \quad (7)$$

where β^+ is the positive contraction coefficient parameter with the feasible range of $[0, 1)$. If the reflected vertex of V_R produces the lowest level of the D response of all design points, the negative contraction process may be made to generate the contracted vertex of V_{CL} using the following equation:

$$V_{CL} = \bar{P} - \beta^- (\bar{P} - V_L), \quad (8)$$

where β^- is the negative contraction coefficient parameter with the feasible range of $[0, 1)$. It is noticed that these two contraction processes may be achieved within the unfavourable D responses for a decision maker. However, in this research there is a use of the IDFA via the IDFM based on all $2(k+3)$ design points instead of the massive contraction or a shrink process to give an additional flexibility. The next run is carried out with parameters set at values corresponding to this new design point. An idea of IMSM's logical decision and its flowchart are shown in Pseudocode 1 and Figure 3.

4. Numerical Results and Discussions

The objective of a surface lapping process (SLP) is to remove material with a smooth and flat surface with dimensionally accurate specimens to high tolerance. Various parameters from the SLP can vary from application such as a lapping machine with different levels of speed and lapping plates, workstations for controlling lapping fixtures and the equipment to control the plate flatness. During the preliminary process parameter analyses, the brainstorming activities among all engineers from departments of process, advanced process development, product, tooling, and quality assurance announced eight possible parameters affecting the process responses of interest. They consist of lapping time, lapping speed, lapping pressure, Al_2O_3 charging pressure, type of lapping plate, aluminum oxide stone grit size, cutting fluid, and the number of experimental specimens per fixture. However, in this research the type of lapping plate, the cutting fluid, the aluminum oxide stone grit size, and the specimens per fixture are controlled with the $20\ \mu m$ lapping plate of cubic boron nitride (CBN) with the hexagonal tiles, the alpha-2 fluid type, the 220 grit size of aluminum oxide stone, and 22 pieces, respectively. The feasible levels, the current operating condition, their possible differences of operation (Δ), and types of all influential parameters of the lapping time (x_1),

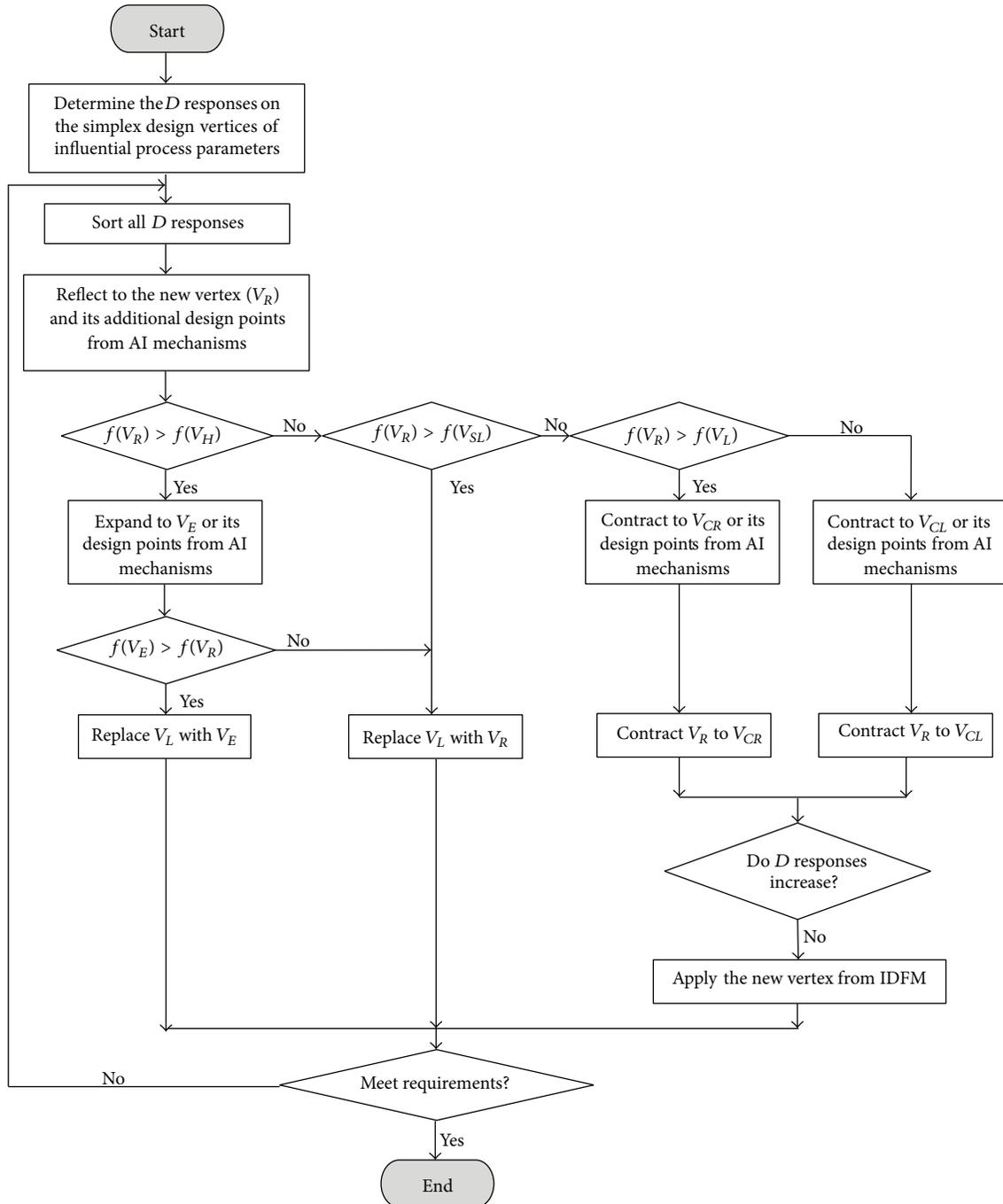


FIGURE 3: Flowchart of IMSM.

lapping speed (x_2), downward pressure (x_3), and charging pressure (x_4) are shown in Table 3.

The purpose of the problem is to improve the SLP of the disk clamps on three different responses by controlling four parameters. Three major SLP's quality performance measures consist of the material removal (y_1), lap width (y_2), and clamp force (y_3) as shown in Figure 4. In addition, the material removal is based on the process requirement specification whereas the remaining responses of lap width and clamp force are measured and depend on customer requirement

specifications. The process and customer specifications are required to reach the nominal value. Currently, it is found that the process quality characteristics are still quite low at 0.4554 of the composite desirability function level or D response. A measure of the sample mean and standard deviation of the initial and finished height is done before and after the lapping process via a free-state height gage from Mitutoyo. The volume removed of materials is then calculated via the multiplication of the sectional area and the mean diameter of specimens. The sample mean and standard

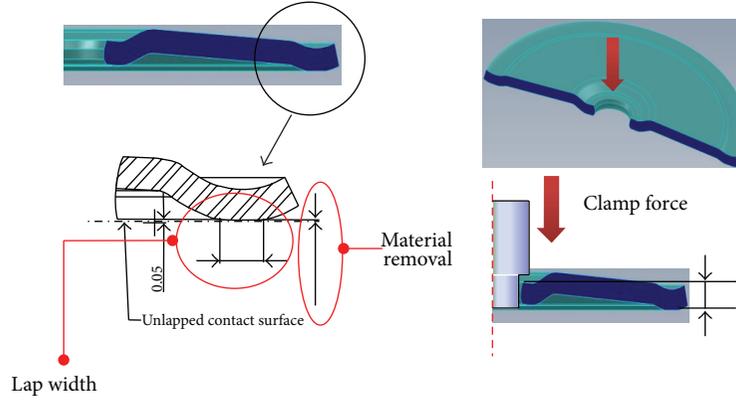

 FIGURE 4: Measurements of y_1 , y_2 , and y_3 for the SLP.

TABLE 3: Process parameters and their feasible and current levels.

Process parameter	Feasible levels			Δ	Unit
	Lower	Upper	Current		
x_1	30	999	60	1.0	sec.
x_2	20	80	30	1.0	rpm
x_3	4	15	8	0.25	psi
x_4	3	15	8	0.25	psi

deviation of the lap width and clamp force are measured via an OGP smartscope CNC vision model: ZIP 250 with DRS 500 laser and Instron 3344 force tester, respectively. Analyses of the test methods and measuring instruments including the entire processes to obtain measurements are considered to determine the uncertainty of our measurement systems. In this research, they were done before performing an improvement of the manufacturing process via % GR&R and P/T ratio to address the precision of a measurement system and the ratio of the measurement system precision to the total tolerance of the manufacturing process, respectively. From additional tests, the levels of P/T ratio and % GR&R are within the acceptable ranges as shown in Table 4.

In screening experiments a completely randomised design or one-way analysis of variance (ANOVA) is used to analyse all process parameters. The past experimental data can summarised the significant parameters categorised by the natural and D responses. If P value exceeds the 5% preset value of significance level (α), there is no effect of parameters. In accordance with the experimenter experience, all SLP parameters except x_4 are being confirmed for their statistically significant effects on all actual and desirability function responses as appeared in Table 5. However, in this investigation, the charging pressure affects the lap width and its desirability function as well.

On the IMSM, four SLP parameters have significant effects on all the responses from the preliminary experiments carried out using the available operating conditions of the company. Though the mean and standard deviation of actual responses of material removal, lap width and clamp force including its Ppk indicate a good reproducibility of the current condition of (x_1 , x_2 , x_3 , and x_4) at (60, 30,

8.0 and 8.0) The design points of the initial simplex consist of five vertices. In each vertex the additional two design points are also established via metaheuristic mechanisms of the HSA and FA. The corresponding D responses in the i th vertex and its metaheuristic design points (V_{iHSA} and V_{iFA}) are determined and the best one will be included in the simplex for the next sequence of the IMSM. The assigned values of the HSA and FA parameters are suggested by the experts of the company and decided by the recommended levels from the literatures. All design points that are carried out according to their corresponding responses are measured in forms of the D response. They are categorised in descending orders of V_H , V_{S3} , V_{S2} , V_{S1} , and V_L . The simplex is then moved in the direction given by the rules of the IMSM including reflection, expansion, and contraction vertices including the interactive desirability function model (IDFM). The experimental conditions from the proposed IMSM based on the flowchart for all the vertices and variations of the parameter levels throughout the multiple response surface optimisation are given in Table 6.

The SLP evolution with four cycles via the IMSM can be described as follows. After ranking all vertices in the initial simplex from the worst vertex with the D response of 0.5829 to the best vertex with the D response of 0.8729, the worst vertex was reflected to the opposite face. The new vertex (V_6) including its neighbourhood design points from the HSA (V_{6HSA}) and the FA (V_{6FA}) were generated and determine their D responses in order to establish the new simplex. During the reflection procedure the experimental domains for all parameters were justified within possible operating levels throughout. When the best reflected vertex was carried out, the D response was worse than the remaining vertices. This result indicated that the V_{CL} including its neighbourhood design points from the HSA (V_{7HSA}) and the FA (V_{7FA}) could be generated to form the new simplex. After determining the best of D responses this led to the worst, the first interactive desirability function model (IDFM¹) was formulated via all previous design points.

In analysing the experimental data for the IDFM¹, the goodness-of-fit checking for the regression equation or the model is necessarily required. These model adequacy checking tests categorised by the actual response includes test

TABLE 4: Measurement system analysis.

Response	Device	P/T ratio	% GR&R
Material removal	Free state height gage	8.0%	5.33%
Lap width	OGP smartscope CNC vision model: ZIP 250 with DRS 500 laser	1.95%	1.4%
Clamp force	Instron 3344 force tester	15.1%	2.28%

TABLE 5: One-way ANOVA: process parameters versus actual and desirability function responses.

Parameter	Actual response			P value			
	y_1	y_2	y_3	$d_1(\hat{y}_1)$	$d_2(\hat{y}_2)$	$d_3(\hat{y}_3)$	D
x_1	0.000	0.000	0.000	0.000	0.000	0.000	0.000
x_2	0.000	0.000	0.000	0.000	0.000	0.000	0.000
x_3	0.000	0.000	0.000	0.000	0.000	0.000	0.000
x_4	0.096	0.018	0.429	0.100	0.020	0.429	0.077

for significance of the regression model, test for significance on the IDFM¹ coefficients, and test for lack of fit via an analysis of variance or ANOVA (Table 7). On analyses of the estimated responses of the material removal (\hat{y}_1), lap width (\hat{y}_2), and clamp force (\hat{y}_3) it is recommended that the linear regression models are statistically significant for all responses. The associated P values for the models of \hat{y}_1 , \hat{y}_2 , and \hat{y}_3 in the form of ANOVA are less than 0.05 with the 95% confidence except for x_3 for \hat{y}_3 . These express that the linear regression models provide good explanations of the relationship between the process parameters and all types of responses. The response equations are given as follows:

$$\begin{aligned}
\hat{y}_1 &= 1.27057 + 0.18951x_1 + 0.12805x_2 + 0.07607x_3 \\
&\quad + 0.05757x_4, \\
\hat{y}_2 &= 0.75331 + 0.038274x_1 + 0.027153x_2 + 0.016471x_3 \\
&\quad + 0.012711x_4, \\
\hat{y}_3 &= 26.8428 - 0.14695x_1 - 0.0769x_2 - 0.05294x_3 \\
&\quad - 0.0403904x_4.
\end{aligned} \tag{9}$$

The actual responses are then transformed into the scale free or desirability function response of the-nominal-the-best for the material removal, lap width, and clamp force denoted as $d_1(\hat{y}_1)$, $d_2(\hat{y}_2)$, and $d_3(\hat{y}_3)$, respectively. It is the level between 0 and 1 and there is an increase when the corresponding response level becomes more desirable. The desirability functions based upon the initial shape, bound, and target are generated for all responses (Figure 5). The associated levels for all DFA transformation and IDFM parameters are given in Tables 8 and 9, respectively. The optimal solution, the IDFM¹ of (x_1 , x_2 , x_3 , and x_4) at (40, 33, 8.0, and 8.0) or V_8 , was solved via the optimisation model. From V_8 and its neighbourhood design points, the corresponding desirability function levels of $d_1(\hat{y}_1)$, $d_2(\hat{y}_2)$, $d_3(\hat{y}_3)$, and D were 1.0000, 0.8842, 0.7615, and 0.8762, respectively. The associated result of \hat{y}_1 was considered satisfactory, but

the estimated actual responses of \hat{y}_2 and \hat{y}_3 were not. Thus, the sequential procedures of the IMSM were continued by forming the new simplex.

With the basic simplex rules each vertex with the best D response from all three possible design points was accumulated to form the new simplex. The reflected rule was then applied to calculate the new vertex (V_9) including its neighbourhood design points from the HSA ($V_{9\text{HSA}}$) and the FA ($V_{9\text{FA}}$) design points. The resulting D response was 0.7277. This result indicated that the V_{CL} including its neighbourhood design points from the HSA ($V_{10\text{HSA}}$) and the FA ($V_{10\text{FA}}$) formed the new simplex. However, the D response deteriorated and there was a use of the second interactive desirability function model (IDFM²). After team brainstorming the decision maker decided to adjust the parameter bounds as appeared in Table 9. The optimal solution, the IDFM² of (x_1 , x_2 , x_3 , and x_4) at (37, 40, 7.0, and 5.0) or V_{11} , was solved via the optimisation model. From V_{11} and its neighbourhood design points, the corresponding desirability function levels of $d_1(\hat{y}_1)$, $d_2(\hat{y}_2)$, $d_3(\hat{y}_3)$, and D were 1.0000, 1.0000, 0.7652, and 0.9146, respectively.

The procedures were repeated to achieve the new reflected vertex of V_{12} , but the outcome was not preferable. The positive contraction process to the reflected vertex or V_{CR} (V_{13}) was then carried out. However, the D response was not satisfactory and there was a use of the third interactive desirability function model (IDFM³) with the new levels of power coefficients as shown in Table 9 and Figure 6. The optimal solution, the IDFM³ of (x_1 , x_2 , x_3 , and x_4) at (30, 40, 8.0, and 5.0) or V_{14} , was solved via the optimisation model. From V_{11} and its neighbourhood design points, the corresponding desirability function levels of $d_1(\hat{y}_1)$, $d_2(\hat{y}_2)$, $d_3(\hat{y}_3)$, and D were 0.9958, 1.0000, 0.8924, and 0.9572, respectively. From Table 6, it was shown that the numerical results on the 4th cycle repeated at the negative contraction process to the worst vertex or V_{CL} (V_{16}) after realizing that the V_R (V_{15}) led to the worse D response when compared. There was then a use of the fourth interactive desirability function model (IDFM⁴) with the shape parameters of (PT₁₁, PT₂₁, PT₁₂, PT₂₂, PT₁₃, and PT₂₃) at (0.1, 0.1, 1.0, 1.0, 0.3, and 0.3) as shown in

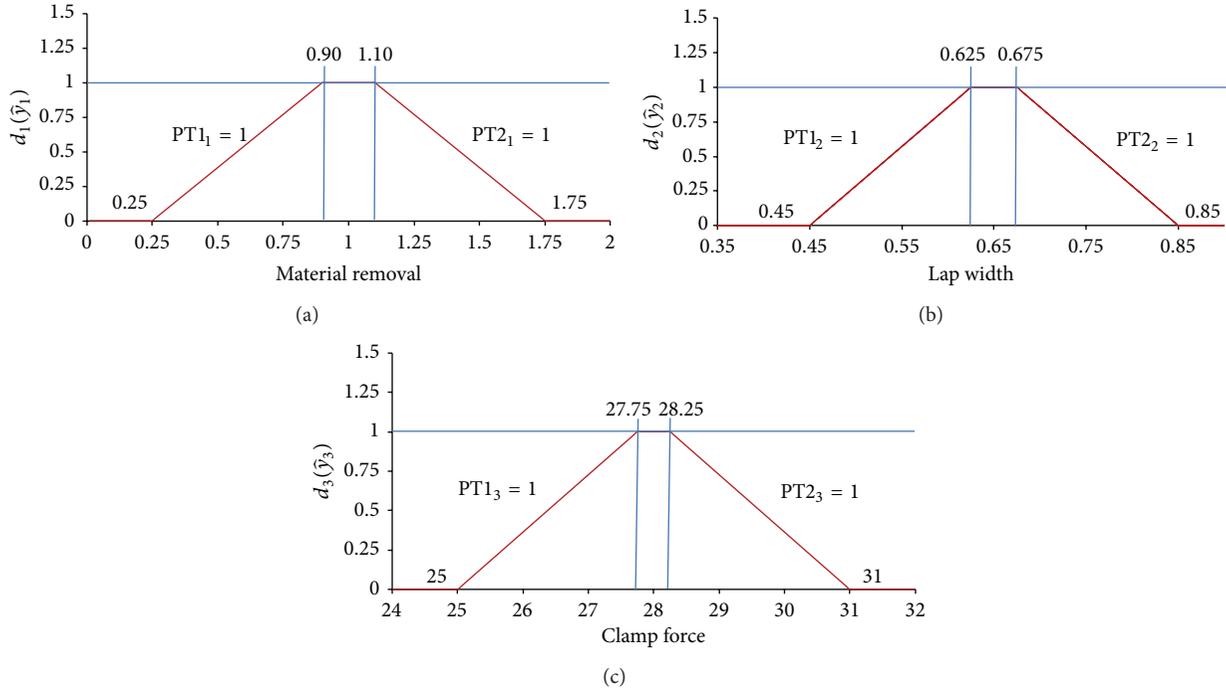


FIGURE 5: Power coefficients of the IDFM¹ and IDFM².

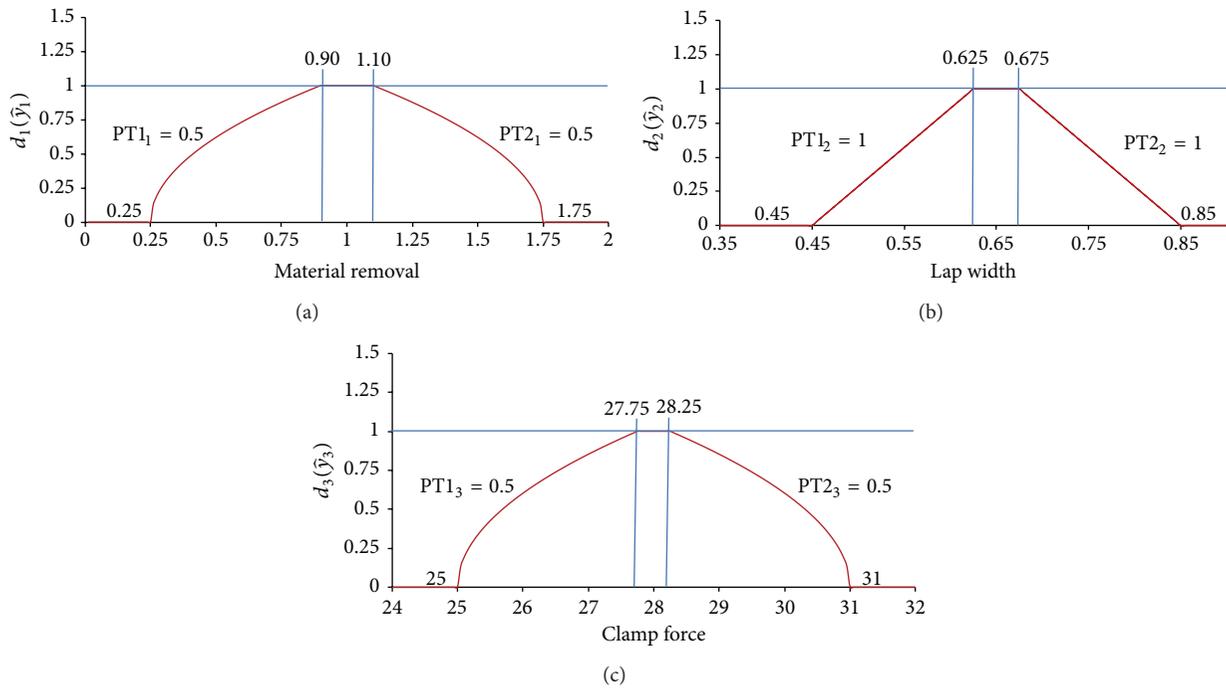


FIGURE 6: Power coefficients of the IDFM³.

Figure 7. The optimal solution, the IDFM⁴ of $(x_1, x_2, x_3,$ and $x_4)$ was at (33, 35, 9.0, and 5.0) and the corresponding desirability function levels of $d_1(\hat{y}_1)$, $d_2(\hat{y}_2)$, $d_3(\hat{y}_3)$, and D were 0.9853, 1.0000, 0.9435, and 0.9734, respectively. Within smaller number of cycles of the numerical results on the SLP

the IMSM successfully found a satisfactory outcome of the D response by adjusting shape parameters.

A confirmation technique via ANOVA is carried out to analyse experimental data in which the actual responses are evaluated under the current operating condition and the new

TABLE 6: Experimental design points and their responses from the IMSM.

Design point	x_1	x_2	x_3	x_4	$d_1(\hat{y}_1)$	$d_2(\hat{y}_2)$	$d_3(\hat{y}_3)$	D	Rank
V_1									
V_{1HSA}	48	26	7.00	7.50	0.9380	0.9444	0.7503	0.8729	V_H
V_{1FA}									
V_2									
V_{2HSA}	45	25	6.50	7.50	0.8093	0.9600	0.7464	0.8339	V_{S3}
V_{2FA}									
V_3									
V_{3HSA}	52	27	7.50	7.50	0.9231	0.7250	0.7310	0.7880	V_{S2}
V_{3FA}									
V_4									
V_{4HSA}	41	24	6.50	7.00	0.7043	0.7120	0.8254	0.7453	V_{S1}
V_{4FA}									
V_5									
V_{5HSA}	56	29	7.50	8.00	0.6297	0.4833	0.6508	0.5829	V_L
V_{5FA}									
V_6									
V_{6HSA}	60	30	12.00	8.00	0.4969	0.4979	0.6202	0.5354	V_R
V_{6FA}									
V_7									
V_{7HSA}	60	40	8.00	8.00	0.6214	0.3212	0.7012	0.5192	V_{CL}
V_{7FA}									
V_{8IDFM}	40	33	8.00	8.00	1.0000	0.8842	0.7615	0.8762	IDFM ¹
V_9									
V_{9HSA}	40	40	8.00	12.00	0.8711	0.6381	0.6933	0.7277	V_R
V_{9FA}									
V_{10}									
V_{10HSA}	40	40	12.00	8.00	0.8395	0.6602	0.7185	0.7357	V_{CL}
V_{10FA}									
V_{11IDFM}	37	40	7.00	5.00	1.0000	1.0000	0.7652	0.9146	IDFM ²
V_{12}									
V_{12HSA}	42	24	6.6	7.1	0.7580	0.8800	0.7889	0.8073	V_R
V_{12FA}									
V_{13}									
V_{13HSA}	60	30	8	8	0.8395	0.7714	0.7105	0.7720	V_{CR}
V_{13FA}									
V_{14IDFM}	30	40	8.00	5.00	0.9958	1.0000	0.8924	0.9572	IDFM ³
V_{15}									
V_{15HSA}	52	28	7.50	7.50	1.0000	0.7250	0.7310	0.8093	V_R
V_{15FA}									
V_{16}									
V_{16HSA}	41	24	6.50	7.00	0.7043	0.9070	0.8254	0.8079	V_{CL}
V_{16FA}									
V_{17IDFM}	33	35	9.00	5.00	0.9853	1.0000	0.9435	0.9734	IDFM ⁴

design point from the IMSM. It can also be seen that these experimental results on all scenario were statistically significant with 95% confidence interval with P value of 0.000. The numerical results suggested that the new one provided the better performance in terms of the statistical significance

of P value for all responses throughout. The desired or major quality characteristics on both process and customer specifications are improved and expressed in terms of sample mean, sample standard deviation, and process performance index as shown in Table 10. With this combination of process

TABLE 7: Estimated effects and coefficients for the actual responses and their ANOVA for regression equation tables on the IDFM¹.

Response		Constant	x_1	x_2	x_3	x_4	Regression ANOVA
\hat{y}_1	Coef.	1.27057	0.18951	0.12805	0.07607	0.05757	
	T	55.76	12.79	8.64	5.13	3.89	0.000
	P value	0.000	0.000	0.000	0.001	0.001	
\hat{y}_2	Coef.	0.75331	0.038274	0.027153	0.016471	0.012711	
	T	168.39	8.56	6.07	3.68	2.84	0.000
	P value	0.000	0.000	0.000	0.001	0.008	
\hat{y}_3	Coef.	26.8428	-0.14695	-0.0769	-0.05294	-0.04039	
	T	3.46	-7.19	-3.76	-2.59	-1.98	0.000
	P value	0.000	0.000	0.001	0.015	0.058	

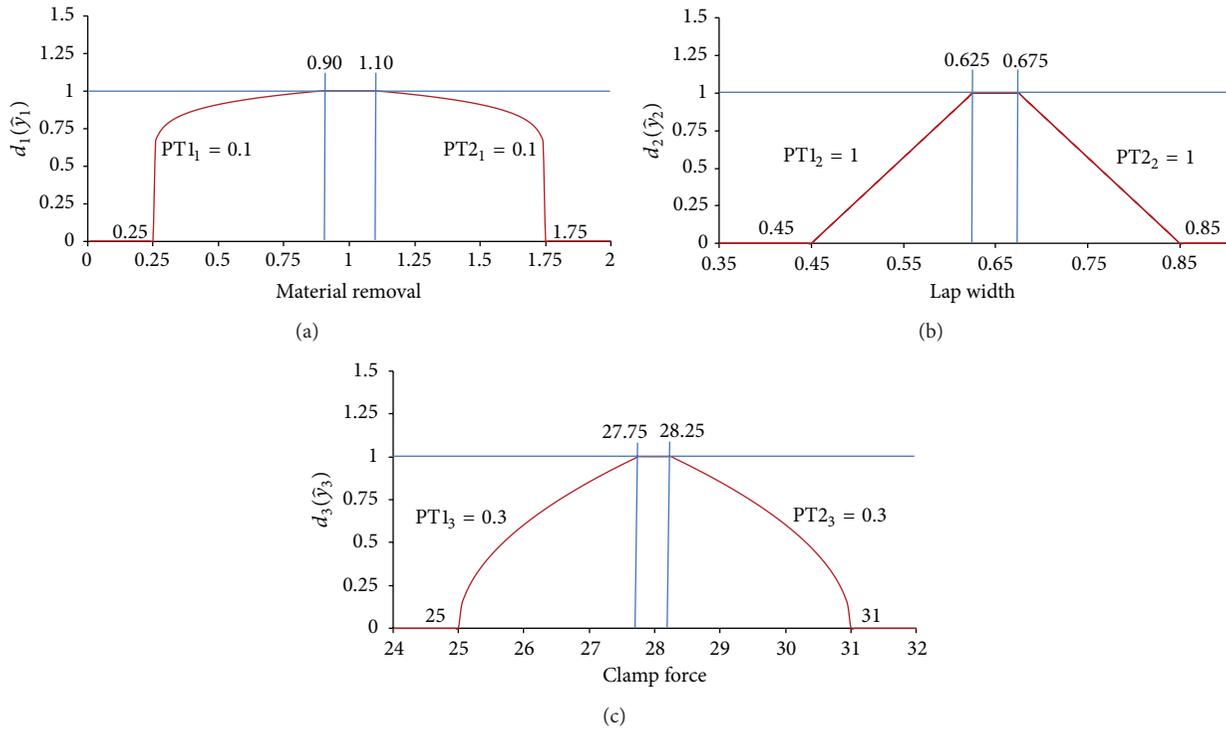


FIGURE 7: Power coefficients of the IDFM¹.

parameters it leads to the successfully controlled surface quality characteristics of the work piece surface in contact with the disk with high manufacturing efficiency. The scenario of the higher levels of the process parameters of lapping speed (x_2) and downward pressure (x_3) indicates that there is constant and stable applied force on the specimen. Consequently, the minor surface quality measurement based on the flatness and the roughness is also improved as shown in Figures 8 and 9, respectively.

5. Summary and Conclusion

With the higher levels of demand of disk clamps in ultraprecision mechanical components, the surface lapping process needs to focus its major performances to meet process and customer satisfactions. This process intends to introduce a finely ground flat on disk clamps and removes material from

TABLE 8: Transformation DFA parameters and their levels.

Level for the i th response	DFA Parameter			
	Y_i^{MIN}	Y_i^{MAX}	T_i^{MIN}	T_i^{MAX}
1	0.25	1.75	0.90	1.10
2	0.50	0.8	0.625	0.675
3	25.0	28.0	27.75	28.25

the disk contact radius to provide the desired dimension of the surface. The parameters of the lapping time, lapping speed, downward pressure, and charging pressure have an important role from the process knowledge when compared to other parameters of the type of lapping plate, aluminum oxide stone grit size, cutting fluid, and the number of experimental specimens per fixture. Consequently, they are controlled during the optimisation phase. However, it is

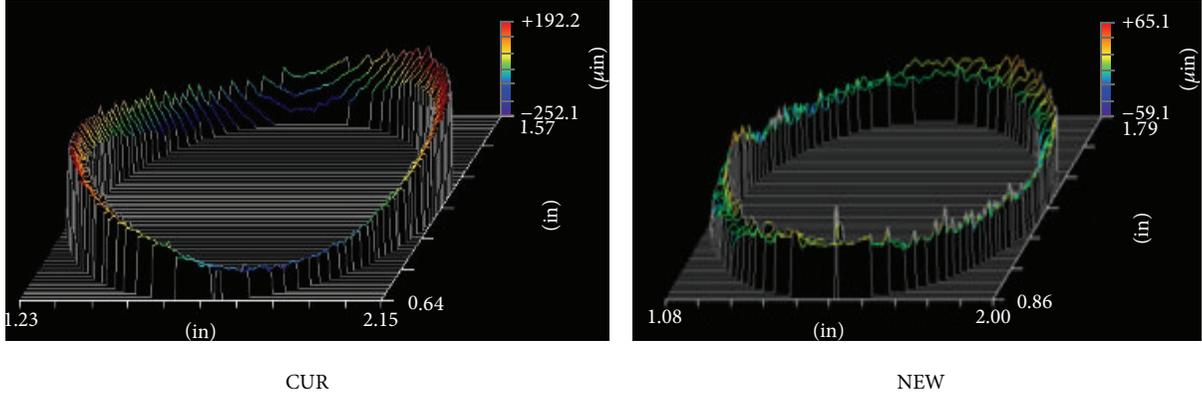


FIGURE 8: Comparative results of the flatness based on the current (CUR) and the new (NEW) operating conditions.

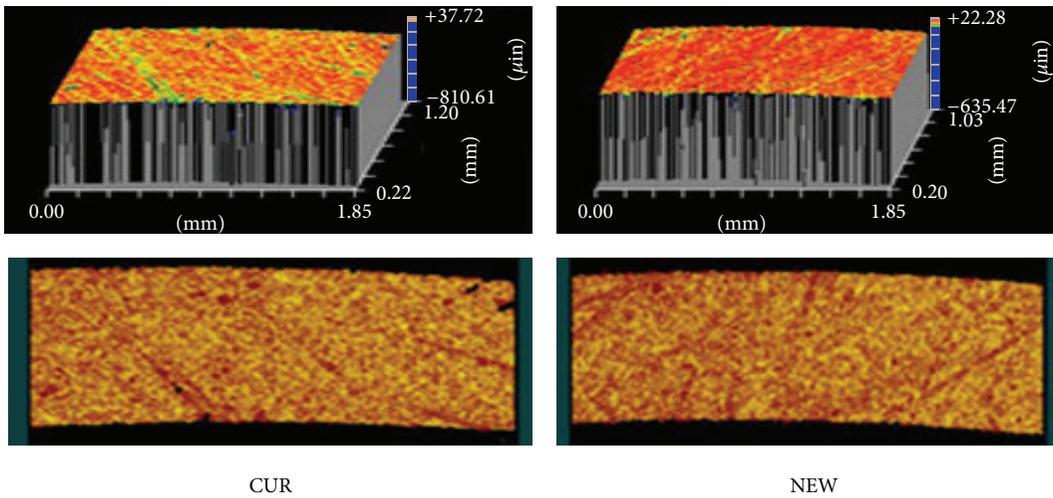


FIGURE 9: Comparative results of the roughness based on the current (CUR) and the new (NEW) operating conditions.

TABLE 9: Lower and upper bounds of process parameters and power coefficients of the IDFM for all iterations.

Parameter		Cycle (n)				
		1	2	3	4	
Process parameter (x_j)	x_1	LB $_j$	40	30	30	30
		UB $_j$	60	60	60	60
	x_2	LB $_j$	30	30	30	30
		UB $_j$	40	40	40	40
	x_3	LB $_j$	8.0	5.0	5.0	5.0
		UB $_j$	12.0	15.0	15.0	15.0
	x_4	LB $_j$	8.0	5.0	5.0	5.0
		UB $_j$	12.0	15.0	15.0	15.0
IDFM parameter for y_i	y_1	PT1 $_i$	1.0	1.0	0.5	0.1
		PT2 $_i$	1.0	1.0	0.5	0.1
	y_2	PT1 $_i$	1.0	1.0	1.0	1.0
		PT2 $_i$	1.0	1.0	1.0	1.0
	y_3	PT1 $_i$	1.0	1.0	0.5	0.3
		PT2 $_i$	1.0	1.0	0.5	0.3

difficult to optimise this process due to multiple responses to be considered and they consist of the material removal, lap width, and clamp force. Therefore, the optimisation of the surface lapping process or SPL is challenge to the existing ultraprecision technologies. With the multiple optimisation all interesting responses are considered in order to achieve their targets. The desirability function approach in the case of the-nominal-the-best with various shape parameters was applied to compromise those actual responses in terms of the compromise desirability function level or D response. The sequential modified simplex method then merges with the interactive desirability function model (IDFM) instead of the massive contraction and it is called the interactive modified simplex method (IMSM).

The IMSM is used to optimise online the parameter levels of the SLP in order to simultaneously optimise all actual responses or maximise the D response. An additional contribution of this study was the development of artificial intelligence mechanisms of the harmony search and firefly algorithms on all vertices of the simplex design points to search the possible better outcome. An aim is to avoid getting the unsatisfactory D responses to continue the sequential

TABLE 10: Desired or major quality characteristics at the current (CUR) and new (NEW) operating condition.

Major quality characteristics	Sample mean		Sample standard deviation		Ppk	
	CUR	NEW	CUR	NEW	CUR	NEW
Material removal	1.383	0.760	0.134	0.097	0.91	2.01
Lap width	0.767	0.635	0.028	0.035	0.95	1.41
Clamp force	26.54	27.28	0.585	0.465	0.88	1.58

procedures of the conventional MSM such as the reflection, the contraction to the reflected, or the worst vertex. To evaluate the performance of the IMSM, numerical experiments on the SLP were presented within four cycles. From the DM and his team, there were bound adjustment, tightening, and relaxation modes based on the shape, bound, and target of the IDFM for the IMSM. The experimental results showed that according to the highest level of the D response of 0.9734, the preferable levels of process parameters of lapping time, lapping speed, downward pressure, and charging pressure from the fourth cycle of the IDFM were 33, 35, 9.0, and 5.0, respectively.

As expected, there is a level decrease of process parameters of the lapping time and charging pressure in the course of the IMSM evolution. Indeed the IMSM provides the lower levels of the lapping time, higher levels of lapping speed, and slightly higher levels of downward pressure in order to compromise all the responses by decreasing the material removal and increasing the lap width and clamp force. The second reason is that these levels have an additional influence on the better surface quality. Lower levels of charging pressure are necessary to avoid the material removal drifting. The material removal can meet the process specification with the levels of mean, standard deviation, and Ppk at 0.760, 0.097, and 2.01, respectively. This operating condition achieved the proper levels of the lap width dimension with the levels of mean, standard deviation, and Ppk at 0.635, 0.035, and 1.41, respectively, and clamp force with the levels of mean, standard deviation, and Ppk at 27.28, 0.465, and 1.58, respectively, when compared to the customer specification. These experimental results allow us to conclude that the interactive modified simplex method with desirability function and artificial intelligence mechanisms is a suitable method for online optimisation in the surface lapping process. It may be easily applied to drive other related processes toward the optimal operating conditions.

In conjunction with the modified simplex method the interactive desirability function model (IDFM) allows the decision maker and his team to apply whenever there is deterioration in the composite desirability function level or D response instead of the conventional massive contraction. They could also adjust any desirability function parameters such as the shape, bound, and target in the preference articulation process (Jeong and Kim, 2009). The IDFM can be effective in providing alternatives of compromise design points that is faithful to the preference structure. Moreover, metaheuristic mechanisms of the harmony search and firefly algorithms are also adapted to this interactive modified simplex method before forming the simplex. Other applications of the optimised parameter levels or other classes

of artificial intelligence techniques to the IMSM are left for future study. Although the proposed modified simplex method can provide the enhancement during the multiple response surface optimisation, it could be expected that the fruitful results will be obtained by combining another family of sequential procedures based on the simplex designs in forms of new evolutionary methods.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the Higher Education Research Promotion and National Research University Project of Thailand, Office of Higher Education Commission.

References

- [1] J. W. Cha, S. C. Hwang, and E. S. Lee, "Evaluation of Y_2O_3 surface machinability using ultra-precision lapping process with IED," *Journal of Mechanical Science and Technology*, vol. 23, no. 4, pp. 1194–1201, 2009.
- [2] Z. Qiu, F. Z. Fang, L. Ding, and Q. Zhao, "Investigation of diamond cutting tool lapping system based on on-machine image measurement," *The International Journal of Advanced Manufacturing Technology*, vol. 56, pp. 79–86, 2011.
- [3] I. D. Marinescu, E. Uhlmann, and T. K. Doi, *Handbook of Lapping and Polishing*, Taylor & Francis, 2007.
- [4] J.-D. Kim and M.-S. Choi, "A study on the optimization of the cylindrical lapping process for engineering fine-ceramics (Al_2O_3) by the statistical design method," *Journal of Materials Processing Technology*, vol. 52, no. 2–4, pp. 368–385, 1995.
- [5] H. L. Lin, "Optimizing the auto-brazing process quality of aluminum pipe and flange via a Taguchi-Neural-Genetic approach," *Journal of Intelligent Manufacturing*, vol. 23, no. 3, pp. 679–686, 2012.
- [6] H. L. Lin, "The use of the Taguchi method with grey relational analysis and a neural network to optimize a novel GMA welding process," *Journal of Intelligent Manufacturing*, vol. 23, no. 5, pp. 1671–1680, 2012.
- [7] T.-R. Lin, "A comparative study between smaller-the-better and normal-the-best quality characteristics when lapping ceramic blocks," *International Journal of Advanced Manufacturing Technology*, vol. 31, no. 1–2, pp. 155–163, 2006.
- [8] Y. Zhang, I. D. Marinescu, and R. Vandenboom, "Optimisation of D2 steel lapping with a polymer plate," *International Journal of Abrasive Technology*, vol. 3, no. 3, pp. 203–214, 2010.

- [9] R. Ramezani and M. Saidi-Mehrabad, "Hybrid simulated annealing and MIP-based heuristics for stochastic lot-sizing and scheduling problem in capacitated multi-stage production system," *Applied Mathematical Modelling*, vol. 37, no. 7, pp. 5134–5147, 2013.
- [10] S. Salcedo-Sanz, D. Manjarrés, Á. Pastor-Sánchez, J. del Ser, J. A. Portilla-Figueras, and S. Gil-López, "One-way urban traffic reconfiguration using a multi-objective harmony search approach," *Expert Systems with Applications*, vol. 40, no. 9, pp. 3341–3350, 2013.
- [11] T. V. Sibalija and V. D. Majstorovic, "An integrated approach to optimise parameter design of multi-response processes based on Taguchi method and artificial intelligence," *Journal of Intelligent Manufacturing*, vol. 23, no. 5, pp. 1511–1528, 2012.
- [12] A. C. C. Carlos and L. B. Ricardo, "Evolutionary multiobjective optimization in materials science and engineering," *Materials and Manufacturing Processes*, vol. 24, no. 2, pp. 119–129, 2009.
- [13] I. J. Jeong and K. J. Kim, "Stochastics and Statistics: an interactive desirability function method to multiresponse optimisation," *European Journal of Operational Research*, vol. 195, no. 2, pp. 412–426, 2009.
- [14] A. Singh, S. Datta, S. S. Mahapatra, T. Singha, and G. Majumdar, "Optimization of bead geometry of submerged arc weld using fuzzy based desirability function approach," *Journal of Intelligent Manufacturing*, vol. 24, no. 1, pp. 35–44, 2013.
- [15] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 36–38, pp. 3902–3933, 2005.
- [16] E. Cuevas, "Block-matching algorithm based on harmony search optimisation for motion estimation," *Applied Intelligence*, vol. 39, no. 1, pp. 165–183, 2013.
- [17] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Press, Beckington, UK, 2008.
- [18] X.-S. Yang, "Firefly algorithms for multimodal optimization," in *Stochastic Algorithms: Foundations and Applications*, vol. 5792 of *Lecture Notes in Computer Sciences*, pp. 169–178, Springer, Berlin, Germany, 2009.
- [19] M. T. Vakil Baghmisheh, M. Peimani, M. H. Sadeghi, M. M. Etefagh, and A. F. Tabrizi, "A hybrid particle swarm-Nelder-Mead optimization method for crack detection in cantilever beams," *Applied Soft Computing Journal*, vol. 12, no. 8, pp. 2217–2226, 2012.
- [20] V. C. Mariani, L. G. Justo Luvizotto, F. A. Guerra, and L. Dos Santos Coelho, "A hybrid shuffled complex evolution approach based on differential evolution for unconstrained optimization," *Applied Mathematics and Computation*, vol. 217, no. 12, pp. 5822–5829, 2011.

Research Article

Swarm Intelligence-Based Hybrid Models for Short-Term Power Load Prediction

Jianzhou Wang,¹ Shiqiang Jin,² Shanshan Qin,³ and Haiyan Jiang²

¹ School of Statistics, Dongbei University of Finance and Economics, Dalian 116025, China

² School of Mathematics & Statistics, Lanzhou University, Lanzhou 73000, China

³ MOE Key Laboratory of Western China's Environmental Systems, Research School of Arid Environment & Climate Change, Lanzhou University, Lanzhou 73000, China

Correspondence should be addressed to Shiqiang Jin; jinsql@gmail.com

Received 6 June 2014; Revised 18 July 2014; Accepted 1 August 2014; Published 30 September 2014

Academic Editor: Fang Zong

Copyright © 2014 Jianzhou Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Swarm intelligence (SI) is widely and successfully applied in the engineering field to solve practical optimization problems because various hybrid models, which are based on the SI algorithm and statistical models, are developed to further improve the predictive abilities. In this paper, hybrid intelligent forecasting models based on the cuckoo search (CS) as well as the singular spectrum analysis (SSA), time series, and machine learning methods are proposed to conduct short-term power load prediction. The forecasting performance of the proposed models is augmented by a rolling multistep strategy over the prediction horizon. The test results are representative of the out-performance of the SSA and CS in tuning the seasonal autoregressive integrated moving average (SARIMA) and support vector regression (SVR) in improving load forecasting, which indicates that both the SSA-based data denoising and SI-based intelligent optimization strategy can effectively improve the model's predictive performance. Additionally, the proposed CS-SSA-SARIMA and CS-SSA-SVR models provide very impressive forecasting results, demonstrating their strong robustness and universal forecasting capacities in terms of short-term power load prediction 24 hours in advance.

1. Introduction

With the additional types of energy integration into the power grid and the development of generation technologies, power utilities are going through a crucial challenge stemming from maintaining the desired security and reliability of the electricity supply [1]. Various technologies in power generation, transmission, distribution, and utility are researched by organizations, and the power grid is facing a major change. The smart grid, an internationally popular topic recently, is pushing the power grid into an open system with the characteristics of robustness and dynamics, which provides a great chance to improve and enhance the load demand response programs [2]. However, the high accuracy of load prediction is the key factor in a power intelligence system, which determines the quality of the smart grid. If the load demands are overestimated, it will induce a conservative operation with excessive energy purchased, thereby resulting in energy waste and unnecessary cost. It has been estimated

that a 1% increase in the forecasting error will result in a 10 million dollar increase in operation costs [3]. If the load demands are underestimated, it will induce a risky operation with a deficient preparation of the spinning reserve, causing the power system to operate in a vulnerable region to the disturbance and power cut [4].

Therefore, according to the aforementioned analysis, the power load prediction should be urgently conducted with high accuracy to guarantee the operational performance of the power system. Over the past few decades, large efforts have been devoted to improving the power load forecasting accuracy. The various methods utilized for load prediction range from the traditional statistical models to the complicated artificial intelligence-based models [5]. For intelligence-based methods, the evolutionary programming, expert systems, artificial neural networks (ANN), and fuzzy inferences are included [6]. Among the intelligent approaches, the ANN method has received more attention because of its good flexibility, easy implementation, and nonlinear

mapping ability between the powers loads and weather variables such as humidity, temperature, wind speed, and historical load patterns [7]. Moreover, to perform a better training process and improve the load forecasting accuracy, the ANN is usually exploited in a hybrid model by combining it with other methods or techniques, such as the support vector regression (SVR). The SVR [8] model is used successfully to address forecasting problems in many fields, such as the financial time series (exchange rate and stocks index) prediction [9, 10], hydroinformatic forecasting [11, 12], and tourist arrival forecasting [13]. Additionally, the SVR model is also successfully applied in forecasting the power load [14–17]. The empirical results demonstrate that the selection of the two parameters C (to trade off the training errors and large weights) and σ (the parameter for the Gaussian kernel function) in an SVR model influences the forecasting accuracy considerably. To conquer the difficult problems in the selection of the parameters for the SVR, the author conducted a series of relevant experiments by employing a hybrid sequence with swarm intelligent optimization algorithms for the parameter determination to overcome the problem of immature convergence (trapped in local optimum) [11, 15, 17]. To continue testing the superiority of the hybrid sequence with swarm intelligent optimization algorithms, this paper tried to employ the cuckoo search (CS) and particle swarm optimization (PSO) algorithm to determine the values of three parameters in an SVR model.

The cuckoo search (CS) algorithm is a new optimization metaheuristic algorithm [21] based on a stochastic global search and the obligate brood-parasitic behavior of cuckoos in combination with the Lévy flight behavior of several birds and fruit flies. It is widely and successfully used in a number of practical problems, such as knapsack problems [22], software test effort estimations [23], scheduling problems [24], test sequence optimization problems [25], convex and nonconvex ED (economic dispatch) problems, and microgrid power dispatch problems [26]. In [26], the author implemented the CS algorithm to solve both the convex and nonconvex ED problems and the micro grid power dispatch problem. Moreover, the author compared the CS algorithm with many other artificial optimization algorithms, such as simulated annealing (SA), evolutionary programming (EP), genetic algorithm (GA), PSO, differential evolution (DE), and bacterial foraging algorithm (BFA). It is seen that the proposed CS algorithm has the ability to converge to a better quality solution than all of the artificial optimization algorithms mentioned above. One of the main advantages of the CS algorithm is that there are fewer parameters to be fine-tuned in the CS algorithm than in the GA and PSO [21]. In [21], Yang and Deb formulated the CS algorithm to search the minimum values of the multimodal objective functions: the bivariate Michalewicz function, De Jong's first function, Shubert's bivariate function, Griewangk's test function, Ackley's function, generalized Rosenbrock's function, Schwefel's test function, Rastrigin's test function, and Michalewicz's test function. Then, by comparing the results searched by the CS algorithm with the PSO and GA, Yang and Deb concluded that the CS algorithm is superior to these existing algorithms for multimodal objective functions. The preliminary studies indicate that it is very

promising and could outperform the existing algorithms, such as the artificial bee colony (ABC), GA, PSO, bacterial foraging (BF), ant colony optimization (ACO), and honeybee mating optimization (HBMO) [27–38]. However, few papers employ the CS algorithm to optimize the parameters of the SARIMA and compare the convergence speed with the PSO, which has a fast convergence speed. In this paper, we conduct the CS algorithm to optimize the parameters of the SVR and SARIMA and meanwhile compare the accuracy of the forecast and convergence speed of the CS algorithm with that of the PSO in predicting the power load.

The most popular and classic statistical models include the linear or nonlinear regression models, time series models, state estimation, and Kalman filtering technology [7]. The multiregression models consider certain factors as the explanatory variables, such as the weather factors, climatic conditions, social activities, and seasonal factors, with the regression coefficients estimated by the least squares estimation or modern regression method. The time series models assume that the load is related to its past values, which can be regarded as an autoregressive process that is therefore forecasted by time series models. Based on the aforementioned literature, however, most studies have been regardless of the seasonal patterns, the fluctuation of which can lead to a deviation in the load forecasts. To solve this, Wang et al. [39] proposed a hybrid model by incorporating the SARIMA and ANN to address the periodic relationships and nonlinear patterns, respectively. SARIMA [40, 41] mainly addresses the linear relationships and considers the periodicity of the time series in a real-life scenario. Considering, insufficient work that is related to the periodicity of load data has been performed. Therefore, a new hybrid model based on SARIMA is proposed in this paper.

The SSA [42, 43] is a powerful and novel technique of time-series analysis and forecasting incorporating the elements of classical time-series analysis, multivariate geometry, multivariate statistics, dynamical systems, and signal processing [44]. The main purpose of the SSA is to develop a decomposition of the raw time series into the sum of several interpretable and independent components, such as a gradually varying trend, oscillatory components (periodic or quasi-periodic components) and a structure-less noise. Then, several of these components are used for time series forecasting. The SSA technique has been successfully used in several fields such as geophysics, hydrology, climatology [45, 46], and economics [47, 48]. Golyandina and Korobeynikov [49] described very detailed steps to show how the methodology of the SSA analysis, forecasting, and parameter estimation can be implemented with the help of the package *Rssa*. Chen et al. [50] assessed the value of the SSA for extracting the time-variable seasonal signals from the GPS time series and compared the SSA-based results to the least-squares analysis and Kalman filtering. The results demonstrate that the SSA is a viable and complementary tool for extracting modulated oscillations from the GPS time series. In [51], Wu and Chau attempts to eliminate the lag effect, often appearing in the ANN modeling process, by using the singular spectrum analysis (SSA). Two watersheds from China are explored with daily collected data. The results demonstrate that the SSA

can significantly enhance the performance of the prediction model and eliminate the lag effect.

If the original load data are directly applied to a train model without eliminating noise, the high-frequency components may disturb the forecasted load patterns [6]. To improve the load forecasting accuracy, a wavelet denoising technique is usually adopted to extract the low frequency component of the load pattern. Therefore, these data are utilized in load forecasting methods. However, many parameters should be determined during the procedure of wavelet analysis, such as the determination of the decomposition layers, wavelet basis function, and threshold function; furthermore, each of the parameters is a large amount and is hard to determine objectively. Because of the subjective selection of the parameters for the wavelet, different researchers obtain various denoising effects; thus, researchers often need to conduct a large amount of data experiments to gain a satisfactory result before continuing the study. However, one of the major advantages of the SSA compared to other approaches is that only two parameters are required to model the time series under analysis [44]. It is a relatively new data-driven or nonparametric technique developed to model a nonlinear and/or nonstationary as well as noisy short time series [52]. Moreover, it is considered from Claudio [44] that the SSA does not depend on a priori defined functions, such as the Fourier approach (based on sine and cosine functions), but it generates a set of components directly from the time series under study. Additionally, the SSA technique can compute periodic or quasiperiodic components and a slowly changing trend. To the best of our knowledge, the SSA is not applied to denoise the power load time series. Most of the references predict the components that are decomposed by the SSA and then reconstruct them. In this paper, we denoise the power load by the SSA and then conduct the linear and nonlinear methods to validate whether this denoising technique can help the linear and nonlinear models further improve the accuracy of forecasting.

This paper starts with a brief description of the related methodology in Section 2, specifies the procedure of the SSA in Section 3.1, and presents the results of the case study for simulating the power load in Section 3.2.

2. Related Methodology

In this section, SSA, SARIMA, the SVR model, the SI algorithm (CS and PSO), and the design of the proposed hybrid SI-based predictive models are summarized as the foundation to construct the proposed hybrid model.

2.1. Singular Spectral Analysis (SSA) Technique. The SSA technique, a well-developed method of time series analysis, can extract major information from a time series, such as the trend and periodicities components without prior knowledge regarding the trend as well as period values [50]. In this section, the information regarding the SSA, which is critical for understanding the implementations of the SSA, is described, and the SSA tool is used for the analysis of the power load series.

The basic SSA consists of two complementary parts: decomposition and reconstruction. For the decomposition part, it comprises two steps: embedding and singular values decomposition. For the reconstruction part, two steps are also involved, which are the Eigentriple grouping and diagonal averaging [51]. Assume a time series $X = \{x_1, x_2, \dots, x_N\}$. The processes of the SSA are given as follows [49].

Part 1 (decomposition). Consider the following.

Step 1 (embedding). The original time series X is mapped into a sequence of multidimensional lagged vectors of size L by performing the embedding procedure

$$X_i = (x_1, x_2, \dots, x_{i+L-1})^T, \quad i = 1, 2, \dots, K, \quad (1)$$

where L ($1 < L < N$) is a positive integer called the window length and the delay time is $K = N - L + 1$. The trajectory matrix of the series X is

$$\mathbf{X} = (x_{ij})_{i,j=1}^{L,K} = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_K \\ x_2 & x_3 & x_4 & \dots & x_{K+1} \\ x_3 & x_4 & x_5 & \dots & x_{K+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & x_{L+2} & \dots & x_N \end{bmatrix}. \quad (2)$$

For the trajectory matrix \mathbf{X} , there are two vital properties, which are the following.

- (i) Both the columns and rows of \mathbf{X} are a subseries of the original series X .
- (ii) The trajectory matrix \mathbf{X} is Hankel with equal elements on the antidiagonals.

Step 2 (singular values decomposition (SVD)). Decompose the trajectory matrix \mathbf{X} by using the following:

$$\mathbf{X} = \sum_{i=1}^L P_i Q_i^T = \mathbf{X}_1 + \mathbf{X}_2 + \dots + \mathbf{X}_L, \quad (3)$$

where $\{P_i\}_{i=1}^L$ is an orthonormal basis in R^L and $Q_i = \mathbf{X}^T P_i$. For the orthonormal basis $\{P_i\}_{i=1}^L$, here we have two versions for consideration:

- (A) basic: $\{P_i\}_{i=1}^L$ are the eigenvectors of $\mathbf{X}\mathbf{X}^T$,
- (B) Toeplitz: $\{P_i\}_{i=1}^L$ are the eigenvectors of the matrix \mathbf{C} , which is given by

$$c_{ij} = \frac{1}{N - |i - j|} \sum_{m=1}^{N-|i-j|} x_m x_{m+|i-j|}, \quad 1 \leq i, j \leq L. \quad (4)$$

In both cases, the eigenvectors are ordered, which can thereby guarantee that the corresponding eigenvalues are placed in decreasing order.

Note that the Case (B) version is only suitable for the analysis of the stationary time series with mean value zero.

Note also that the Case (A) version corresponds to the SVD of \mathbf{X} ; that is, $\mathbf{X} = \sum_i \sqrt{\lambda_i} U_i V_i^T$, $P_i = U_i$ are the left

singular vectors of \mathbf{X} . Consider $Q_i = \sqrt{\lambda_i}V_i$, where λ_i are the eigenvalues of $\mathbf{X}\mathbf{X}^T$. $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_L \geq 0$ and V_i are the orthonormal systems of the eigenvectors corresponding to these eigenvalues as well as the right singular vectors of \mathbf{X} . The collection (λ_i, U_i, V_i) is called the i th eigentriple of the SVD.

Part 2 (reconstruction). Consider the following.

Step 3 (eigentriple grouping). Let $d = \max\{j : \lambda_j \neq 0\}$. After decomposition, the grouping procedure splits the set of indices $\{1, \dots, d\}$ into m disjoint subsets I_1, \dots, I_m . Let $I = \{I_1, I_2, \dots, I_m\}$, and define the resultant matrix \mathbf{X}_I corresponding to the group I by the following equation:

$$\mathbf{X}_I = \mathbf{X}_{I_1} + \mathbf{X}_{I_2} + \dots + \mathbf{X}_{I_m}. \quad (5)$$

The above procedure of selecting the sets I_1, \dots, I_m is called the eigentriple grouping. If $m = d$ and $I_j = \{j\}$, $j = 1, \dots, d$, then the corresponding grouping is called elementary. The selection of several leading eigentriples for the Case (A) version determines the approximation of the original time series and corresponds to the well-known optimality property of the SVD.

Step 4 (diagonal averaging). In this step, each matrix \mathbf{X}_{I_j} is transformed into a new series of length N . Let \mathbf{Y} be the $L \times K$ matrix with elements y_{ij} , $1 \leq i \leq L$, $1 \leq j \leq K$, and $L \leq K$. By performing diagonal averaging, we transfer the matrix \mathbf{Y} into the series $(\hat{y}_1, \hat{y}_2, \dots, \hat{y}_N)$ using the following:

$$\hat{y}_s = \sum_{(l,k) \in A_s} \frac{y_{lk}}{|A_s|}, \quad (6)$$

where $A_s = \{(l, k) : l + k = s + 1, 1 \leq l \leq L, 1 \leq k \leq K\}$ and $|A_s|$ denote the number of elements in the set A_s . The diagonal averaging utilized for a resultant matrix \mathbf{X}_{I_k} generates a reconstructed series $\widehat{X}^{(k)} = (\widehat{X}_1^{(k)}, \widehat{X}_2^{(k)}, \dots, \widehat{X}_N^{(k)})$. Then, the initial series $X = \{x_1, x_2, \dots, x_N\}$ is decomposed into a sum of m reconstructed series:

$$x_n = \sum_{k=1}^m \widehat{X}_n^{(k)}, \quad n = 1, 2, \dots, N. \quad (7)$$

The reconstructed series produced by the elementary grouping is called the elementary reconstructed series.

The SSA is a data-driven technique that can extract information from a short and noisy time series without prior knowledge of the dynamics affecting the time series. A significant characteristic of the SSA is that trend patterns obtained in this way are not necessarily linear [50]. Most importantly, the intricate periodicities lying in the time series can be modulated and extracted out.

2.2. Seasonal Autoregressive Moving Average (SARIMA). The time series that mainly contains the periodic and stochastic components can be forecasted by the SARIMA model, which is the most popular linear model for a seasonal time series and has achieved great success in both academic research and industrial applications over the past few decades [53].

A time series $\{X_t \mid t = 1, 2, \dots, N\}$ is generated by a SARIMA $(p, d, q)(P, D, Q)_s$ process if

$$\phi_p(B) \Phi_p(B^s) (1-B)^d (1-B^s)^D X_t = \theta_q(B) \Theta_Q(B^s) \varepsilon_t, \quad (8)$$

where p, d, q, P, D, Q are integers and s is the periodicity length. Consider the following:

$$\begin{aligned} \phi_p(B) &= 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p, \\ \Phi_s(B^s) &= 1 - \Phi_s B^s - \Phi_{2s} B^{2s} - \dots - \Phi_{Ps} B^{Ps}, \\ \theta_q(B) &= 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q, \\ \Theta_Q(B^s) &= 1 - \Theta_s B^s - \Theta_{2s} B^{2s} - \dots - \Theta_{Qs} B^{Qs}, \end{aligned} \quad (9)$$

are polynomials in B of degree p, q, P , and Q . Herein, B, d , and D denote the backward shift operator, the regular differential order, and the seasonal differential order, respectively. ε_t and X_t are the estimated residual and the observed value at time t ; $t = 1, 2, \dots, k$, separately. The residual series should be identical and independent as a white noise with an average value equal to zero and a constant variance value.

When fitting a SARIMA model, the following four steps are involved [54].

Step 1 (model identification). Identify the SARIMA $(p, d, q)(P, D, Q)_s$, which determines the most relevant combination of the autoregression and moving average process.

Step 2 (model estimation). These parameters are estimated and determined by the maximum likelihood estimation (MLE).

Step 3 (model validation). The precision of the chosen model is tested, and possible enhancements are also established during this step.

Step 4 (model forecasting). The future values that are reforecasted based on the constructed SARIMA $(p, d, q)(P, D, Q)_s$.

2.3. Basic Description of the Support Vector Regression (SVR). The SVR is an adaptation of a recently developed machine learning theory (MLT) known as the support vector machine (SVM) proposed by Vapnik et al. [55]. In the SVR model, a regression function $y = f(x)$ is fit, and then it is applied to predict the outputs based on a new input set. A brief review of the SVR is introduced as follows [56].

Step 1. A nonlinear mapping $\varphi(\cdot) : \mathcal{R}^n \rightarrow \mathcal{R}^{n_h}$ is defined to solve a nonlinear regression problem by mapping the training sets $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ into a high dimensional feature space \mathcal{R}^{n_h} .

Step 2. In the high dimensional feature space, the nonlinear regression problem in the lower dimension space is transformed into a linear one by a linear function, namely, the SVR function

$$f(\mathbf{x}) = \mathbf{w}^T \cdot \varphi(\mathbf{x}) + b, \quad (10)$$

where $f(\mathbf{x})$ denotes the forecasting values; the coefficients \mathbf{w} ($\mathbf{w} \in \mathcal{R}^{n_h}$) and b ($b \in \mathcal{R}$) are adjustable.

Step 3. Define the empirical risk $R_{\text{emp}}(f)$:

$$R_{\text{emp}}(f) = \frac{1}{N} \sum_{i=1}^N \Theta_{\varepsilon}(y_i, \mathbf{w}^T \varphi(\mathbf{x}_i) + b), \quad (11)$$

where $\Theta_{\varepsilon}(y, f(\mathbf{x}))$ is the ε -intensive loss function and is given by

$$\Theta_{\varepsilon}(y, f(\mathbf{x})) = \begin{cases} |f(\mathbf{x}) - y| - \varepsilon, & \text{if } |f(\mathbf{x}) - y| \geq \varepsilon \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

The ε -intensive loss function is utilized to control the sparsity of the solutions and generalization of the models.

Step 4. Determine the training overall errors between the training data and ε -insensitive loss function, which can be defined as a quadratic optimization problem with inequality constraints. Consider the following:

$$\text{Min}_{\mathbf{w}, b, \xi_i^*, \xi_i} R_{\varepsilon}(\mathbf{w}, \xi_i^*, \xi_i) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^N (\xi_i + \xi_i^*). \quad (13)$$

Subject to

$$\begin{aligned} y_i - \mathbf{w}^T \varphi(\mathbf{x}_i) - b &\leq \varepsilon + \xi_i^*, \quad i = 1, 2, \dots, N, \\ \mathbf{w}^T \varphi(\mathbf{x}_i) - y_i + b &\leq \varepsilon + \xi_i, \quad i = 1, 2, \dots, N, \\ \xi_i^*, \xi_i &\geq 0, \quad i = 1, 2, \dots, N. \end{aligned} \quad (14)$$

The first term in (13) is employed to regularize weight sizes, to penalize large weights, and to maintain regression function flatness. The second term in (13) penalizes the training errors of $f(\mathbf{x})$ and \mathbf{y} by exploiting the ε -intensive loss function. Herein, C is a parameter to balance these two terms. Training errors below $-\varepsilon$ are denoted as ξ_i . Otherwise, they are denoted as ξ_i^* .

Step 5. Obtain the parameter vector \mathbf{w} by solving the quadratic optimization problem defined in Step 4:

$$\mathbf{w} = \sum_{i=1}^N (\beta_i^* - \beta_i) \varphi(\mathbf{x}_i), \quad (15)$$

where β_i^*, β_i are the Lagrangian multipliers.

Step 6. Establish the SVR regression function by using the following equations:

$$f(\mathbf{x}) = \sum_{i=1}^N (\beta_i^* - \beta_i) K(\mathbf{x}_i, \mathbf{x}_j) + b, \quad (16)$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2), \quad \gamma > 0,$$

where $K(\mathbf{x}_i, \mathbf{x}_j)$ is the kernel function and $K(\mathbf{x}_i, \mathbf{x}_j) = \varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j)$. Several types of kernel functions can be selected to build the model. However, the most commonly used kernel functions are the Gaussian radial basis functions (RBF) and the polynomial kernel functions. Until now, it has been difficult to determine which type of kernel functions for specific

data patterns is suitable [57]. In this paper, the RBF is selected as the kernel function because of its easy implementation and its strong capability of nonlinearly mapping the training sets into an infinite dimensional space, which is suitable to handle nonlinear relationship problems. Therefore, the Gaussian RBF kernel function is specified in this study.

2.4. *Swarm Intelligence Optimization Algorithms.* In recent years, the metaheuristic optimization algorithms and evolutionary computation have been a noticeable part for solving real mathematics and engineering problems [19], especially in the field of parameter determination. In this section, several of the optimization algorithms utilized in this study are described briefly.

2.4.1. *Cuckoo Search (CS).* The CS algorithm, inspired by the breeding behavior of cuckoos, is a recently developed metaheuristic algorithm by Yang and Deb [21]. For the CS algorithm, two behaviors are adapted and combined from nature that fulfills the criteria of a metaheuristic algorithm, which are described as follows [22–24].

Breeding Behavior. Many species of cuckoos lay their eggs in communal nests, but to increase the hatching probability of their own eggs, they always remove other cuckoos' eggs. Once a host cuckoo discovers an alien egg (does not belong to itself), then it will either throw the egg away or discard the current nest and build another nest elsewhere. For the CS algorithm in each step with the new solutions generated, the poorer solutions are abandoned.

Lévy Flight. Generally, the flight path of many birds is effectively a random walk, which is representative of Lévy flights, with a step length drawn from the Lévy distribution. In the CS-based algorithm for producing a new solution $x^{(t+1)}$ for a cuckoo, a Lévy flight is defined as the following expression:

$$x_i^{t+1} = x_i^t + \alpha \oplus \text{Lévy}(\lambda), \quad (17)$$

$$\alpha = \alpha_0 (x_j^{(t)} - x_i^{(t)}), \quad (18)$$

$$\text{Lévy}(\lambda) \sim l^{-\lambda}, \quad (1 < \lambda < 3), \quad (19)$$

where x_i^t is the eggs (samples), i is the sample size, t is the iterations, and α is the step size that is mostly utilized as in (18). The symbol \oplus denotes an entry-wise multiplication, while the Lévy(λ) values are found in the Lévy distribution defined in (19).

Figure 1 illustrates the basic steps of the CS algorithm and Lévy flight together with a detailed immigration of a cuckoo toward a goal habitat. More information regarding the CS method can be found in literature [21].

2.4.2. *Particle Swarm Optimization (PSO).* The PSO algorithm, inspired by the social behaviors of animal movements, investigates the search space by applying a flock of potential solutions named particle swarms, characterized by their corresponding position and velocity [58]. The individual

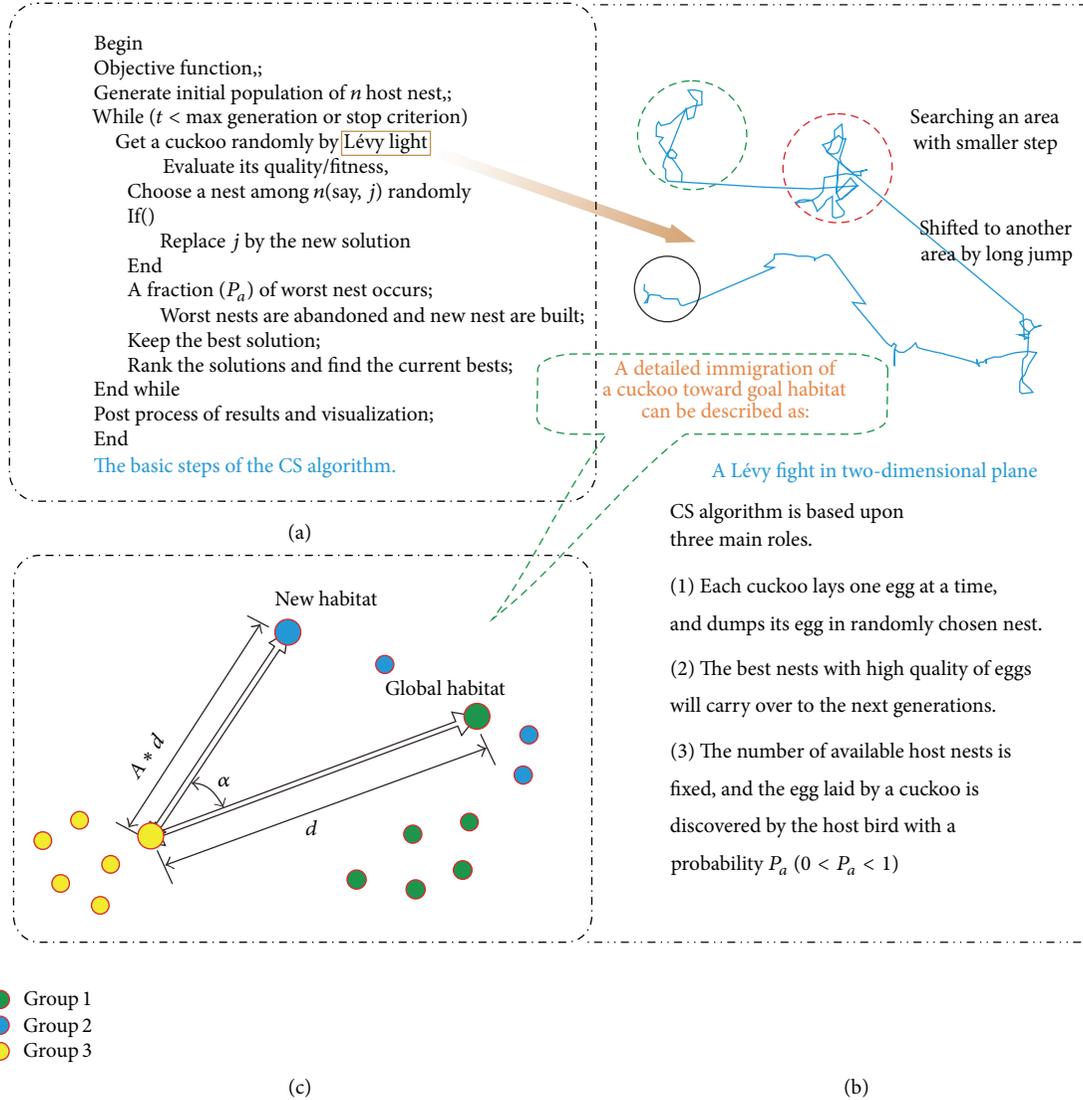


FIGURE 1: The basic steps of the CS algorithm and Lévy flight together with a detailed immigration of a cuckoo toward a goal habitat [18–20]. In this paper, we set the initial population size as 25 and the iteration to 100; an alien egg laid by a cuckoo bird is discovered by the host cuckoo with a probability $P_a = 0.25$.

particles' position and the velocity are given by (20) and (21), respectively [20],

$$x_i^{k+1} = x_i^k + v_i^k, \quad (20)$$

$$v_i^{k+1} = \omega v_i^k + c_1 r_1 (P_{\text{best},i} - x_i^k) + c_2 r_2 (G_{\text{best}} - x_i^k), \quad (21)$$

where ω is the inertia weight; c_1 and c_2 denote the acceleration constants; $P_{\text{best},i}$ and G_{best} are the individual and global best position, separately. For each step, the velocity of each particle is changed toward the best local and global locations. The process is repeated until a predefined termination condition is reached. More details on the PSO are available in literature [32]. In this paper, we set the initial population size as 25 and the iteration as 100, which are the same as the CS algorithm, and the acceleration c_1 and c_2 equals 1.49455.

2.5. Design of the Proposed Hybrid SI-Based Predictive Models.

The original power load data have some noisy information. If not eliminating noise and directly training and building models, the high frequency components may disturb the forecasted load patterns [6]. To improve the load forecasting accuracy, the SSA technique is adopted to extract the trend and periodic components and remove the useless information from the load pattern. After this procedure of data preprocessing, a smoother power loads time series is obtained for forecasting. Furthermore, it can also calculate the period. It is well known that the predictive accuracy of an SVR model largely relies on a reasonable setting of the kernel parameter σ and hyperparameters C . Therefore, the determination of two parameters is a significant issue [59]. However, there is no structural approach or any shortage of opinions on the efficient selection of SVR parameters. The traditional

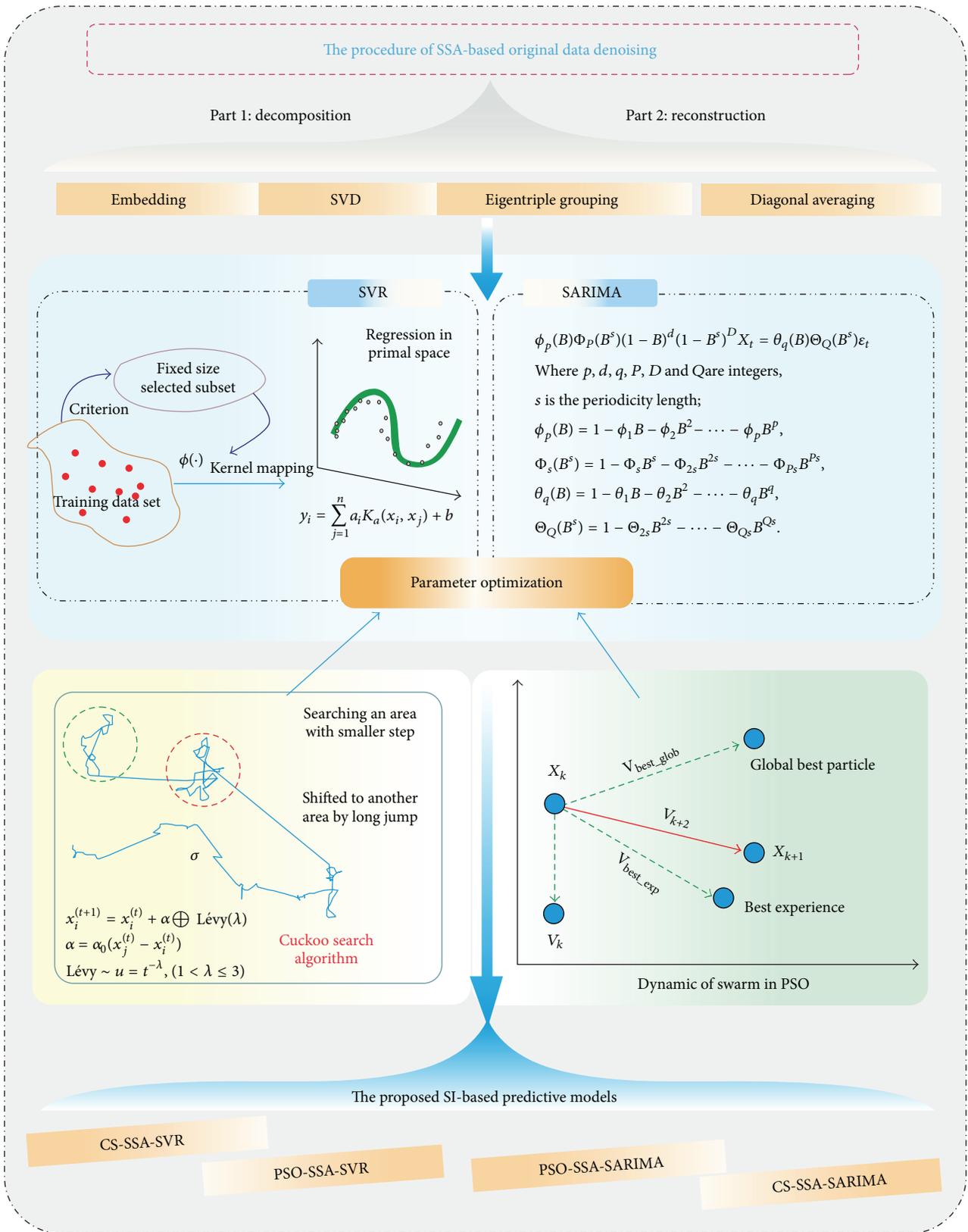


FIGURE 2: The structure of the proposed SI-based hybrid models.

method to determine the parameters of the SVR is a grid search, which is time-consuming and not effective enough to obtain a satisfactory result. In this study, the artificial intelligence optimization (AIO) algorithm, that is, the CS and PSO, is adopted to optimize the parameter selection of the proposed SVR model. Additionally, the traditional method of estimating the parameters of SARIMA is the maximum likelihood estimation, which has an assumption of a normal distribution or another known distribution. However, in the real-world scenario, the power load data are not strictly normally distributed. Thus, employing CS and PSO to optimize the parameters of SARIMA is necessary to build a more proper time series model. The structure of the proposed hybrid models is given in Figure 2.

2.6. Evaluation Criteria. The determination of which prediction model outperforms the other models is of prime concern. In most study cases, model performance is evaluated by numerous error evaluation criteria that can be classified into two main types: absolute error and relative error. For the absolute error, there are the mean absolute error (MAE) and root mean square error (RMSE). For the relative error, there are the mean absolute percentage error (MAPE) and symmetrical mean absolute percentage error (SMPAE). All of them are commonly used to evaluate the accuracy. In this paper, the mean absolute error (MAE) and mean absolute percentage error (MAPE) are used to measure the prediction accuracy of these models. The smaller these values are, the better the predictive performance is.

The MAE can be defined as

$$\text{MAE} = \frac{1}{T} \sum_{t=1}^T |y_t - \hat{y}_t|. \quad (22)$$

The MAPE can be defined as

$$\text{MAPE} = \frac{1}{T} \sum_{t=1}^T \left| \frac{y_t - \hat{y}_t}{y_t} \right| \times 100\%, \quad (23)$$

where y_t and \hat{y}_t denote the real and predictive values at time t , respectively.

3. Case Study

As described previously, the SSA technique is widely used in extracting principal information; that is, its trend and oscillation components, which are then effectively used for time series forecasting. Understanding that extracting leading information by the SSA is also a procedure of denoising such as wavelet denoising, in this work, we decompose the power load time series and then reconstruct the principal components into a smoother time series. To demonstrate the performance of the SSA denoising used in the power load, in Section 3.1, we specify the procedure of simulating the power load time series from NSW. In Section 3.2, to eliminate noisy information that may disturb the forecasted accuracy, the results of the SARIMA and SVR prediction after the SSA-based denoising are displayed to test the capability of SSA noise elimination. Moreover, forecasting models optimized

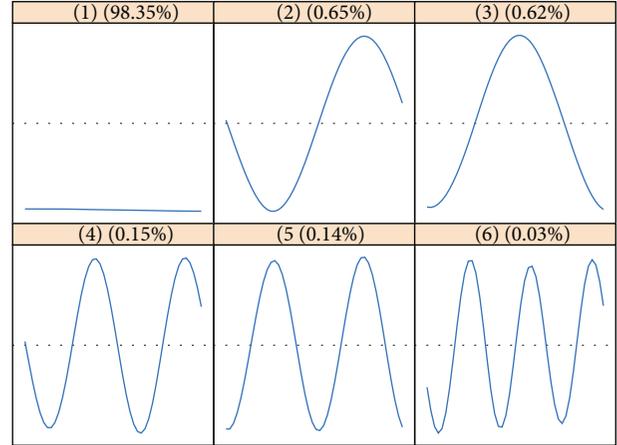


FIGURE 3: 1st stage: eigenvectors ($L = 48$).

by the PSO and CS algorithms are proposed in this section to further improve the predictive ability.

We choose two samples of a half-hour power load, each of which contains 336 training data and 48 test data from April to May in NSW of Australia.

3.1. SSA-Based Denoising. The power load time series in Case 1 is chosen as an example to show the detailed process of SSA-based denoising. The data in Case 2 are denoised by the same procedure as described below. SSA-based denoising has two main stages: the 1st stage is to extract the trend; the 2nd stage is to extract the seasonal components from the residuals from the first step and then reconstruct it.

1st Stage (decomposition). In this stage, the main task is to extract the trend component from the original time series with a small window length. For a varying form of the trend, its extraction is similar to a smoothing effect, and we begin with choosing a possibly minimal window length which in this case is $L = 48$. The SSA with small L performs a smoothing effect of the series by using a filter of order $2L - 1$. The reason we choose this window length is similar to that in moving the averaging procedure: because the smoothing time series includes a periodic component, the window length should be divisible by the period.

Six leading eigenvectors are displayed in Figure 3, which reflects a large contribution of the leading eigentriple. The leading eigenvector contains nearly constant coordinates, and thus, it corresponds to a pure smoothing by the Bartlett filter.

1st Stage (reconstruction). In Figure 4, the result of reconstruction by each of the six eigentriple is illustrated. Combining both figures confirms that the first eigentriple corresponds to the trend, while the rest of the eigentriples contains high frequency components and thus are not related to the first component, the trend. Additionally, Figure 4(1) is roughly considered as the weekly trend, having a share of 98.35% of the power load time series according to Figure 3(1). The trend in Figure 5 is precisely the trend depicted by one

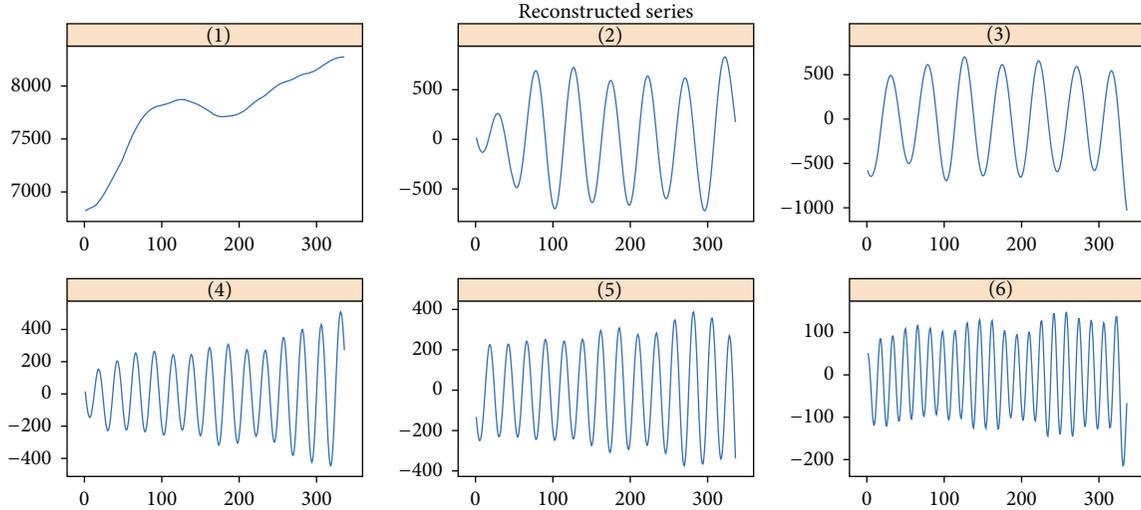


FIGURE 4: 1st stage: elementary reconstructed series ($L = 48$).

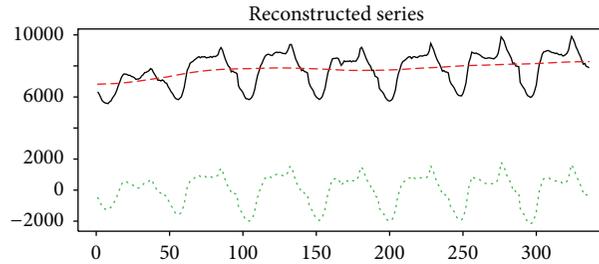


FIGURE 5: 1st stage: initial series, estimated trend, and residuals ($L = 48$, ET1).

leading eigentriple and coincides with the first reconstructed component from Figure 4(1).

2nd Stage (decomposition and visual information). Because we have extracted out the trend in the first stage, this stage is the extraction of seasonality from the residual component depicted as a green dotted line in Figure 5.

To properly identify the sine waves, we use the graph of eigenvalues (Figure 6), scatterplots of eigenvectors (Figure 7), periodogram (Figure 8), and w -correlation matrix of the elementary components (Figure 9). In Figure 6, we see that several steps are obtained by approximately equal eigenvalues. Eight pairs of eigenvectors are illustrated in Figure 7, showing four nearly regular polygons. The number of edges of polygons represents their periods. ET1-2, ET3-4, ET5-6, and ET7-8 correspond to the periods of 48, 24, 16, and 12 in Figure 8, respectively, and correspond to F1-F2, F3-F4, F5-F6, and F7-F8 in Figure 9. These periods are obtained by the seasonality, are clearly explained by the periodogram (Figure 8), and are estimated as 48.14255, 23.94845, 16.04126, and 12.00288. Figure 9 displays the considered pairs of components for a high correlation within and for a low correlation between.

2nd Stage (reconstruction and plotting of the results). The extracted seasonality (ET1-8) is illustrated in Figure 10(c). A slow change in the sine wave phases and amplitudes is

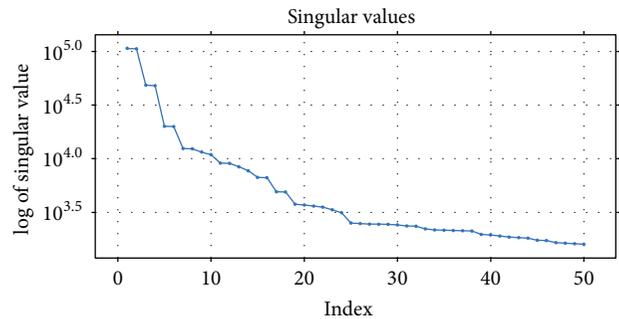


FIGURE 6: 2nd stage: eigenvalues of residuals ($L = 168$).

observed in Figure 7 and produces a periodic performance with a complicated form. Figure 10 demonstrates the resultant decomposition of both stages of SSA. Figure 11 depicts the reconstructed time series in comparison with the actual power load time series.

3.2. Forecasting Results Analysis of Proposed SI-Based Forecasting Models. In this section, we employ SARIMA and SVR to build the forecasting models after denoising the useless information in the power load by SSA. To further improve

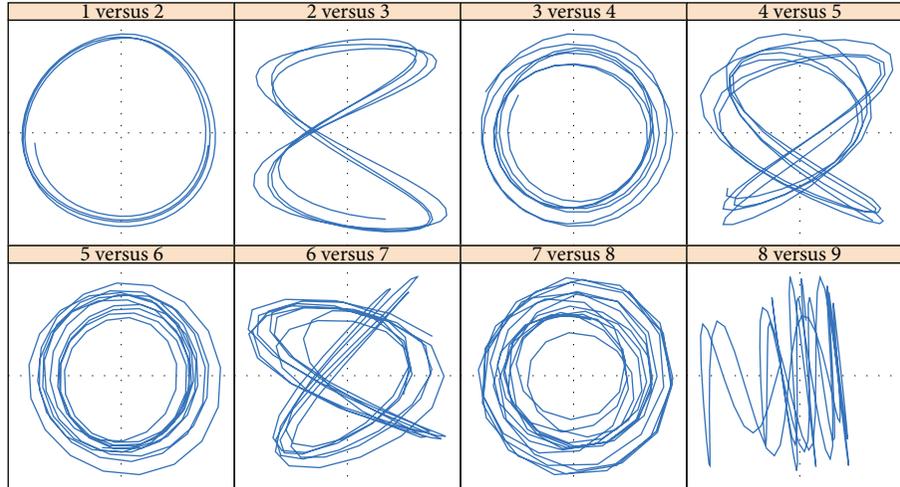


FIGURE 7: 2nd stage: scatterplot for eigenvector pairs ($L = 168$).

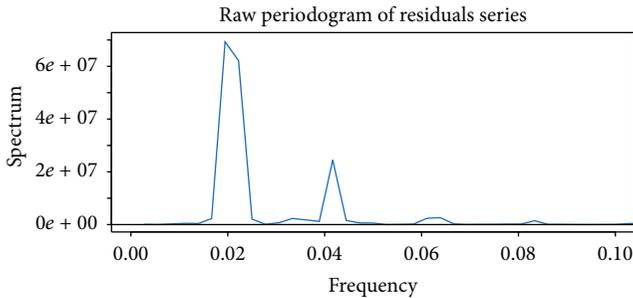


FIGURE 8: 2nd stage: periodogram of the series (i.e., of the residual at the 1st stage).

the accuracy of forecasting, we optimize the parameters of SAMRIA and SVR by the CS and PSO.

The proposed hybrid models are applied for short-term (half an hour) load forecasting with a 48-step ahead of NSW in Australia over a prediction of one day in two cases. The performance of the proposed methods is marked as CS-SSA-SARIMA and CS-SSA-SVR. The comparisons of the power load forecasting results are intuitively shown in Figures 12 and 13. The forecasts for Case 1 based on the PSO-SSA-SARIMA, CS-SSA-SARIMA, PSO-SSA-SVR, and CS-SSA-SVR models are illustrated in Figure 12, while Figure 13 shows the results of the proposed methods for Case 2. Figures 12(b) and 12(c) are boxplots of the percent error (PE) and error (E), respectively. In Figure 12(b), the first to fourth boxplots correspond to the models of SARIMA, SSA-SARIMA, PSO-SSA-SARIMA, and CS-SSA-SARIMA. The positive values of PE and E mean the forecasting values are underestimated, while the negative values indicate the forecasting values are overestimated. The rest of the graphs and subgraphs can be identified in the same manner.

It can be seen from Figures 12 to 13 over the predictive horizon that the forecasting series obtained by the proposed hybrid models CS-SSA-SARIMA and CS-SSA-SVR perform substantially better with synchronicity and a range of

the vibration of the series approximating to the original series. In Figures 12(a) and 12(d), it is obvious that both of the curves for SSA-SARIMA and SSA-SVR are substantially closer to the original data. A consistent conclusion can be seen in Figures 12(b), 12(c), 12(e), and 12(f). In Figures 12(b) and 12(c), the boxplots from left to right represent the accuracy of SARIMA, SSA-SARIMA, PSO-SSA-SARIMA, and CS-SSA-SARIMA. Through observing the trend of these four boxplots from Figures 12(b) and 12(c), we discover a trend evidently decreasing progressively. Similarly, a decreasing tendency can be observed from Figures 12(e) and 12(f). Furthermore, when observing each of boxplot from Figure 12(b), we notice that many forecasting values of SARMIA model are underestimated, while that of SSA-SARIMA are less underestimated because the median of SSA-SARIMA's boxplot is much smaller than that of SARIMA's boxplot. These results combined with the similar results obtained from Figure 13 demonstrate the excellent denoising performance of the SSA and that processed data by the SSA can enhance the forecasting estimate of the models built by SARIMA and SVR.

In Figures 12(a)–12(c), it can obviously be observed that the forecasting performance of SARIMA, SSA-SARIMA, and PSO-SSA-SARIMA are substantially lower than that obtained by the proposed model, especially during the periods of peak load, which is marked with black dotted rectangles. Moreover, the same phenomenon can be obtained from Figures 12(d)–12(f) with the poorest forecasting results obtained by a single SVR model. Additionally, from Figures 12(e) and 12(f) the median of CS-SSA-SVR's boxplot is close to zero, while median of other models are not. This indicates that the forecasting values of proposed model are only slightly underestimated. In sum, the forecasting values of model SSA-SVR are substantially better than that of the SVR but are inferior to that generated by the model CS-SSA-SVR. As for Case 2, the similar results can be concluded from Figure 13. These figures reveal that the models of SSA-SARIMA and SSA-SVR based on the optimization of the CS have a higher predictive precision.

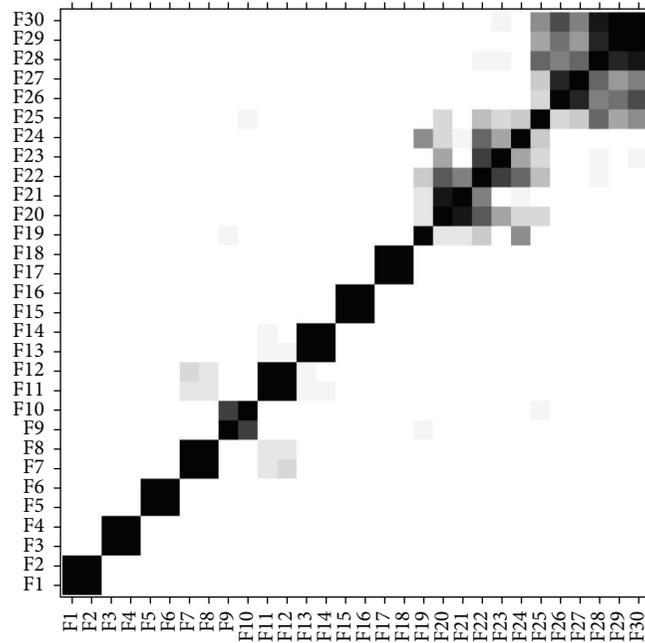


FIGURE 9: 2nd stage: w-correlation matrix ($L = 168$).

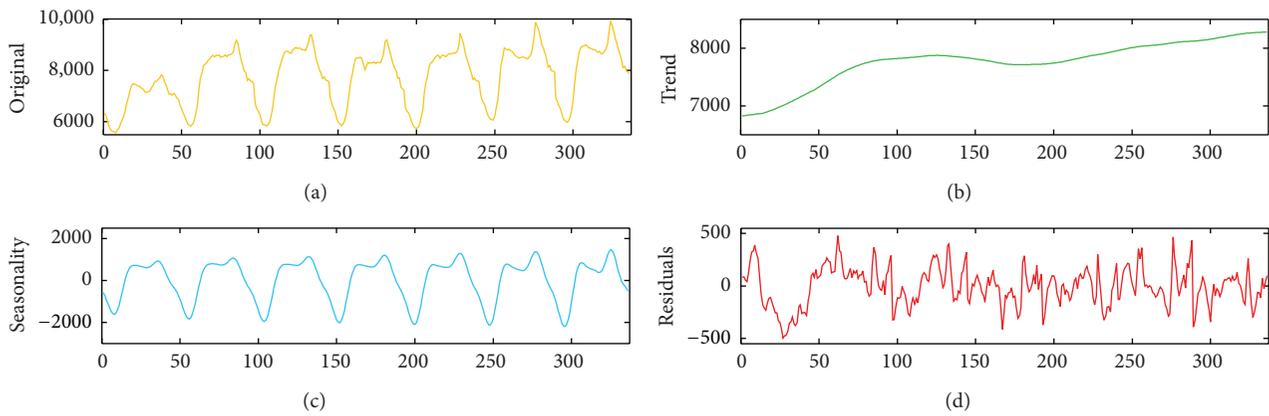


FIGURE 10: 2nd stages: original series and its trend-periodic-residuals decomposition.

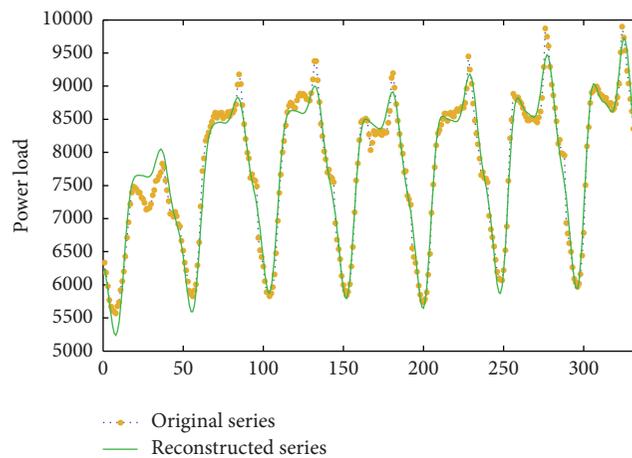


FIGURE 11: Reconstruction series and original series.

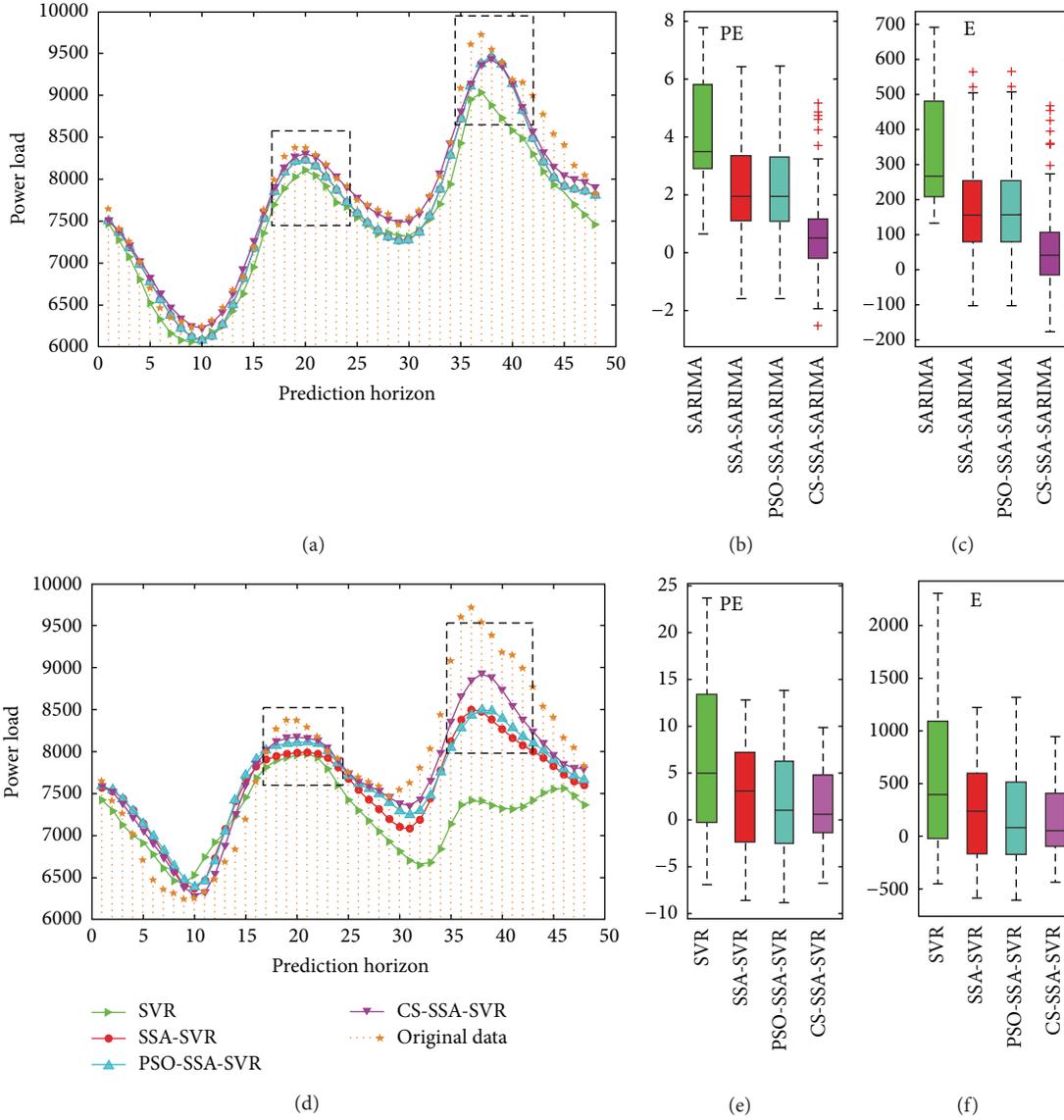


FIGURE 12: The forecasting results of the proposed PSO-SSA-SARIMA, CS-SSA-SARIMA, PSO-SSA-SVR, and CS-SSA-SVR models for Case 1 ((b, e): $PE(t) = (y(t) - \hat{y}(t))/y(t)$, (c, f): $E(t) = y(t) - \hat{y}(t)$).

Figure 14 illustrates the convergence speed by PSO and CS optimizing the SSA-SARIMA model in Case 1 and Case 2, respectively. It is evident that in both cases, both the convergence point and MSE of the PSO is smaller than that of the CS, while the precision of the prediction for the CS-SSA-SARIMA model is higher than that of the PSO illustrated from Figures 12 and 13 and listed in Table 1. This demonstrates that PSO has a faster convergence speed but a lower accuracy than the CS in these two cases, implying that the PSO appears to be overfitting.

To evaluate the forecasting model quantitatively, the statistical errors are computed in testing datasets over the forecasting horizon. Table 1 and Figure 15 give the statistical error measures' comparisons between different models. As shown in Table 1 and Figure 15, MAPE of SARIMA decreases from 4.22% to 2.37% in Case 1 and from 6.87% to 4.60% in Case

TABLE 1: Statistical error measures' comparison between different models.

Model	MAPE (%)		MAE	
	Case 1	Case 2	Case 1	Case 2
SARIMA	4.22	6.87	344.32	613.62
SSA-SARIMA	2.37	4.60	190.68	404.10
PSO-SSA-SARIMA	2.37	4.47	190.79	392.06
CS-SSA-SARIMA	1.38	3.82	113.72	333.12
SVR	8.75	6.64	734.83	580.10
SSA-SVR	5.53	5.76	451.26	516.21
PSO-SSA-SVR	4.96	4.68	404.26	392.42
CS-SSA-SVR	3.42	3.73	278.86	305.88

2. The result of the MAE is similar to the MAPE, decreasing

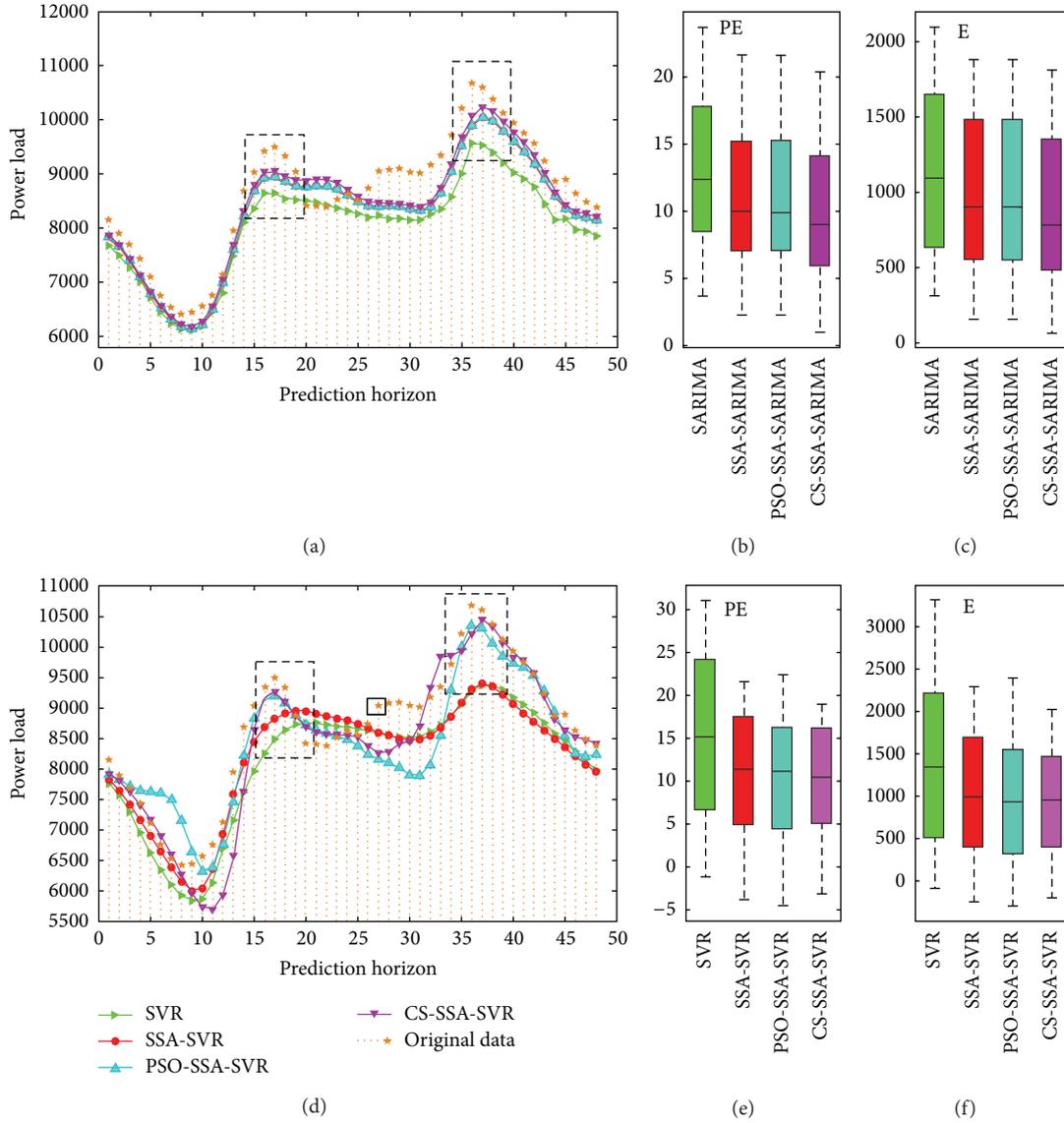


FIGURE 13: The forecasting results of the proposed PSO-SSA-SARIMA, CS-SSA-SARIMA, PSO-SSA-SVR, and CS-SSA-SVR models for Case 2 ((b, e): $PE(t) = (y(t) - \hat{y}(t))/y(t)$, (c, f): $E(t) = y(t) - \hat{y}(t)$).

from 344.32 to 190.68 in Case 1 and from 613.62 to 404.10 in Case 2. As for the SVR, the SSA-SVR outperforms the single SVR method because both the MAPE and MAE decrease in Case 1 and Case 2. This result demonstrates that the SSA has a strong capacity for noise elimination, which can improve the forecasting performance of SARIMA and SVR. Considering that the cuckoo search can be a very effective tool in parameter searching for further improving the accuracy of the SARIMA and SVR models, in this study, the CS-SSA-SARIMA and CS-SSA-SVR models outperform the SSA-SARIMA, PSO-SSA-SARIMA, SSA-SVR, and PSO-SSA-SVR models, respectively. These results prove that the cuckoo search is capable of improving the accuracy of model forecasting. Then, a comparison between the CS-SSA-SARIMA and PSO-SSA-SARIMA models and a comparison between the CS-SSA-SVR and PSO-SSA-SVR models indicate that the CS

algorithm outperforms the PSO algorithm in the application of improving the forecasting capacity of the proposed hybrid models.

4. Conclusions and Future Work

This paper presents hybrid swarm intelligent forecasting model strategies to accurately predict the short-term power load. The results obtained in this study illustrate that the SSA technique can be successfully used as a noise eliminating technique for time series similar to the short-term power load time series used here. The SSA-based denoising technique is capable of extracting important trend and seasonal components and then reconstructing it into smooth data to enhance the forecasting accuracy for SARIMA and SVR. In addition, the good noise eliminating ability via SSA could

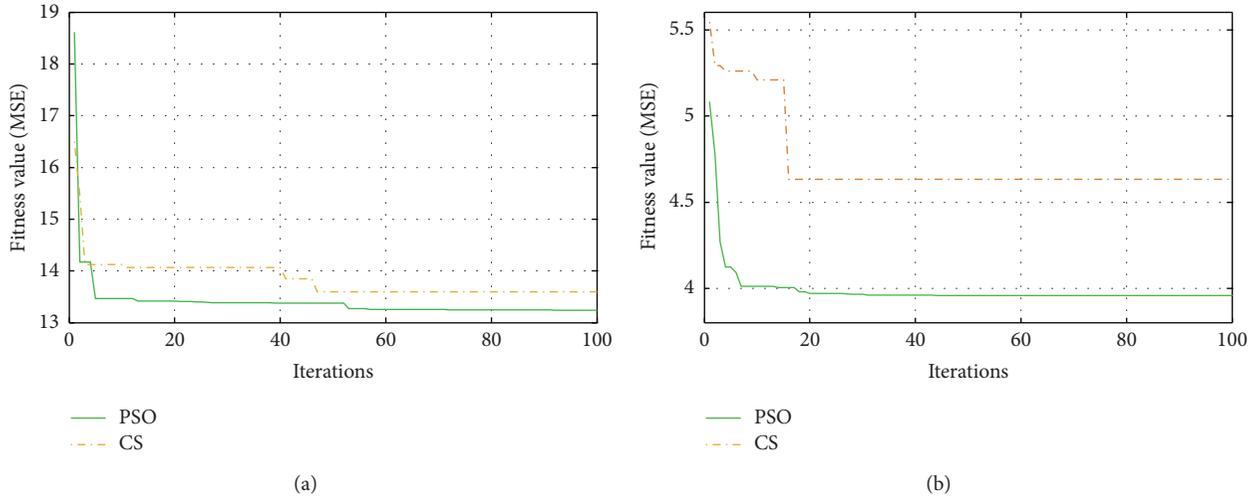


FIGURE 14: CS and PSO convergence procedure during the training for the NSW load.

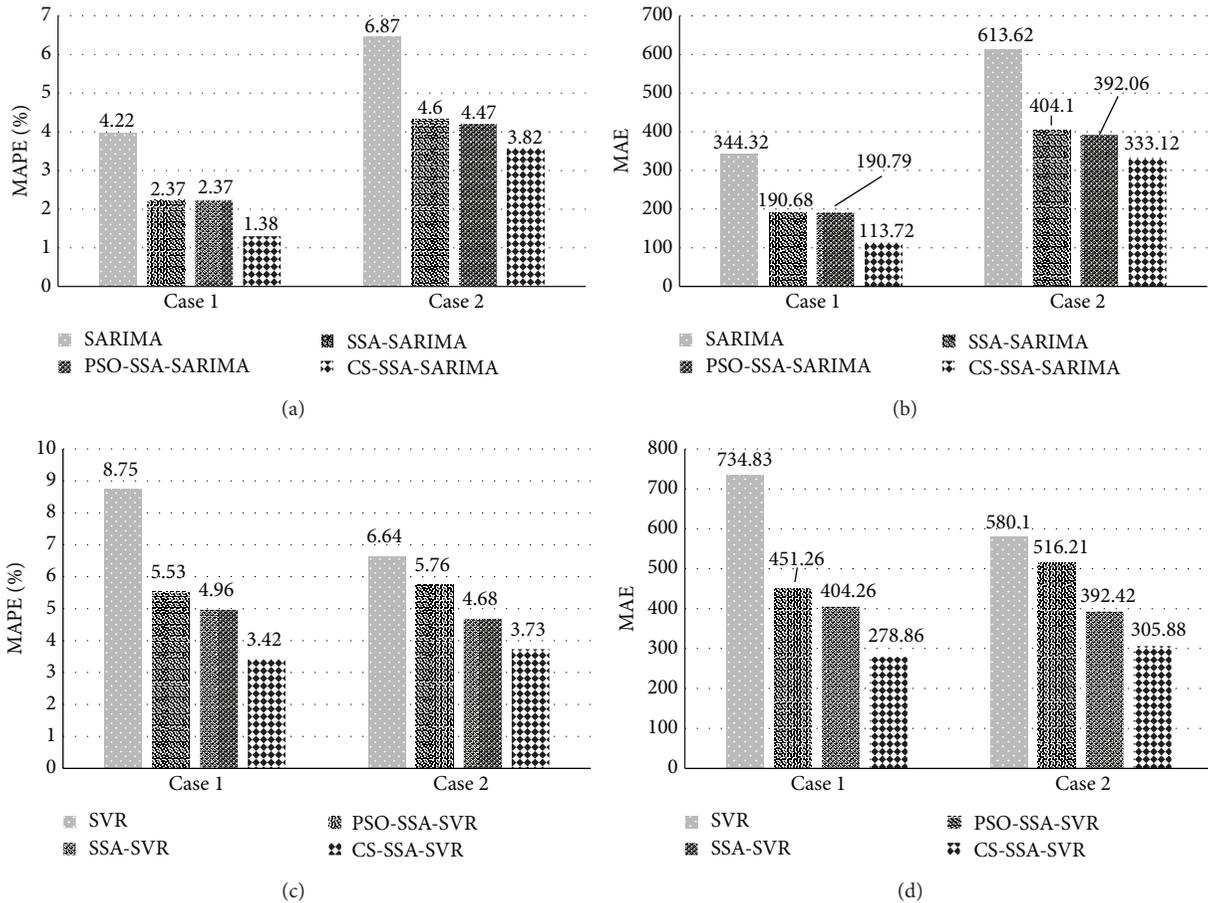


FIGURE 15: Statistical error measures' comparison between different models.

make these characteristics more obvious when modeling and could provide a more accurate forecast by SARIMA and SVR.

The CS algorithm is a recently developed metaheuristic artificial intelligence algorithm of parameter optimization. It has the ability to search parameters outstrips, that of

the maximum likelihood estimation method and that of the traditional optimization algorithm (PSO) when estimating the parameters of SARIMA. Similarly, its capability of expanding the scope of the search intelligently provides an optimization that is more effective and efficient than a grid

search when searching the optimal hyperparameters in SVR. Although PSO has a faster convergence velocity, it appears to have an overfitting problem. In our cases, it is revealed that by optimizing the parameters of the SSA-SARIMA and SSA-SVR models, the CS algorithm can further enhance the accuracy of prediction in short-term power loads and obtains a higher precision than PSO. The proposed hybrid swarm intelligent forecasting model could predict the short-term power load in a real-world scenario, which helps to enhance the predictive accuracy of the power system.

In this paper, our contribution is that an SI-based forecasting model is proposed to highly increase the accuracy. However, we did not sufficiently compare other feasible forecasting models, data preprocessing methods, and AI algorithms, such as BP, autoregressive integrated moving average (ARIMA), wavelet analysis, and GA. A more detailed comparison between the proposed method and other feasible forecasting models, data preprocessing methods, and SI optimizations is required. This is a very heavy workload but is very meaningful research; thus, it is necessary to perform additional research in future work.

For future work, we outline four directions. The first direction is to study the use of the other feasible forecasting models mentioned above within our framework. The second direction is to study in detail the other feasible SI optimization algorithms mentioned in this paper to search parameters of various forecasting models. The third direction is to study the use of other feasible data preprocessing methods, including the methods of not only denoising useless information but also removing outliers. The fourth direction is to explore the ability of different SI optimization algorithms to search certain parameters of certain forecasting models.

Highlights

- (i) A novel swarm intelligence-based hybrid approach is proposed for short-term load forecasting.
- (ii) The proposed approach consists of three steps to increase the forecasting accuracy.
- (iii) SSA is used for removing noised information in the first step.
- (iv) SARIMA and SVR are used for forecasting in the second step.
- (v) CS is employed to optimize the parameters of SARIMA and SVR.
- (vi) The proposed approach can improve the forecasting accuracy.

Conflict of Interests

The authors declare that they have no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant no. 7117110).

References

- [1] A. U. Haque, J. Meng, P. Mandal, and R. L. Pineda, "Performance evaluation of different optimization algorithms for power demand forecasting applications in a smart grid environment," *Procedia Computer Science*, vol. 12, pp. 320–325, 2012.
- [2] F. Javed, N. Arshad, F. Wallin, I. Vassileva, and E. Dahlquist, "Forecasting for demand response in smart grids: an analysis on use of anthropologic and structural data and short term multiple loads forecasting," *Applied Energy*, vol. 96, pp. 150–160, 2012.
- [3] D. Niu, Y. Wang, and D. D. Wu, "Power load forecasting using support vector machine and ant colony optimization," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2531–2539, 2010.
- [4] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 134–141, 2012.
- [5] M. López, S. Valero, C. Senabre, J. Aparicio, and A. Gabaldon, "Application of SOM neural networks to short-term load forecasting: the Spanish electricity market case study," *Electric Power Systems Research*, vol. 91, pp. 18–27, 2012.
- [6] R.-A. Hooshmand, H. Amooshahi, and M. Parastegari, "A hybrid intelligent algorithm based short-term load forecasting approach," *International Journal of Electrical Power and Energy Systems*, vol. 45, no. 1, pp. 313–324, 2013.
- [7] M. Moazzami, A. Khodabakhshian, and R. Hooshmand, "A new hybrid day-ahead peak load forecasting method for Iran's National Grid," *Applied Energy*, vol. 101, pp. 489–501, 2013.
- [8] V. Vapnik, S. Golowich, and A. Smola, "Support vector machine for function approximation, regression estimation, and signal processing," *Advances in Neural Information Processing Systems*, vol. 9, pp. 281–287, 1996.
- [9] W.-M. Hung and W.-C. Hong, "Application of SVR with improved ant colony optimization algorithms in exchange rate forecasting," *Control and Cybernetics*, vol. 38, no. 3, pp. 863–891, 2009.
- [10] P.-F. Pai, C.-S. Lin, W.-C. Hong, and C.-T. Chen, "A hybrid support vector machine regression for exchange rate prediction," *International Journal of Information and Management Sciences*, vol. 17, no. 2, pp. 19–32, 2006.
- [11] S. Ch, N. Anand, B. K. Panigrahi, and S. Mathur, "Streamflow forecasting by SVM with quantum behaved particle swarm optimization," *Neurocomputing*, vol. 101, pp. 18–23, 2013.
- [12] G.-F. Lin, Y.-C. Chou, and M.-C. Wu, "Typhoon flood forecasting using integrated two-stage support vector machine approach," *Journal of Hydrology*, vol. 486, pp. 334–342, 2013.
- [13] W.-C. Hong, Y. Dong, L.-Y. Chen, and S.-Y. Wei, "SVR with hybrid chaotic genetic algorithms for tourism demand forecasting," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1881–1890, 2011.
- [14] W.-C. Hong, "Hybrid evolutionary algorithms in a SVR-based electric load forecasting model," *International Journal of Electrical Power and Energy Systems*, vol. 31, no. 7-8, pp. 409–417, 2009.
- [15] W.-C. Hong, "Chaotic particle swarm optimization algorithm in a support vector regression electric load forecasting model," *Energy Conversion and Management*, vol. 50, no. 1, pp. 105–117, 2009.
- [16] W.-C. Hong, "Electric load forecasting by support vector model," *Applied Mathematical Modelling*, vol. 33, no. 5, pp. 2444–2454, 2009.

- [17] W.-C. Hong, "Application of chaotic ant swarm optimization in electric load forecasting," *Energy Policy*, vol. 38, no. 10, pp. 5830–5839, 2010.
- [18] S. Pino-Povedano and F.-J. González-Serrano, "Comparison of optimization algorithms in the sensor selection for predictive target tracking," *Ad Hoc Networks*, vol. 20, pp. 182–192, 2014.
- [19] H. Kaydani and A. Mohebbi, "A comparison study of using optimization algorithms and artificial neural networks for predicting permeability," *Journal of Petroleum Science and Engineering*, vol. 112, pp. 17–23, 2013.
- [20] J. Ahmed and Z. Salam, "A Maximum Power Point Tracking (MPPT) for PV system using Cuckoo Search with partial shading capability," *Applied Energy*, vol. 119, pp. 118–130, 2014.
- [21] X. S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *World Congress on Nature & Biologically Inspired Computing*, pp. 210–214, IEEE, 2009.
- [22] A. Gherboudj, A. Layeb, and S. Chikhi, "Solving 0-1 knapsack problems by a discrete binary version of cuckoo search algorithm," *International Journal of Bio-Inspired Computation*, vol. 4, no. 4, pp. 229–236, 2012.
- [23] P. R. Srivastava, A. Varshney, P. Nama, and X.-S. Yang, "Software test effort estimation: a model based on cuckoo search," *International Journal of Bio-Inspired Computation*, vol. 4, no. 5, pp. 278–285, 2012.
- [24] M. K. Marichelvam, "An improved hybrid Cuckoo Search (IHCS) metaheuristics algorithm for permutation flow shop scheduling problems," *International Journal of Bio-Inspired Computation*, vol. 4, no. 4, pp. 200–205, 2012.
- [25] P. R. Srivastava, C. Sravya, A. S. Kamiseti, and M. Lakshmi, "Test sequence optimisation: an intelligent approach via cuckoo search," *International Journal of Bio-Inspired Computation*, vol. 4, no. 3, pp. 139–148, 2012.
- [26] M. Basu and A. Chowdhury, "Cuckoo search algorithm for economic dispatch," *Energy*, vol. 60, pp. 99–108, 2013.
- [27] E. Cuevas, F. Sención, D. Zaldivar, M. Pérez-Cisneros, and H. Sossa, "A multi-threshold segmentation approach based on artificial bee colony optimization," *Applied Intelligence*, vol. 37, no. 3, pp. 321–336, 2012.
- [28] R. C. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceeding IEEE International Conference on Evolutionary Computation*, vol. 1, pp. 81–86, 2001.
- [29] K. Hammouche, M. Diaf, and P. Siarry, "A multilevel automatic thresholding method based on a genetic algorithm for a fast image segmentation," *Computer Vision and Image Understanding*, vol. 109, no. 2, pp. 163–175, 2008.
- [30] M.-H. Horng, "Multilevel minimum cross entropy threshold selection based on the honey bee mating optimization," *Expert Systems with Applications*, vol. 37, no. 6, pp. 4580–4592, 2010.
- [31] M.-H. Horng and T.-W. Jiang, "Multilevel image thresholding selection using the artificial bee colony algorithm," in *Artificial Intelligence and Computational Intelligence*, vol. 6320 of *Lecture Notes in Computer Science*, pp. 318–325, Springer, Berlin, Germany, 2010.
- [32] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, December 1995.
- [33] E. García-Gonzalo and J. L. Fernández-Martínez, "A Brief historical review of particle swarm optimization (PSO)," *Journal of Bioinformatics and Intelligent Control*, vol. 1, no. 1, pp. 3–16, 2012.
- [34] L. Ali and S. L. Sabat, "Particle swarm optimization based universal solver for global optimization," *Journal of Bioinformatics and Intelligent Control*, vol. 1, no. 1, pp. 95–105, 2012.
- [35] Z. H. Cui and X. J. Cai, "Integral particle swarm optimization with dispersed accelerator information," *Fundamenta Informaticae*, vol. 95, no. 4, pp. 427–447, 2009.
- [36] Z. H. Cui, X. J. Cai, J. C. Zeng, and Y. F. Yin, "PID-controlled particle swarm optimization," *Journal of Multiple-Valued Logic and Soft Computing*, vol. 16, no. 6, pp. 585–609, 2010.
- [37] P. D. Sathya and R. Kayalvizhi, "Optimum multi level image thresholding based on Tsallis entropy method with bacterial foraging algorithm," *International Journal of Computer Science*, vol. 7, no. 5, pp. 336–343, 2010.
- [38] W. Tao, H. Jin, and L. Liu, "Object segmentation using ant colony optimization algorithm and fuzzy entropy," *Pattern Recognition Letters*, vol. 28, no. 7, pp. 788–796, 2007.
- [39] J. Wang, J. Wang, Y. Li, S. Zhu, and J. Zhao, "Techniques of applying wavelet de-noising into a combined model for short-term load forecasting," *International Journal of Electrical Power & Energy Systems*, vol. 62, pp. 816–824, 2014.
- [40] G. P. Box and G. Jenkins, *Time Series Analysis, Forecasting and Control*, Holden-Day, San Francisco, Calif, USA, 1970.
- [41] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*, Holden-Day, San Francisco, Calif, USA, 1976.
- [42] J. B. Elsner and A. A. Tsonis, *Singular Spectrum Analysis. A New Tool in Time Series Analysis*, Plenum Press, New York, NY, USA, 1996.
- [43] N. Golyandina, V. Nekrutkin, and A. Zhigljavsky, *Analysis of Time Series Structure—SSA and Related Techniques*, vol. 90, Chapman & Hall/CRC, Boca Raton, Fla, USA, 2001.
- [44] M. R. S. Claudio, "Singular spectrum analysis and forecasting of failure time series," *Reliability Engineering & System Safety*, vol. 114, no. 1, pp. 126–136, 2013.
- [45] R. Vautard and M. Ghil, "Singular spectrum analysis in non-linear dynamics, with applications to paleoclimatic time series," *Physica D: Nonlinear Phenomena*, vol. 35, no. 3, pp. 395–424, 1989.
- [46] R. Vautard, P. Yiou, and M. Ghil, "Singular-spectrum analysis: a toolkit for short, noisy chaotic signals," *Physica D: Nonlinear Phenomena*, vol. 58, no. 1–4, pp. 95–126, 1992.
- [47] M. de Carvalho, P. C. Rodrigues, and A. Rua, "Tracking the US business cycle with a singular spectrum analysis," *Economics Letters*, vol. 114, no. 1, pp. 32–35, 2012.
- [48] H. Hassani, S. Heravi, and A. Zhigljavsky, "Forecasting European industrial production with singular spectrum analysis," *International Journal of Forecasting*, vol. 25, no. 1, pp. 103–118, 2009.
- [49] N. Golyandina and A. Korobeynikov, "Basic singular spectrum analysis and forecasting with R," *Computational Statistics & Data Analysis*, vol. 71, pp. 934–954, 2014.
- [50] Q. Chen, T. van Dam, N. Sneeuw, X. Collilieux, M. Weigelt, and P. Rebeschung, "Singular spectrum analysis for modeling seasonal signals from GPS time series," *Journal of Geodynamics*, vol. 72, pp. 25–35, 2013.
- [51] C. L. Wu and K. W. Chau, "Rainfall-runoff modeling using artificial neural network coupled with singular spectrum analysis," *Journal of Hydrology*, vol. 399, no. 3–4, pp. 394–409, 2011.
- [52] H. Hassani, "Singular spectrum analysis: methodology and comparison," *Journal of Data Science*, vol. 5, pp. 239–257, 2007.
- [53] K.-Y. Chen and C.-H. Wang, "A hybrid SARIMA and support vector machines in forecasting the production values of the

- machinery industry in Taiwan,” *Expert Systems with Applications*, vol. 32, no. 1, pp. 254–264, 2007.
- [54] M. Bouzerdoum, A. Mellit, and A. M. Pavan, “A hybrid model (SARIMA-SVM) for short-term power forecasting of a small-scale grid-connected photovoltaic plant,” *Solar Energy*, vol. 98, pp. 226–235, 2013.
- [55] V. Vapnik, S. Golowich, and A. J. Smola, “Support vector method for function approximation, regression estimation and signal processing,” *Advanced Neural Information Processing System*, vol. 9, pp. 281–287, 1996.
- [56] W.-C. Hong, Y. Dong, W. Y. Zhang, L.-Y. Chen, and B. K. Panigrahi, “Cyclic electric load forecasting by seasonal SVR with chaotic genetic algorithm,” *International Journal of Electrical Power and Energy Systems*, vol. 44, no. 1, pp. 604–614, 2013.
- [57] S. Amari and S. Wu, “Improving support vector machine classifiers by modifying kernel functions,” *Neural Networks*, vol. 12, no. 6, pp. 783–789, 1999.
- [58] S. Pino-Povedano and F. J. González-Serrano, “Comparison of optimization algorithms in the sensor selection for predictive target tracking,” *Ad Hoc Networks*, vol. 20, pp. 182–192, 2014.
- [59] W.-C. Hong, Y. Dong, W. Y. Zhang, L.-Y. Chen, and B. K. Panigrahi, “Cyclic electric load forecasting by seasonal SVR with chaotic genetic algorithm,” *Electrical Power and Energy Systems*, vol. 44, no. 1, pp. 604–614, 2013.

Research Article

A Modified Artificial Bee Colony Algorithm Based on Search Space Division and Disruptive Selection Strategy

Zhen-an He,^{1,2,3} Caiwen Ma,¹ Xianhong Wang,¹ Lei Li,⁴ Ying Wang,⁴
Yuan Zhao,^{1,2,5} and Huinan Guo¹

¹ Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, Shaanxi 710019, China

² University of Chinese Academy of Sciences, Beijing 100049, China

³ Unit 96165 of the Chinese People's Liberation Army, Shangrao, Jiangxi 334000, China

⁴ North Optoelectronic Stock Ltd., Xi'an, Shaanxi 710032, China

⁵ Engineering University of the Chinese Armed Police Force, Xi'an, Shaanxi 710086, China

Correspondence should be addressed to Zhen-an He; std_2009@163.com

Received 27 May 2014; Revised 28 August 2014; Accepted 11 September 2014; Published 28 September 2014

Academic Editor: Fang Zong

Copyright © 2014 Zhen-an He et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Artificial bee colony (ABC) algorithm has attracted much attention and has been applied to many scientific and engineering applications in recent years. However, there are still some insufficiencies in ABC algorithm such as poor quality of initial solution, slow convergence, premature, and low precision, which hamper the further development and application of ABC. In order to further improve the performance of ABC, we first proposed a novel initialization method called search space division (SSD), which provided high quality of initial solutions. And then, a disruptive selection strategy was used to improve population diversity. Moreover, in order to accelerate convergence rate, we changed the definition of the scout bee phase. In addition, we designed two types of experiments to testify our proposed algorithm. On the one hand, we conducted experiments to make sure how much each modification makes contribution to improving the performance of ABC. On the other hand, comprehensive experiments were performed to prove the superiority of our proposed algorithm. The experimental results indicate that SDABC significantly outperforms other ABCs, contributing to higher solution accuracy, faster convergence speed, and stronger algorithm stability.

1. Introduction

Optimization problems exist extensively in information science, control engineering, industrial design, operational research, and so on. Therefore, optimization algorithms, which are of great importance to engineering and scientific research, have attracted more and more of the researchers' attention and many achievements have been got. However, conventional optimization algorithms fail to process problems characterized as nonconvex, nondifferentiable, discontinuous, high dimensional, and so forth; swarm intelligence algorithms that have some advantages, such as scalability, fault tolerance, adaptation, speed, modularity, autonomy, and parallelism [1], provide a promising way to solve the complex optimization problems mentioned above. Some excellent algorithms, such as genetic algorithm (GA) [2], particle swarm optimization (PSO) [3], differential evolutionary (DE)

algorithm [4], ant colony optimization (ACO) [5], simulate anneal arithmetic (SAA) [6], artificial bee colony (ABC) algorithm [7], have been proposed and successful applications have been made in many fields. Among them, by simulating foraging behavior of honey bee swarm, ABC developed by Karaboga is competitive to other swarm intelligence algorithms [8]. As a result, it has been widely used in multiobjective optimization, circuit design, SAR image segmentation, flow shop scheduling, and related optimization problems [9–11]. Nevertheless, as research on ABC is at the primary phase, many problems such as poor quality of initial solutions, slow convergence, premature, and low precision are still hampering the development and application of ABC.

To overcome the above mentioned shortages, researchers have presented many kinds of ABC variants. Zhu and Kwong [12] proposed a gbest-guided ABC (GABC) algorithm by incorporating the information of global best (gbest) solution

into the solution search equation to improve the exploitation. Kang et al. [13] proposed a memetic algorithm which combined Hooke-Jeeves pattern search with artificial bee colony algorithm. Alatas [14] introduced the chaotic maps into the initialization and chaotic search into the scout bee phase to improve the convergence characteristics and to prevent the ABC to get stuck on local solutions. Gao and Liu [15] proposed a modified ABC by using a modified solution search equation and new initial method. Akay and Karaboga [1] proposed a modified ABC algorithm by controlling the frequency of perturbation and introducing the ratio of the variance operator into search equation. Mohammed [16] proposed a generalized opposition-based ABC by introducing the concept of generalized opposition-based learning through the initialization step and generation jumping. Luo et al. [17] used segmental-search strategy to make sure of fast convergence and to avoid trapping into local optimum. More extensive review of ABC can be seen in [18, 19].

According to the literatures mentioned above, the ABC variants have achieved a better performance, but some shortcomings of slow convergence and premature and low precision have not been solved yet. We note that to improve the performance of ABC, researchers mainly focus on search strategy research, but other important factors affecting performance, such as initialization method, selective strategy, and fitness value, are insufficient.

In order to further improve the performance of ABC, we proposed a modified artificial bee colony algorithm based on search space division and disruptive selection strategy (SDABC). In SDABC, first of all, a new initialization method called search space division, which steadily provided high quality of initial solutions, was presented. Note that convergence speed, final solution accuracy, and stability were improved by using search space division. It is well known that greedy strategy used in ABC is useful because it is quick to think up and often gives good approximations to the optimum. However, greedy strategy mostly (but not always) fails to find the globally optimal solution, because it usually does not operate exhaustively on all the data. It can make commitments to certain choices too early which prevent it from finding the best overall solution later. And thus, in this paper, we introduced the disruptive selection strategy to increase the diversity of population. Moreover, in order to accelerate convergence speed, a new search equation was presented in the scout bee phase. In addition, our proposed algorithm was evaluated on fourteen standard benchmark functions which had ever been applied to verify optimization methods in continuous optimization problems. The simulation results show that SDABC significantly outperforms other ABCs, contributing to higher solution accuracy, faster convergence speed, and stronger algorithm stability.

The rest of this paper is structured as follows. In Section 2, we make a brief review of standard ABC. Section 3 provides detailed descriptions of our proposed algorithm. Subsequently, some experiments are conducted to verify the performance of SDABC in Section 4. This is followed by conclusion and future research directions in Section 5.

2. Standard Artificial Bee Colony Algorithm

The ABC algorithm which was originally introduced by Karaboga is a recently proposed optimization algorithm inspired by simulating the foraging behavior of a honey bee swarm [1]. In the minimal model, a colony of artificial bees consists of three kinds of bees [7, 8]: employed bees, onlooker bees, and scout bees. Both of the number of employed bees and onlooker bees are equal to half of the bee colony. There is a one-to-one correspondence between each employed bee and every food source. Employed bees are responsible for searching food sources, gathering information, and sharing food information with onlooker bees around the hive. The employed bee whose food source has been exhausted by the bees becomes a scout. Scout bee, just as its name implies, could have the fast discovery of the group of feasible solutions as a task. In the ABC 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. The ABC algorithm can be concluded into four phases: initialization phase, employed bee phase, onlooker bee phase, and scout bee phase, as described below.

(1) *Initialization Phase.* To begin with, a population of SN individuals is generated randomly. Each initial food source X_i ($i = 1, 2, \dots, FN$) as a D -dimensional vector is produced by (1), where FN denotes food source, namely, solution

$$x_{ij} = lb_j + \text{rand}(0, 1)(ub_j - lb_j), \quad (1)$$

where $i = 1, 2, \dots, FN$, $j = 1, 2, \dots, D$, and D is the number of optimization parameters; ub_j and lb_j are the upper and lower bounds for the dimension j , respectively. And then, each solution is evaluated by

$$\text{fitness}_i = \begin{cases} \frac{1}{(1 + f_i)}, & f_i \geq 0 \\ 1 + |f_i|, & f_i < 0, \end{cases} \quad (2)$$

where fitness_i denotes fitness value of solution i and f_i represents cost function of solution i .

After the initialization, solutions repeat processes of employed bee phase, onlooker bee phase, and scout bee phase until stop conditions are satisfied.

(2) *Employed Bee Phase.* At this phase, each employed bee produces a new food source (solution) in the neighborhood of the previous selected solution. The position of the new food source is calculated by

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}), \quad (3)$$

where $j \in \{1, 2, \dots, D\}$ and $k \in \{1, 2, \dots, FN\}$ are randomly chosen indexes and k has to be different from i and v_{ij} is a new solution and ϕ_{ij} is a random number between $[-1, 1]$.

After that, fitness value of the new food source is evaluated, and compares with that of the old food source X_i . The old food source will be replaced by the new one if the new

Begin

- Step 1. population initialization based on (1).
 Step 2. Calculate fitness value by (2).
 Step 3. $cycle = 1$
 Step 4. Repeat
 Step 5. Update the position of the employed bee by (3).
 Step 6. A greedy selection strategy is applied to select the better food source.
 Step 7. Calculate probability value by (4).
 Step 8. Update the position of the onlooker bee by (3).
 Step 9. A greedy selection strategy is applied to select the better food source.
 Step 10. Determine the abandoned position, and generate a new position by (1).
 Step 11. Memorize the best food source achieved till now. $cycle = cycle + 1$
 Step 12. Until ($cycle = \text{maximum cycle number}$)

End.

ALGORITHM 1: Standard artificial bee colony algorithm.

one has a better fitness value. Otherwise, the old one will be stored.

(3) *Onlooker Bee Phase*. When all employed bees have finished their searching process, they share positions and fitness information of the food sources with onlooker bees, each of which selects a food source depending on probability p_i calculated by (4). After an onlooker bee chooses a food source with a probability, a new food source is generated and fitness value is calculated. Subsequently, a greedy selection is applied between the selected food source and the old one:

$$p_i = \frac{\text{fitness}_i}{\left(\sum_{j=1}^{FN} \text{fitness}_j\right)}. \quad (4)$$

(4) *Scout Bee Phase*. If the fitness value of a food source does not improve for a certain number of cycles, the food source will be abandoned and the corresponding employed bee becomes a scout bee. Then, the scout bee produces a new position randomly as in (1) to replace the position of abandoned food source.

On the basis of the above analysis, the pseudocode of standard artificial bee colony algorithm is simply described in Algorithm 1.

3. Our Proposed Algorithm: SDABC

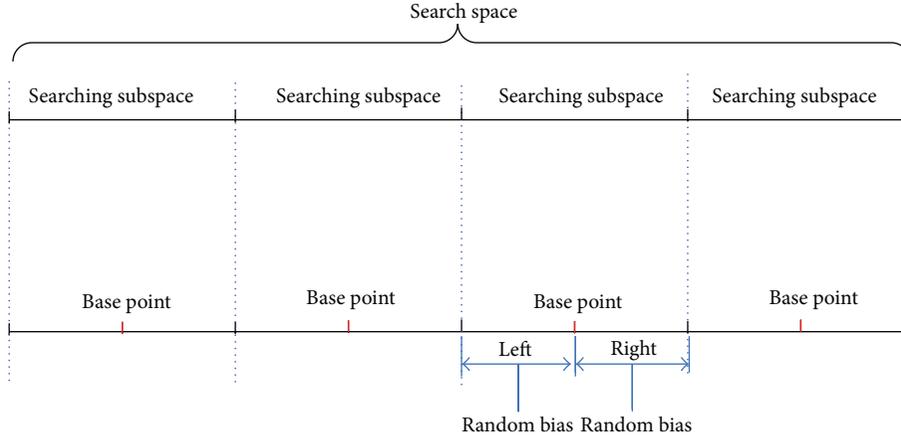
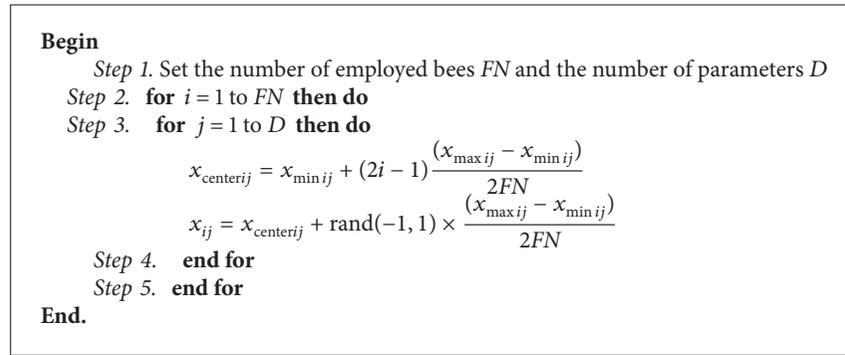
Attracted by the prospect and potential of the ABC algorithm, researchers have focused on its variants. However, there are still some problems which need to be solved. In this section, to further improve its performance, we introduce the modified artificial bee colony from three aspects, namely, population initialization based on search space division, disruptive selection strategy, and the improved scout bee phase. The detailed descriptions are as follows.

3.1. Population Initialization Based on Search Space Division. An initialization method capable of providing a better exploration of the search-space and presenting only high-quality

solutions should improve the performance of a metaheuristic algorithm [20]. Specifically, population initialization influences convergence speed, final solution accuracy, and stability. Generally speaking, in absence of a prior knowledge about the solution, the random initialization is the most commonly used way to produce initial population. However, because of its randomness, it is impossible for random initialization to get high quality of initial solution steadily. That is to say, once in a while, an initial solution produced by random initialization close to optimal solution could result in a fast convergence. Otherwise, it will take considerably more time. Logically, we should be looking in all directions simultaneously [21], in other words, initial solutions should be distributed evenly across the whole search space.

Accordingly, we proposed a novel initialization algorithm called search space division (SSD). In order to make it easy to understand, we take one-dimensional space as an example, as shown in Figure 1. The basic ideal of SSD is given as follows: to begin with, SSD divides the search space into FN segments (e.g., $FN = 4$). There is one and only one initial solution in every segment (searching subspace), which makes sure initial solutions should be distributed relatively evenly across the whole search space. After that, SSD uses each midpoint in searching subspace as a base point, and initial solutions are random bias from base points which can move from left or right for up to half the length of segment. Here, random numbers generated in a limited scope are used to produce different initial solutions at each run, which make a chance for algorithm to succeed in the next run while it failed in the previous run. It is necessary to emphasize that SSD can easily work on the high dimensions. Based on the research described above, the main steps of SSD can be summarized as in Figure 1.

Let $x_{ij} \in [x_{\min ij}, x_{\max ij}]$ be a real number, which is the j th dimension of i th solution. Where $i \in [1, 2, \dots, FN]$ (FN denotes the number of initial solutions) and $j \in [1, 2, \dots, D]$ (D represents the number of parameters), $x_{\max ij}$ and $x_{\min ij}$ are the upper and lower bounds for the j th dimension of i th

FIGURE 1: Initial solutions produced by SSD for $FN = 4$.

ALGORITHM 2: Population initialization based on search space division.

solution, respectively. Then, base points are generated by (5), and initial solutions are produced by using (6)

$$x_{centerij} = x_{minij} + (2i - 1) \frac{(x_{maxij} - x_{minij})}{2FN}, \quad (5)$$

where $x_{centerij}$ is the base point corresponding to the j th dimension of i th solution

$$x_{ij} = x_{centerij} + \text{rand}(-1, 1) \times \frac{(x_{maxij} - x_{minij})}{2FN}. \quad (6)$$

Through the above analysis, we use SSD instead of pure random initialization to produce initial solutions, and its pseudocode is given in Algorithm 2.

3.2. Disruptive Selection Strategy. Fitness value is of crucial importance to the ABC algorithm because it is the sole criterion of nectar amount. In the ABC algorithm, fitness value is generated by (2). As described in [22], there is a vital problem which needs to be solved. For example, when finding the minimum value, (2) is used to evaluate function value. However, when the function value is very close to zero, for example, $1E - 20$, fitness value calculated by (2) is rounded up to be 1 ($1E - 20$ is ignored). Subsequently, the fitness of all solutions will be equal to 1 in later iterations. That is to

say, a new solution that gives a better fitness value than the old solution will be ignored and the solution will stagnate at the old solution [22]. In order to solve this problem, literature [22] directly used the objective value of function for comparison and selection of the better solution. To some extent, the issue has been solved by the above amendment, but it has raised a new problem: onlooker bees lose search direction, which results in falling into local optimum. As the number of the iterations increases, difference of the objective function value becomes smaller and smaller, which requires fitness value to have a sensitive response to the slight change. Otherwise, onlooker bees cannot select appropriate food sources. Through the above analysis, we proposed a new equation to calculate the fitness value, which is given as follows:

$$\text{fitness}_i = \begin{cases} \frac{1}{f_i}, & f_i > 0 \\ |f_i|, & f_i \leq 0, \end{cases} \quad (7)$$

where f_i is the objective value of function and fitness_i is the fitness value.

After that, greedy selection strategy is applied to select food source. However, as is known to all, greedy selection strategy often falls into locally optimal solution, which results from lack of population diversity. In order to solve this issue,

TABLE 1: Benchmark functions.

Name	Function	Search space	Minimum
Sphere	$f_1(X) = \sum_{i=1}^D x_i^2$	$[-100, 100]^D$	0
SumSquare	$f_2(X) = \sum_{i=1}^D i x_i^2$	$[-10, 10]^D$	0
Rosenbrock	$f_3(X) = \sum_{i=1}^{D-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-2.048, 2.048]^D$	0
Schwefel2.21	$f_4(X) = \max_{i=1}^D \{ x_i , 1 \leq i \leq D\}$	$[-100, 100]^D$	0
Schwefel2.22	$f_5(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_i $	$[-10, 10]^D$	0
Step	$f_6(X) = \sum_{i=1}^D x_i + 0.5 ^2$	$[-100, 100]^D$	0
SumPower	$f_7(X) = \sum_{i=1}^D x_i ^{(i+1)}$	$[-1, 1]^D$	0
Exponential	$f_8(X) = \left \exp\left(0.5 * \sum_{i=1}^D x_i\right) - 1 \right $	$[-1.28, 1.28]^D$	0
Rotated	$f_9(X) = \sum_{i=1}^D \sum_{j=1}^i x_j^2$	$[-65.536, 65.536]^D$	0
Griewank	$f_{10}(X) = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]^D$	0
Schwefel	$f_{11}(X) = 418.982887 * D - \sum_{i=1}^D \left(x_i \sin\left(\sqrt{ x_i }\right)\right)$	$[-500, 500]^D$	0
Ackley	$f_{12}(X) = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^D x_i^2}\right) - \exp\left(\frac{1}{D} \sum_{i=1}^D \cos(2\pi x_i)\right) + 20 + e$	$[-32.768, 32.768]^D$	0
Rastrigin	$f_{13}(X) = \sum_{i=1}^D (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$[-5.12, 5.12]^D$	0
Weierstrass	$f_{14}(X) = \sum_{i=1}^D \left(\sum_{k=0}^{20} [a^k \cos(2\pi b^k (x_i + 0.5))] \right) - D \sum_{k=0}^{20} [a^k \cos(2\pi b^k 0.5)]$ $a = 0.5, b = 3$	$[-0.5, 0.5]^D$	0

we introduced the disruptive selection strategy to maintain population diversity. The disruptive selection strategy gives more chances for higher and lower individuals to be selected by changing the definition of the fitness function as in (8) and (9) [23]

$$\text{fit}_i = \left| \text{fitness}_i - \overline{\text{fitness}_a} \right|, \quad (8)$$

$$P_i = \frac{\text{fit}_i}{\sum_{n=1}^{FN} \text{fit}_n}, \quad (9)$$

where $\overline{\text{fitness}_a}$ is the average value of the fitness value fitness_i of the individuals in the population. Based on the above amendments, the main steps of disruptive selection strategy are shown in Algorithm 3.

3.3. Improved Scout Bee Phase. When an employed bee becomes a scout bee, (1) is used to randomly generate a new position. Because of its randomness, the new position produced by (1) is more likely to be far away from global optimum or locates in the search area which has been explored.

```

Begin
  % Change the definition of the fitness function
  If  $f_i > 0$  then do
     $\text{fitness}_i = \frac{1}{f_i}$ ;
  Else do
     $\text{fitness}_i = |f_i|$ ;
  End if
   $\text{fit}_i = \left| \text{fitness}_i - \overline{\text{fitness}_a} \right|$ ;
  % Calculate probability
   $P_i = \frac{\text{fit}_i}{\sum_{n=1}^{FN} \text{fit}_n}$ ;
End.

```

ALGORITHM 3: Disruptive selection strategy.

The guidance of the best solution will rapidly accelerate convergence speed, which has been proved in literature [12]. It means that the scout bee has yet to take advantage of

```

Begin
                                {--Population initialization--}
Step 1. Set population size  $CS$ , the number of food sources  $FN$ , the non-improvement number of food source  $limit$ ,
the number of maximum iterations  $max\ Cycle$ , and the number of parameters  $D$ .
Step 2. Generate initial solutions using population initialization based on SSD as in (5) and (6), and calculate their fitness
values by (7) and (8).
Step 3.  $cycle = 1$ 
Step 4. While  $cycle \leq max\ Cycle$  do
                                {--Employed bee phase--}
    For  $i = 1 : FN$  do
        Update the position of the employed bee based on (3).
        Check whether it is out of boundaries or not.
        Calculate fitness value by (7) and (8).
        A greedy selection strategy is applied to select the better food source.
        A food source doesn't improve  $trial(i) = trial(i) + 1$ , otherwise  $trial(i) = 0$  where  $trial(i)$  represents
        the number of the  $i$ th trial.
    End for
                                {--Onlooker bee phase--}
Step 5. Calculate  $p_i$  based on (9), and initialize  $t = 0$ ,  $trial = 0$ .
    While  $t < FN$  then do
        If  $random < p_i$  then do
            Update the position of the onlooker bee by (3).
            Check whether it is out of boundaries or not.
            Calculate fitness value by (7) and (8).
            A greedy selection strategy is applied to select the better food source.
            A food source doesn't improve  $trial(i) = trial(i) + 1$ , otherwise  $trial(i) = 0$ .
        End if
    End while ( $t = FN$ )
                                {--Scout bee phase--}
Step 6. if  $max(trial) > limit$  then do
    Generate a new food source for the scout bee by (10).
    End if
Step 7. Memorize optimal solution achieved till now.  $cycle = cycle + 1$ ;
Step 8. End while ( $cycle = max\ Cycle$ )
End.

```

ALGORITHM 4: SDABC.

the information from the optimal solution achieved so far. Moreover, it is a hard work to determine $limit$ value because we are facing a big dilemma: setting the value of $limit$ that is too large results in poor population diversity, and the value of $limit$ that is too small renders slow convergence. As a matter of fact, the sole purpose of a scout bee is to maintain the population diversity, which can be replaced by other strategies. As a result, in this paper, we adopted the disruptive selection strategy to keep population diversity. In addition, for the purpose of accelerating convergence speed, we assigned a new task to the scout bee, and that is to further exploit the promising position. Therefore, a new equation generating the new position for scout bee is given as follows:

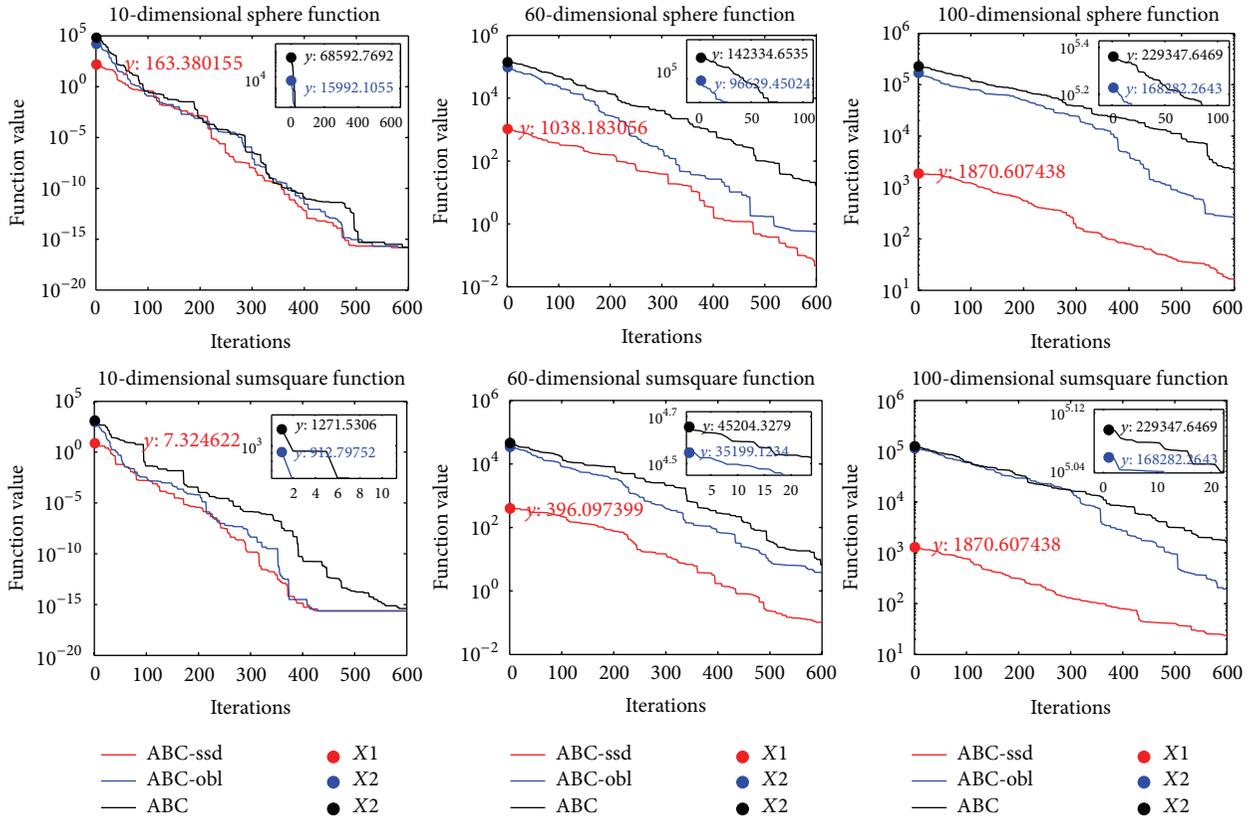
$$x_{ij} = x_{best,j} + \text{rand}(-1, 1)(x_{best,j} - x_{abandon,j}), \quad (10)$$

where $j \in [1, 2, \dots, D]$, x_{best} is the best solution achieved till now, and $x_{abandon}$ is the solution abandoned by employed bee. Through the above operations, the main steps of SDABC can be simply presented in Algorithm 4.

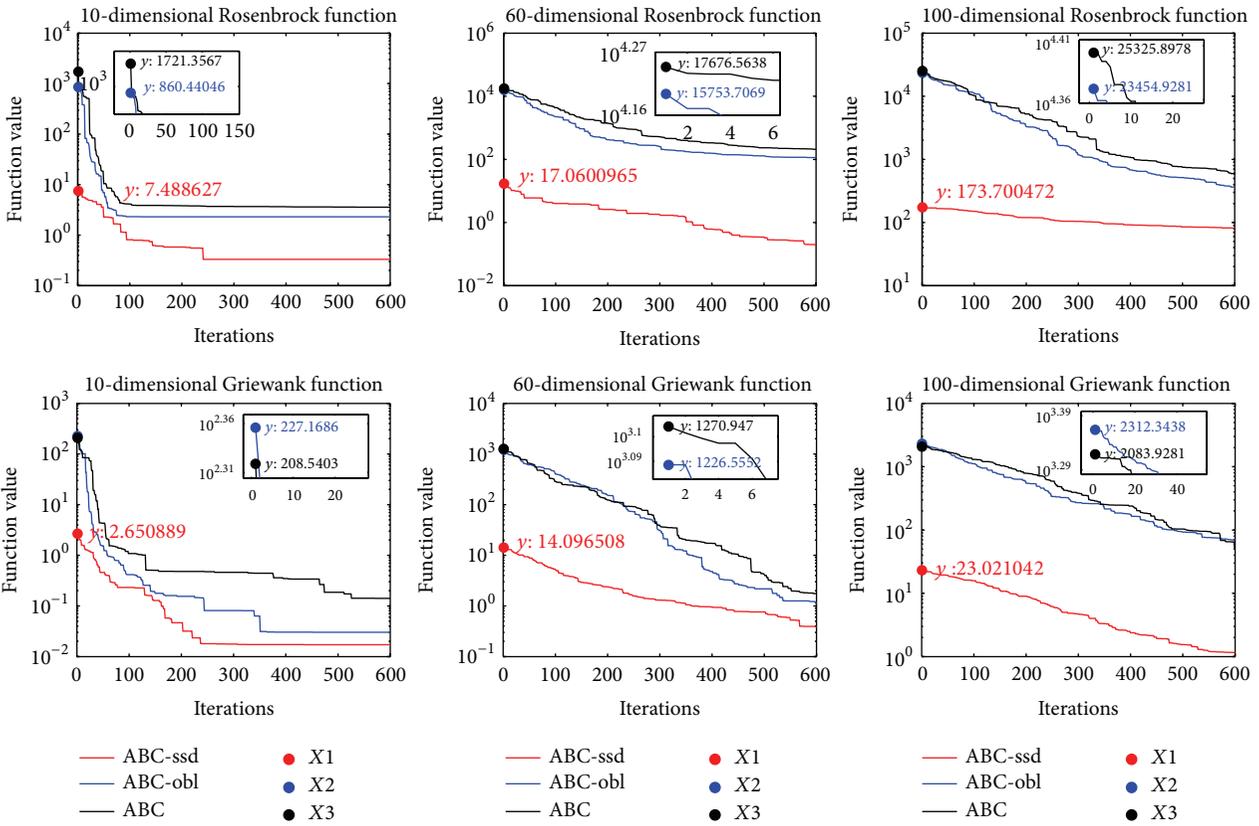
4. Simulation Experiments

In this section, a series of experiments were conducted to certify the effectiveness of our proposed algorithm SDABC in standard benchmark functions. Well-defined benchmark functions which are based on mathematical functions benefit the testing of our proposed algorithm in this paper. As a result, we selected fourteen benchmark functions to fulfill such experiments. The formulations and properties of these benchmark functions were summarized in Table 1. Specifically, $f_1 \sim f_9$ are unimodal functions and $f_{10} \sim f_{14}$ are multimodal functions. Hereinto, f_6 is discontinuous step function, f_{11} is bound-constrained function, and f_3 is a nonconvex function with multiple minima in high dimension case.

It is necessary to emphasize that all the experiments were implemented in the same hardware and software environment. Specifically, computer's hardware configuration is given as follows: an Intel Core 2 Duo processor running at 2.20 GHz, 512 M of RAM, and an 80 G hard driver. The operating system is Microsoft Windows sp3. Our execution was compiled by the default setting of Matlab R2010a.



(a)



(b)

FIGURE 2: Continued.

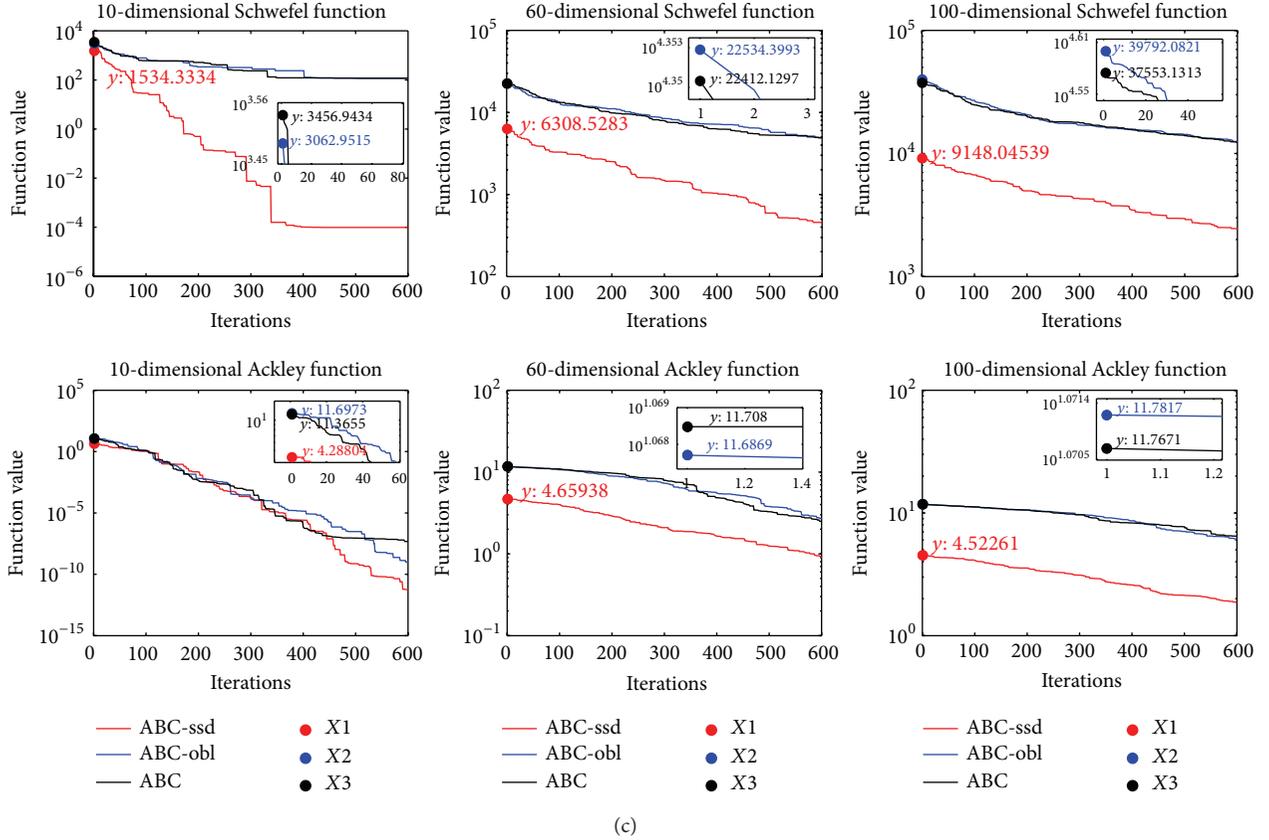


FIGURE 2: Convergence process of the best solution values over 100 runs.

4.1. Effects of Each Modification on the Performance of SDABC.

In this section, we conducted two types of experiments to analyze each modification, respectively. On the one hand, in order to further certify the superiority of our proposed initialization algorithm, we compared SSD with other initialization algorithms including opposition-based learning (OBL for short) proposed in [21] and random initialization (RI for short). For more clarity, we called the original ABC with our proposed population initialization algorithm (i.e., replacing RI in the original ABC with SSD) as ABC-ssd and the original ABC with OBL as ABC-obl. Note that ABC utilizes RI to generate initial population. In the experiments, the population size was set to be 20, the maximum number of cycles was set to be 600, and the value of *limit* was set to be 150. And then, we compared the convergence performance and final solution accuracy of different initialization methods on a set of six benchmark functions including three unimodal functions and three multimodal functions with $D = 10, 60,$ and $100,$ respectively. For all functions, each algorithm was performed 100 independent runs. The results were shown in Table 2 in terms of mean and standard deviation (Std for short) of solutions obtained by each algorithm. Moreover, the best and the second best solutions were marked in bold and a two-tailed t -test at a 0.05 level of significance was used. Note that the value “+” denotes our proposed algorithm significantly better than other algorithms. It can be observed that ABC-ssd has higher solution accuracy on all functions.

In addition, convergence process of the best solution values over 100 runs on six functions is presented in Figure 2. Hereinto, X1, X2, and X3 denote the best initial solutions generated by SSD, OBL, and RI, respectively. Moreover, some partial enlarged drawings are also provided in Figure 2 to see how much the different population initialization methods make contribution to improving quality of initial solutions. It can be seen clearly that SSD provides the highest quality of initial solution X1. For the six benchmark functions with $D = 30, 60$ and $100,$ SSD is significantly superior to OBL and RI during the whole progress of finding global minimum. There is a tendency that the more the number of optimization parameters is, the more obvious the superiority of SSD shows. Eventually, OBL performs the second best at finding global minimum. Specifically, for unimodal functions, OBL is somewhat superior to RI in terms of initial solution and convergence performance. However, for multimodal functions, the advantages may disappear.

On the other hand, because of correlative dependence, we combined disruptive selection strategy and the improved scout bee to investigate their effects on SDABC in terms of population diversity and convergence performance. To begin with, population diversity is defined as follows [24]:

$$D(X) = \frac{1}{D} \frac{\sum_{i=1}^D \|X_i - \bar{X}\|}{\max_{1 \leq i, j \leq D} \|X_i - X_j\|}, \quad (11)$$

TABLE 2: Mean and Std deviation of solutions obtained by the original ABC with different initialization algorithms.

Function	D	ABC		ABC-obl		ABC-ssd		Significant
		Mean	Std	Mean	Std	Mean	Std	
Sphere	10	$5.24e - 16$	$3.12e - 16$	$4.41e - 16$	$3.06e - 16$	$2.20e - 16$	$1.01e - 16$	+
	60	30.25	6.77	0.93	0.44	$8.09e - 2$	$7.31e - 2$	+
	100	$2.98 + 3$	$4.86e + 2$	$3.70e + 2$	$2.45e + 1$	$2.63e + 1$	3.44	+
SumSquare	10	$6.33e - 16$	$2.61e - 16$	$3.00e - 16$	$2.07e - 16$	$2.24e - 16$	$1.85e - 16$	+
	60	9.67	3.48	6.01	2.38	$3.04e - 1$	$2.76e - 1$	+
	100	$2.68e + 3$	$4.09e + 2$	$2.51e + 2$	$9.16e + 1$	3.09	1.15	+
Rosenbrock	10	6.06	3.85	3.92	2.78	$5.62e - 1$	$8.11e - 2$	+
	60	$3.18e + 2$	$1.09e + 2$	$1.72e + 2$	$8.45e + 1$	$2.85e - 1$	$7.71e - 2$	+
	100	$8.92e + 2$	$3.84e + 2$	$4.73e + 2$	$2.66e + 2$	$1.06e + 2$	$4.97e + 1$	+
Griewank	10	$2.81e - 1$	$1.39e - 1$	$4.47e - 2$	$3.74e - 2$	$2.23e - 2$	$2.45e - 2$	+
	60	3.24	2.37	1.83	1.44	$5.02e - 1$	$3.80e - 1$	+
	100	$6.62e + 1$	$2.58e + 1$	$7.99e + 1$	$3.01e + 1$	2.84	1.22	+
Schwefel	10	$1.18e + 2$	$3.12e - 2$	$1.18e + 2$	$4.63e - 2$	$4.46e - 3$	$6.69e - 3$	+
	60	$5.96e + 3$	$5.44e + 2$	$6.46e + 3$	$1.05e + 3$	$4.92e + 2$	$4.09e + 1$	+
	100	$1.23e + 4$	$2.08e + 2$	$1.84e + 4$	$3.47e + 2$	$4.04e + 3$	$8.99e + 1$	+
Ackley	10	$8.44e - 8$	$5.38e - 8$	$2.81e - 9$	$9.40e - 9$	$1.03e - 11$	$4.92e - 11$	+
	60	4.50	4.55	5.82	3.46	1.65	$8.08e - 1$	+
	100	$2.86e + 1$	8.98	$2.43e + 1$	8.66	3.42	1.01	+

where D is the number of optimization parameters and \bar{X} is the average point which is defined as follows:

$$\bar{X} = \frac{1}{D} \sum_{1 \leq i \leq D} X_i. \quad (12)$$

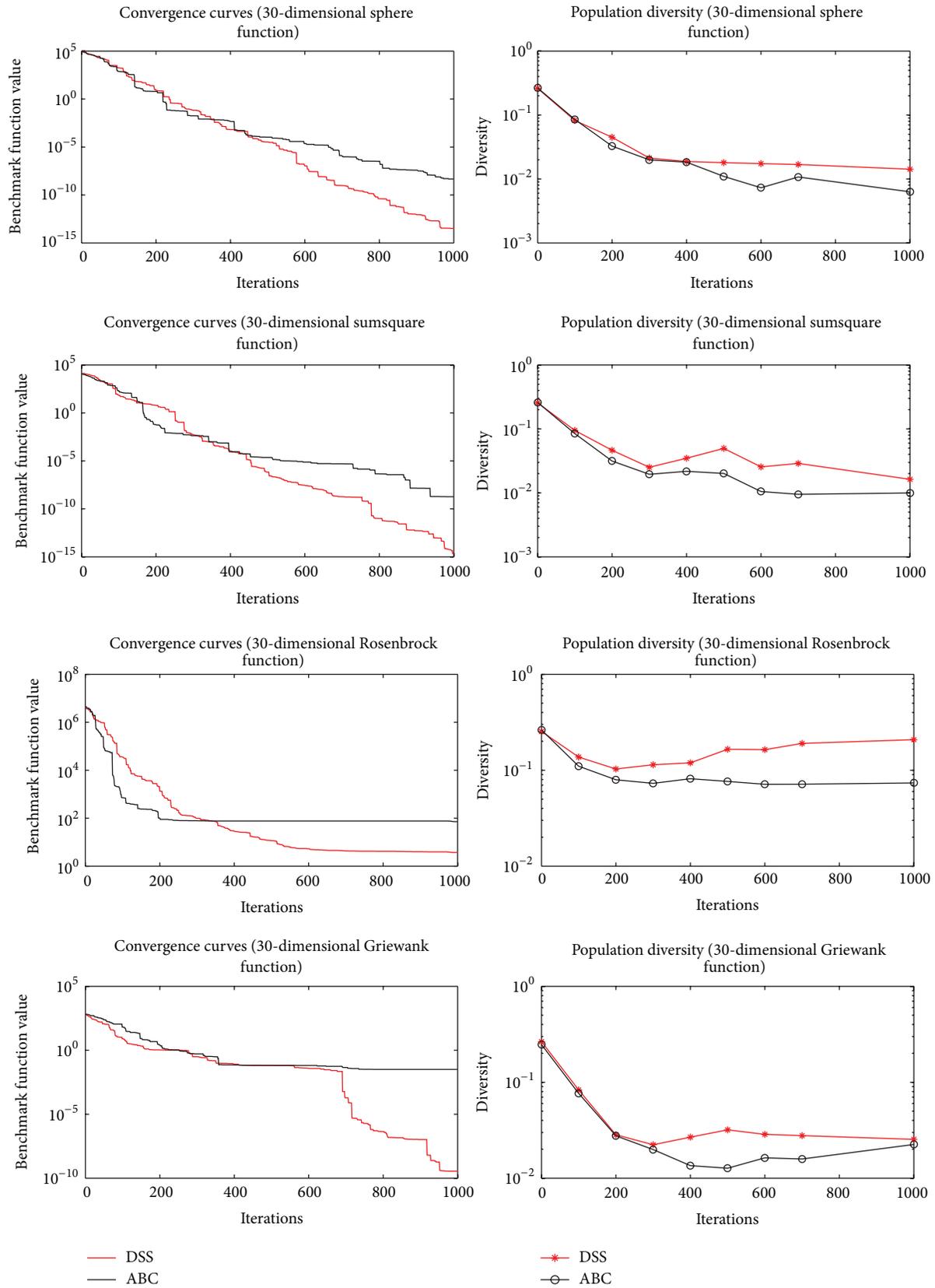
A comparison was conducted between ABC and the standard ABC algorithm with both the disruptive selection strategy and the improved scout bee (DSS for short) on the above six benchmark functions with $D = 30$. The number of population size was set to be 20 and the maximum number of cycles was set to be 1000. Note that the value of *limit* was set to be 150 ($0.5 \times FN \times D$) for ABC [25] and 20 for DSS. Because of changing the definition of the scout bee, we obtained the value of *limit* through testing again and again several times. For all functions, each algorithm was performed 100 independent runs. The convergence performance of different ABCs and their corresponding population diversity are presented in Figure 3. It can be observed that both the population diversity of ABC and DSS are almost the same as each other in the early stage, this is because the same initialization method is applied to obtain initial solutions. It means that they start with almost the same diversity towards global optimum. However, with the increasing iteration times, DSS shows higher population diversity and better convergence performance than that of ABC. For Rosenbrock function, we note that ABC has a fast convergence initially towards the known local optimum. As the procedure proceeds, ABC falls into the local minimum, while DSS gets closer and closer to the minimum. In a word, the better results are obtained by the combined effects of disruptive selection strategy and the improved scout bee.

4.2. Comparison between ABC and SDABC. In this section, a set of experiments were conducted to testify the efficiency of our proposed algorithm on 14 benchmark functions with $D = 30$ and 100; as presented in Table 1, the experiment results were compared with those of ABC. To allow a fair comparison, all the experiments were performed using the following parameter settings: population size was set to be 20; namely, $FN = 10$, and the maximum number of cycles was set to be 3000. It is necessary to emphasize that the value of *limit* was set to be 200 for ABC and 20 for SDABC, and the reason has been shown in the previous section. For all functions, the algorithms carry out 50 independent runs. For more clarity, the best solutions were marked in bold. The results are shown in Table 3 in terms of the best, worst, mean, and standard deviation of solutions obtained by each algorithm over 50 independent runs. In Table 3, “Std” and “AT” denote the standard deviation of solutions and the average elapsed time, respectively. It can be observed that SDABC has higher accuracy on all the functions than ABC. In particular, SDABC can find optimum solutions on functions f_6 , f_8 , f_{10} , f_{13} , and f_{14} with $D = 30$. Moreover, the average elapsed time of SDABC on almost all functions is less than that of ABC except functions f_3 with $D = 100$, f_8 and f_{10} with $D = 30$ and 100. However, the superiority of ABC to SDABC is not significantly obvious in terms of the computational time. In other words, SDABC has higher convergence rate and accuracy.

In addition, the statistical comparison of SDABC with ABC uses a two-tailed t -test at a 0.05 level of significance. Note that value “+” in 9th column in Table 3 represents that our proposed algorithm performs significantly better than

TABLE 3: Best, worst, mean, Std, and AT deviation of solutions obtained by ABC and SDABC through 50 independent runs on 14 functions.

Function	D	Algorithm	Best	Worst	Mean	Std	AT(s)	Significant
f_1	30	ABC	$7.77e-16$	$1.06e-15$	$9.23e-16$	$1.06e-16$	7.47	+
		SDABC	$4.71e-42$	$9.69e-39$	$3.45e-39$	$3.93e-39$	7.29	
	100	ABC	$2.12e-7$	$1.59e-5$	$5.98e-6$	$9.32e-6$	7.66	+
		SDABC	$2.77e-10$	$6.84e-9$	$2.54e-9$	$2.65e-9$	7.62	
f_2	30	ABC	$6.84e-16$	$9.67e-16$	$7.70e-16$	$1.12e-16$	9.05	+
		SDABC	$3.10e-44$	$3.42e-42$	$1.43e-42$	$1.82e-42$	7.29	
	100	ABC	$3.95e-8$	$5.42e-7$	$2.65e-7$	$2.10e-7$	12.65	+
		SDABC	$1.17e-11$	$7.74e-11$	$2.80e-11$	$2.79e-11$	7.52	
f_3	30	ABC	4.01	20.32	9.85	7.04	8.49	+
		SDABC	$4.96e-7$	$6.35e-5$	$2.10e-5$	$2.48e-5$	8.35	
	100	ABC	$1.53e+2$	$2.51e+2$	$2.03e+2$	$3.47e+1$	8.55	+
		SDABC	$4.74e-2$	$1.05e-1$	$6.81e-2$	$2.37e-2$	8.57	
f_4	30	ABC	6.63	$1.55e+1$	$1.13e+1$	3.74	10.02	+
		SDABC	0.38	1.32	0.70	0.37	7.18	
	100	ABC	$8.27e+1$	$9.01e+1$	$8.73e+1$	3.00	8.24	+
		SDABC	$1.05e+1$	$1.40e+1$	$1.18e+1$	1.31	8.19	
f_5	30	ABC	$1.32e-15$	$1.87e-15$	$1.52e-15$	$2.20e-16$	11.66	+
		SDABC	$3.21e-22$	$2.74e-21$	$1.37e-21$	$1.20e-21$	11.46	
	100	ABC	$3.35e-4$	$8.56e-4$	$4.16e-4$	$2.21e-4$	16.24	+
		SDABC	$7.30e-7$	$1.84e-6$	$1.10e-6$	$4.33e-7$	16.11	
f_6	30	ABC	$5.47e-16$	$7.23e-16$	$6.74e-16$	$7.34e-17$	8.32	+
		SDABC	0	0	0	0	8.27	
	100	ABC	$2.35e-7$	$1.20e-6$	$5.74e-7$	$4.51e-7$	8.74	+
		SDABC	$2.60e-10$	$1.05e-9$	$6.49e-10$	$3.01e-10$	8.59	
f_7	30	ABC	$4.34e-17$	$1.06e-16$	$7.06e-17$	$2.52e-17$	12.13	+
		SDABC	$6.11e-86$	$1.69e-78$	$3.39e-79$	$7.58e-79$	12.09	
	100	ABC	$2.59e-12$	$8.27e-10$	$1.95e-10$	$3.56e-10$	22.23	+
		SDABC	$8.67e-26$	$9.11e-20$	$1.82e-20$	$4.07e-20$	22.14	
f_8	30	ABC	$1.79e-8$	$1.39e-6$	$6.38e-7$	$5.51e-7$	8.16	+
		SDABC	0	0	0	0	8.21	
	100	ABC	$1.65e-7$	$2.10e-6$	$1.16e-6$	$7.39e-7$	8.37	+
		SDABC	0	$1.11e-17$	$8.88e-18$	$4.96e-18$	8.43	
f_9	30	ABC	$6.99e-16$	$7.49e-16$	$7.31e-16$	$2.05e-17$	34.88	+
		SDABC	$7.34e-42$	$3.67e-39$	$8.95e-40$	$1.56e-39$	34.82	
	100	ABC	$2.72e-7$	$1.52e-6$	$9.38e-7$	$6.28e-7$	56.84	+
		SDABC	$3.15e-14$	$4.26e-12$	$1.43e-12$	$2.18e-12$	54.21	
f_{10}	30	ABC	$9.99e-16$	$1.73e-2$	$4.19e-3$	$7.51e-3$	13.97	+
		SDABC	0	0	0	0	14.01	
	100	ABC	$1.26e-7$	$1.78e-2$	$2.62e-3$	$9.93e-3$	14.91	+
		SDABC	$1.37e-9$	$4.95e-5$	$9.94e-6$	$2.21e-5$	14.96	
f_{11}	30	ABC	$3.82e-4$	$1.90e+2$	$6.17e+1$	$8.82e+1$	8.90	+
		SDABC	$3.81e-4$	$3.81e-4$	$3.81e-4$	$8.15e-13$	8.86	
	100	ABC	$2.23e+3$	$3.81e+3$	$2.88e+3$	$5.96e+2$	10.12	+
		SDABC	$1.34e-3$	118.45	24.25	17.64	10.06	
f_{12}	30	ABC	$8.35e-14$	$7.41e-13$	$5.99e-13$	$3.89e-14$	9.51	+
		SDABC	$5.06e-15$	$8.61e-14$	$3.62e-14$	$1.54e-15$	9.42	
	100	ABC	$4.98e-4$	$2.03e-3$	$1.22e-3$	$7.05e-4$	10.25	+
		SDABC	$2.10e-5$	$7.04e-5$	$3.66e-5$	$2.01e-5$	10.21	
f_{13}	30	ABC	$5.68e-14$	$2.05e-4$	$6.85e-5$	$1.18e-4$	7.78	+
		SDABC	0	0	0	0	7.65	
	100	ABC	7.19	13.16	9.52	2.42	9.23	+
		SDABC	3.60	7.27	5.65	1.37	9.08	
f_{14}	30	ABC	$1.32e-8$	$1.76e-8$	$1.55e-8$	$1.63e-9$	41.86	+
		SDABC	0	0	0	0	41.60	
	100	ABC	$2.12e-7$	$3.08e-7$	$2.54e-7$	$3.92e-8$	85.12	+
		SDABC	$4.65e-8$	$6.47e-8$	$5.57e-8$	$7.25e-9$	84.35	



(a)

FIGURE 3: Continued.

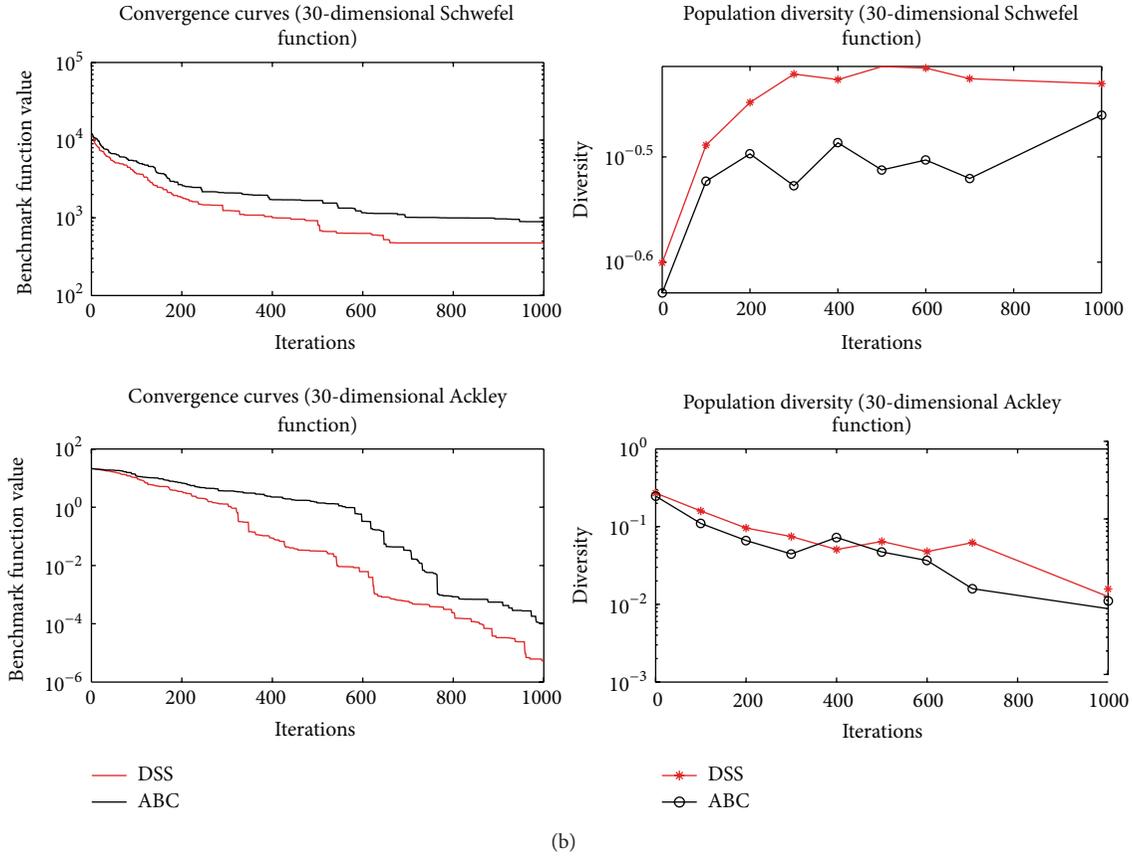


FIGURE 3: Population diversity and convergence performance of different ABCs on six benchmark functions.

ABC algorithm. It also indicates that SDABC outperforms ABC.

4.3. Comparison with Other ABCs. In this section, in order to further prove the superiority of our proposed algorithm, we compared SDABC with other state-of-the-art ABCs which are OABC [16], GABC [12], CABC [14], and OOABC [26] on fourteen functions with $D = 30$ presented in Table 1. Given space limitations, this paper cannot fully explore all functions with different dimensions. Population size was set to be 20 and the maximum number of cycles was set to be 3000. The value of limit was set to be 20 for SDABC and 200 for other ABCs. Other parameter settings can be seen in [12, 14, 16, 26]. In addition, we used a two-tailed t -test at a 0.05 level of significance to compare the results obtained by the best and the second best algorithms. For more clarity, the best and the second best solutions were marked in bold. Note that, the value “NA” denotes not applicable, when the best and the second best algorithms achieve the same solution accuracy. The results are shown in Table 4. It can be seen that the performance of SDABC is significantly superior to that of other ABCs on almost all functions except f_6 . In case of f_6 , both OOABC and SDABC can find optimum solution, which means that they can achieve the same solution accuracy.

5. Conclusions and Future Work

To further improve the performance of ABC, we first proposed a novel initialization method called search space division, which provided high quality of initial solution. Subsequently, a disruptive selection strategy was used to improve population diversity. Moreover, in order to accelerate convergence rate, we changed the definition of the scout bee phase. In addition, we designed two types of experiments to testify our proposed algorithm. On the one hand, we conducted single-factor experiments to make sure of how much each modification makes contribution to improving the performance of ABC. On the other hand, comprehensive experiments were performed to prove the superiority of our proposed algorithm. The experimental results indicate that SDABC significantly outperforms other ABCs, contributing to higher solution accuracy, faster convergence speed, and stronger algorithm stability.

Our future work will focus on two issues. On the one hand, we would apply SDABC to solve the real-world problems such as data mining, industrial design, and clustering. On the other hand, we would extend SDABC to handle the combinational optimization problems.

TABLE 4: Mean and Std deviation of solutions obtained by SDABC and other ABCs through 50 independent runs on fourteen benchmark functions.

Function	OABC		GABC		CABC		OOABC		SDABC		Significant
	Mean	Std									
f_1	8.53e-16	4.65e-16	6.73e-16	2.01e-16	3.55e-16	1.64e-16	5.27e-16	2.46e-16	1.03e-39	2.90e-39	+
f_2	7.74e-16	2.52e-16	6.41e-16	2.56e-16	4.77e-16	1.81e-17	3.37e-16	2.45e-17	1.97e-42	1.24e-42	+
f_3	7.88	4.92	26.32	5.97	24.65	8.92	4.34	2.07	2.96e-5	3.36e-5	+
f_4	5.09	1.57	4.13	1.19	6.99	2.01	10.68	4.37	3.11e-1	2.92e-1	+
f_5	1.40e-15	2.00e-16	1.92e-15	3.44e-16	1.08e-13	1.22e-13	2.38e-14	1.09e-14	8.63e-22	3.00e-23	+
f_6	6.88e-16	2.34e-16	5.56e-16	5.33e-17	4.37e-16	1.16e-17	0	0	0	0	NA
f_7	5.85e-17	3.85e-17	2.00e-17	7.88e-18	2.74e-17	4.50e-18	3.31e-16	5.98e-16	8.56e-80	4.7e-80	+
f_8	1.14e-7	1.37e-7	5.83e-6	5.26e-6	1.32e-5	1.80e-5	4.56e-6	6.67e-6	0	0	+
f_9	5.87e-16	1.79e-16	5.06e-16	4.24e-17	4.66e-16	1.67e-17	8.99e-16	3.08e-16	8.95e-40	1.56e-39	+
f_{10}	4.42e-7	7.65e-7	1.13e-2	1.92e-2	3.39e-2	3.26e-2	9.17e-3	4.23e-3	0	0	+
f_{11}	4.22e+1	7.63e+1	1.49e+2	2.56e+1	1.15e+2	2.89e+1	5.34e+1	7.87e+1	2.84e-4	6.17e-13	+
f_{12}	7.86e-14	5.07e-15	4.67e-14	4.42e-15	5.72e-14	4.19e-15	6.61e-14	3.32e-15	3.02e-14	1.63e-15	+
f_{13}	5.68e-14	1.01e-20	2.05e-11	4.07e-11	4.87e-8	3.63e-8	4.17e-5	6.06e-5	0	0	+
f_{14}	2.23e-8	5.96e-9	2.10e-8	7.90e-9	3.47e-8	1.20e-9	8.43e-9	2.21e-9	0	0	+

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] B. Akay and D. Karaboga, "A modified Artificial Bee Colony algorithm for real-parameter optimization," *Information Sciences*, vol. 192, no. 1, pp. 120–142, 2012.
- [2] K. S. Tang, K. F. Man, S. Kwong, and Q. He, "Genetic algorithms and their applications," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 22–37, 1996.
- [3] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [4] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [5] M. Dorigo and T. Stutzle, *Ant Colony Optimization*, MA MIT Press, Cambridge, Mass, USA, 2004.
- [6] V. Granville, M. Krivanek, and J.-P. Rasson, "Simulated annealing: a proof of convergence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 652–656, 1994.
- [7] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Tech. Rep. TR06, Erciyes University, Kayseri, Turkey, 2005.
- [8] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization*, vol. 39, no. 3, pp. 459–471, 2007.
- [9] S. N. Omkar, J. Senthilnath, R. Khandelwal, G. Narayana Naik, and S. Gopalakrishnan, "Artificial Bee Colony (ABC) for multi-objective design optimization of composite structures," *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 489–499, 2011.
- [10] M. Ma, J. Liang, M. Guo, Y. Fan, and Y. Yin, "SAR image segmentation based on artificial bee colony algorithm," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5205–5214, 2011.
- [11] M. F. Tasgetiren, Q.-K. Pan, P. N. Suganthan, and A. H. Chen, "A discrete artificial bee colony algorithm for the total flowtime minimization in permutation flow shops," *Information Sciences*, vol. 181, no. 16, pp. 3459–3475, 2011.
- [12] G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166–3173, 2010.
- [13] F. Kang, J. J. Li, and H. J. Li, "Artificial bee colony algorithm and pattern search hybridized for global optimization," *Applied Soft Computing Journal*, vol. 13, no. 4, pp. 1781–1791, 2013.
- [14] B. Alatas, "Chaotic bee colony algorithms for global numerical optimization," *Expert Systems with Applications*, vol. 37, no. 8, pp. 5682–5687, 2010.
- [15] W. F. Gao and S. Y. Liu, "Improved artificial bee colony algorithm for global optimization," *Information Processing Letters*, vol. 111, no. 17, pp. 871–882, 2011.
- [16] E. A. Mohammed, "Generalized opposition-based artificial bee colony algorithm," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '12)*, pp. 1–4, Brisbane, Australia, 2012.
- [17] J. Luo, X.-H. Xiao, L. Fu, and Q. Wang, "Modified artificial bee colony algorithm based on segmental-search strategy," *Control and Decision*, vol. 27, no. 9, pp. 1402–1410, 2012.
- [18] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," *Artificial Intelligence Review*, vol. 42, no. 1, pp. 21–57, 2014.
- [19] D. Karaboga and B. Akay, "A survey: algorithms simulating bee swarm intelligence," *Artificial Intelligence Review*, vol. 31, no. 1–4, pp. 61–85, 2009.
- [20] V. Melo and A. C. Delbem, "Investigating Smart Sampling as a population initialization method for Differential Evolution in continuous problems," *Information Sciences*, vol. 193, pp. 36–53, 2012.

- [21] R. S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 64–79, 2008.
- [22] A. Banharnsakun, T. Achalakul, and B. Sirinaovakul, "The best-so-far selection in artificial bee colony algorithm," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2888–2901, 2011.
- [23] T. Kuo and S.-Y. Hwang, "Using disruptive selection to maintain diversity in genetic algorithms," *Applied Intelligence*, vol. 7, no. 3, pp. 257–267, 1997.
- [24] C. Zhang and Z. Yi, "Scale-free fully informed particle swarm optimization algorithm," *Information Sciences*, vol. 181, no. 20, pp. 4550–4568, 2011.
- [25] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 687–697, 2008.
- [26] X. Zhang, X. Zhang, S. Y. Yuen, S. L. Ho, and W. N. Fu, "An improved artificial bee colony algorithm for optimal design of electromagnetic devices," *IEEE Transactions on Magnetics*, vol. 49, no. 8, pp. 4811–4816, 2013.

Research Article

PSO-Based Robot Path Planning for Multisurvivor Rescue in Limited Survival Time

N. Geng, D. W. Gong, and Y. Zhang

School of Information and Electrical Engineering, China University of Mining and Technology, Xuzhou 221116, China

Correspondence should be addressed to D. W. Gong; dwgong@vip.163.com

Received 5 June 2014; Revised 5 August 2014; Accepted 19 August 2014; Published 25 September 2014

Academic Editor: Fang Zong

Copyright © 2014 N. Geng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Since the strength of a trapped person often declines with time in urgent and dangerous circumstances, adopting a robot to rescue as many survivors as possible in limited time is of considerable significance. However, as one key issue in robot navigation, how to plan an optimal rescue path of a robot has not yet been fully solved. This paper studies robot path planning for multisurvivor rescue in limited survival time using a representative heuristic, particle swarm optimization (PSO). First, the robot path planning problem including multiple survivors is formulated as a discrete optimization one with high constraint, where the number of rescued persons is taken as the unique objective function, and the strength of a trapped person is used to constrain the feasibility of a path. Then, a new integer PSO algorithm is presented to solve the mathematical model, and several new operations, such as the update of a particle, the insertion and inversion operators, and the rapidly local search method, are incorporated into the proposed algorithm to improve its effectiveness. Finally, the simulation results demonstrate the capacity of our method in generating optimal paths with high quality.

1. Introduction

Due to the influence of natural hazards and terrorism, various disasters frequently occur in our world. Since robots are very useful in dangerous or abominable environments where human could not reach certain targets, robot rescue in disaster remains an interesting and challenging subject for researchers [1, 2]. However, as one key issue in robot navigation, how to plan an optimal rescue path of a robot, along which the robot could rescue as many survivors as possible in limited time, has not yet been fully solved.

Up to now, much work has been done on robot path planning using various approaches, such as the cell decomposition [3], artificial potential field [4], visibility graph [5], probabilistic roadmap [6], quick random search tree [7], and intelligent search algorithm [8–11], to say a few. However, most of these approaches overlook the fact that survivors have only limited survival time because of atrocious circumstance. Actually, after a disaster, a survivor's strength will sharply decline, so she/he must be rescued within the short period of her/his survival time. In this case, it is of essence to plan a reasonable path so as to rescue as many survivors as possible in limited time.

This paper studies the problem of robot rescue path planning with time-constraint. Since a robot needs to rescue as many survivors as possible in limited time, the survival time of a survivor is introduced to reflect the feasibility of a path, and the number of rescued survivors is used to evaluate the quality of the path. Based on this, the above problem is formulated as a constraint optimization problem. To solve this problem, a modified PSO algorithm is presented. PSO was originally proposed by Kennedy and Eberhard [12]. Since it has a simple structure, fast convergence, and few parameters to be set, it has been widely applied in the field of robot path planning [13–15]. However, there have been no studies on robot rescue path planning with time-constraint.

The main contributions of this paper are as follows. (1) The mathematical model of the problem of robot rescue path planning is formulated, considering that a survivor's survival time is limited; (2) a modified integer PSO is proposed to solve the above model, and several new operators, such as the update of a particle, the insertion and inversion operators, and the rapidly local search method, are incorporated to enhance the capability of the proposed PSO; (3) a series of simulations are done to verify the effectiveness of the proposed method.

The remainder of this paper is organized as follows. Section 2 is a review on related work. The problem of rescue path planning is formulated in Section 3. The modified PSO algorithm used to solve the established model which is elaborated in Section 4. Section 5 is the simulation results. Section 6 draws the conclusion and provides some open questions.

2. Related Work

2.1. Rescue Path Planning of a Robot. There has been much meaningful work on the problem of rescue path planning. In order to search and rescue survivors in a building after a disaster, Kibler et al. constructed a flooding wireless protocol and a flooding fulfill search method and used an autonomous mobile robot as the development platform to verify the effectiveness of the wireless protocol and the search method [16]. Mandal et al. utilized a generalized local Voronoi graph to establish the models of a disaster site, coverage range, and team work and planned a robot path online based on information provided by sensors during rescue [17]. Ferdous et al. proposed a method of transferring people within the shortest time and the minimum cost, which focuses on disasters caused by flood, tsunami, and typhoon. The proposed method can also be used in other time-critical man-made disaster rescue operations [18]. In order to overcome the deficiency that a robot needs human help during the outdoor search and rescue, Doroodgar et al. presented a method of searching outdoors based on the semiautonomous human-computer interaction, which used information from sensors to guide a robot to perform rescue operations [19]. Aiming at the rescue problem in mines, Tian et al. developed a modified neural network to expand the Kalman filter and realized a method of robot path planning based on multisensor fusion [20]. Zhang et al. employed a modified ant colony algorithm (ACO) to the problem of rescue path planning [21]. Basilico and Amigoni introduced a strategy based on a multicriterion decision to solve the problem of robot path planning [22]. In addition, Chien et al. used a robot team to search, rescue, patrol, and do other military tasks, so as to seek for the divergence between human manual control and autonomous path planning [23].

These studies enrich the method of robot rescue path planning. However, the above work omits the fact that a survivor has limited time to wait for rescue, and there have been very few studies on formulating the mathematical model of the rescue problem. Moreover, various intelligent algorithms have been applied to robot path planning but rarely applied to the rescue problem.

2.2. PSO. PSO is a population-based stochastic global optimization technique, originated from the simulation of bird flock and fish school seeking food [12]. In PSO, a swarm of individuals (called particles) fly through the search space. Each particle represents a candidate solution, and particles move in the search space to search for the optimal solution(s) by updating the position of each particle based on the experience of its own and its neighboring particles. The best previous position of a particle is recorded as the personal best,

called *pbest*, and the best position obtained by the swarm so far is the global best, called *gbest*. PSO searches for the optimal solution(s) by updating the position and velocity of each particle according to the following equations:

$$\begin{aligned} V_{ij}(T+1) &= \omega V_{ij}(T) + c_1 r_1 (p_{ij}(T) - X_{ij}(T)) \\ &\quad + c_2 r_2 (p_{gj}(T) - X_{ij}(T)), \\ X_{ij}(T+1) &= X_{ij}(T) + V_{ij}(T+1), \end{aligned} \quad (1)$$

where T denotes the number of generations during the evolution, ω is the inertia weight, c_1 and c_2 represent the acceleration constants, r_1 and r_2 are random values uniformly distributed in $[0, 1]$, and p_{ij} and p_{gj} refer to the elements of *pbest* and *gbest* in the j th dimension.

2.3. Robot Path Planning Based on PSO. Masehian and Sedighzadeh presented two novel PSO-based algorithms for robot path planning and evaluated a path based on two criteria. The experiment results empirically verify the effectiveness of the proposed method. However, this method only works for continuous optimization problem [24]. Aiming at the simulated robot path planning, Adham and Somnuk proposed a modified PSO by extending the search space of a particle [25]. In order to obtain an optimal and smooth path, Lee et al. introduced an online planning method to completely cover a path based on PSO [26]. Considering danger sources in an environment, Zhang et al. took the danger degree and the path length as two objectives, formulated a mathematical model of the path planning problem, and presented an interval multiobjective PSO to seek for solutions of the problem [27]. Since there exist errors during the robot movement, Liu et al. proposed a path planning method of multirobot, avoiding collisions and satisfying the constraints on velocity and acceleration [28]. Tang and Eberhard presented a modified PSO by improving the method of updating a particle based on an enhanced Lagrangian operator and applied it to swarm-robot path planning [29]. In addition, Xue and Liu demonstrated a method of robot path planning based on PSO, which utilizes the danger degree grid to provide information related to an environment, and designed a fitness function based on the weighted sum of the danger degree and the path length [30].

These studies enrich the method of robot path planning. However, all these approaches do not consider the strength of a trapped person. So, they are not suitable for solving the problem of robot rescue path planning with constraints.

3. Formulation of the Problem of Robot Rescue Path Planning

As one key issue in robot rescue, the problem of rescue path planning with time constraint is investigated and formulated in this section.

3.1. Problem Descriptions. In order to describe the problem of rescue path planning, the following assumptions are made.

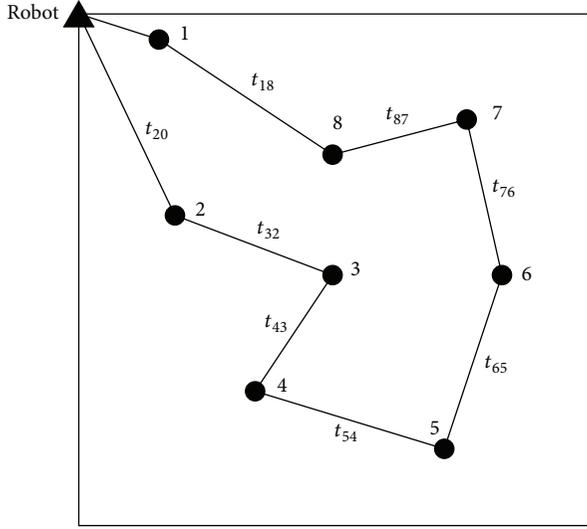


FIGURE 1: Robot rescue environment.

- (1) There is only one robot and its power is enough to complete the whole rescue task.
- (2) The position of a trapped person (target) is known in advance and fixed during rescue.

Given that the two-dimensional workplace contains many targets, the problem of robot rescue path planning can be stated as conducting the rescue task according to the planned path in order to rescue the most targets in limited time. If the strength of a trapped person is smaller than or equal to a threshold set in advance, it is of no necessity to rescue her/him.

Assume that there are N trapped persons, denoted as $1, 2, \dots, N$, and the robot follows a certain sequence, represented as s_1, s_2, \dots, s_N , to move along, where $s_i \in Z$, $1 \leq s_i \leq N$. If the initial position of the robot is regarded as a target, denoted as s_0 , and is set to 0, a path, denoted as $s_0, s_1, s_2, \dots, s_N$, can be obtained according to the above rescue sequence. The time sequence of the robot passing through two adjacent points is denoted as $t_{s_0 s_1}, t_{s_1 s_2}, \dots, t_{s_{N-1} s_N}$, respectively, where $t_{s_i s_{i+1}}$ means the moving time from target s_i to s_{i+1} . Figure 1 depicts the positions of eight targets and a robot, where a target and a robot are represented by a solid circle and a solid triangle, respectively. If the robot follows the path given in Figure 1, the sequence is represented as 1, 8, 7, 6, 5, 4, 3, 2, and the corresponding robot rescue path is 0 1 8 7 6 5 4 3 2.

3.2. Mathematical Model. It is possible that the robot cannot rescue all targets when it follows the sequence given in Figure 1, since a target may die before the robot reaches it, as a result of the strength of this person smaller than the threshold. So, a reasonable path planning strategy is required to rescue the most alive targets. To this end, the number of rescued targets is utilized to evaluate a path, and the limited strength of a target should be considered to evaluate the feasibility of the path.

The rescue sequence is denoted as s_1, s_2, \dots, s_N , which forms a rescue path, S . In order to formulate the objective of the problem, the limited strength of a target is first calculated when the robot reaches her/him; second, the strength is compared with the threshold, and if the former is larger than the latter, the number of the rescued targets increases one; otherwise, this number remains unchanged. If the number of the rescued targets is denoted as $F(S)$, it can be represented as follows:

$$F(S) = \sum_{i=1}^N f(s_i), \quad f(s_i) = \begin{cases} 1, & \sigma_{s_i} - \Delta\sigma > 0, \\ 0, & \text{otherwise,} \end{cases} \quad (2)$$

where $f(s_i)$ represents a sign that reflects whether target s_i is rescued or not, $\Delta\sigma$ is the threshold set in advance, and σ_{s_i} means the strength of target s_i when the robot reached s_i .

Generally, the value of σ_{s_i} at the current time can be estimated by the time spent in the robot moving and that spent in rescuing a target. Based on the method in [31], the value of σ_{s_i} can be estimated by the following equations:

$$\sigma_{s_i} = \sigma_{s_i}^0 e^{-0.037t_{s_i}}, \quad (3)$$

$$\sigma_{s_i}^0 = \sigma_o \cdot \min \left\{ \frac{d_{s_i}}{DS}, 1 \right\}, \quad (4)$$

$$t_{s_i} = \frac{P_{s_i}}{v} + ST, \quad (5)$$

where t_{s_i} is the time spent by the robot in reaching target s_i from the start position. As formula (5) shows, it includes the following two parts: one is the time spent by the robot in moving from the start position to target s_i , equivalent to the path length from target s_i to the start position, P_{s_i} , divided by the velocity of the robot, v ; the other is the time spent by the robot in rescuing a target, equal to ST .

In formula (4), σ_o is the average strength of a healthy person, $d_{s_i} = \sqrt{(x_{s_i} - x_{DS})^2 + (y_{s_i} - y_{DS})^2}$ means the distance between a trapped person and the source of the disaster, DS is the maximum radius of the region influenced by the disaster, and $\sigma_{s_i}^0$ refers to the initial strength (after the disaster) of target s_i . From formula (4), the smaller the distance between a trapped person and the source of the disaster is, the higher the injury degree of this person should be. If the trapped person is far from the source of the disaster, the injury will be slight, the strength will be larger, and the survival time will be longer.

Overall, the problem of rescue path planning can be formulated as follows:

$$\begin{aligned} & \max F(S) \\ & \text{s.t. } \sigma_{s_i} > \Delta\sigma, \quad i = 1, \dots, N. \end{aligned} \quad (6)$$

4. The Proposed PSO Algorithm

In order to solve the problem formulated as formula (6), an effective method is needed.

Since the problem is NP hard, various methods can be employed to solve it, especially genetic algorithm (GA) and PSO. Both PSO and GA are random search algorithms and suitable for solving the above problem. Though they both

have a similar evolutionary process, PSO has memory but crossover and mutation operators which are owned by GA.

Compared with GA, PSO has a different mechanism in sharing information. In GA, information sharing is done among chromosomes. So the whole population moves toward the optimal areas uniformly. While in PSO, only $gbest$ and $pbest$ provide information to other particles, which is a one-way flow of information. Therefore, the search process follows the current optimal solution(s). As a consequence, compared with GA, all particles may converge to the optimal solution(s) more rapidly.

Besides, PSO has many advantages, such as simple structure, fast convergence, and few parameters to be set. So PSO is employed to solve the above problem in this study, with the purpose of expanding the application range of PSO.

There exist various well-established discrete PSO algorithms [32, 33]. However, most of them use the binary encoding method, which cannot reflect the rescue sequence, resulting in unsuitable for solving the above problem. A modified PSO algorithm is thus proposed in this study, and several operators are employed to improve the performance of the proposed algorithm.

Since the above problem is discrete, a new integer coding method is first introduced to encode a solution; then, a method of updating a particle that is suitable for the integer coding and an approach to updating $gbest$ based on the insertion and inversion operators are presented, followed that a rapidly local search method is incorporated to search for a better solution in the neighbor of a particle; finally, an exchange operator is used to improve the diversity of the swarm.

4.1. Particle Coding. For the decision variable, S , in formula (2), its corresponding rescue sequence is s_1, s_2, \dots, s_N , which is encoded as a number of integers, denoted as X , and different decision variables correspond to different coding strings. When PSO is employed to solve formula (2), different particles correspond to different coding strings. The length of a particle's coding string is the same as the number of targets, denoted as N . Assume that a rescue path is 1 2 4 3, and then the path indicates that the robot starts from target 1, goes through targets 2 and 4 sequentially, and at last reaches target 3. So, the number of targets to be rescued is four, and the corresponding particle is encoded as $x = (1, 2, 4, 3)$.

4.2. Particle Updating. Considering the swarm with n particles, its i th particle's position and velocity are $X_i(T) = (X_{i1}(T), X_{i2}(T), \dots, X_{iN}(T))^T$ and $V_i(T) = (V_{i1}(T), V_{i2}(T), \dots, V_{iN}(T))^T$, respectively. $pbest$ and $gbest$ are $p_i(T) = (p_{i1}(T), p_{i2}(T), \dots, p_{iN}(T))$ and $p_g(T) = (p_{g1}(T), p_{g2}(T), \dots, p_{gn}(T))$, respectively. As each element of a particle is an integer, $X_{ij}(T)$ is updated by the following formulas:

$$V_{ij}(T+1) = \lfloor \omega V_{ij}(T) + c_1 r_1 (p_{ij}(T) - X_{ij}(T)) + c_2 r_2 (p_{gi}(T) - X_{ij}(T)) \rfloor, \quad (7)$$

$$X_{ij}(T+1) = \text{Mod}(X_{ij}(T) + V_{ij}(T+1), N) + 1, \quad (8)$$

where $i = 1, 2, \dots, n$, $j = 1, 2, \dots, N$, $\lfloor \cdot \rfloor$ represents the rounding down function, and $\text{Mod}(\cdot, \cdot)$ means the mod function.

Traditional methods of updating a particle focus mainly on a continuous numerical optimization problem. Different from the traditional methods, formula (7) performs the rounding down function on the velocity of a particle, with the purpose of the velocity being an integer after updating. Furthermore, in formula (8), a mod operation is done to $(x_{ij}(T) + v_{ij}(T+1))$, leading to the gained number being an integer in the range of $[1, N]$. As a result, the updated particle can meet the coding requirement.

Although formulas (7) and (8) ensure that each element of the newly generated particle locates in the range of $[1, N]$, there exists the case that a part of elements reappear in a particle. Assume that a particle is $X_i(T) = (1, 2, 4, 3)$, and its newly generated position is $X_i(T+1) = (1, 3, 3, 4)$ after updating. It is obvious that this particle is infeasible.

The particle after update is denoted as $X_i(T+1) = (X_{i1}(T+1), X_{i2}(T+1), \dots, X_{iN}(T+1))$. The following gives a method that makes an infeasible particle, $X_i(T+1)$, feasible by deleting those repeated elements, we call it the correction operator.

Assume that q elements of a particle are the same; in order to obtain a feasible rescue sequence, the same $(q-1)$ elements in particle $X_i(T+1)$ should be deleted, and integers within $[1, N]$ that have not been selected should be added to $X_i(T+1)$ in turn.

Taking the particle, $X_i(T+1) = (1, 3, 3, 4)$, as an example, it has two elements. So the particle should be modified. To this end, the last element with the same value in $X_i(T+1)$ is first deleted and $X_i(T+1) = (1, 3, 4)$. Then, an integer within $[1, 4]$ that has not been selected, that is, $D = (2)$, is selected and added to $X_i(T+1)$. Finally, $X_i'(T+1) = (1, 3, 4, 2)$.

4.3. Update of $pbest$. For the problem of robot rescue path planning, the number of optimal solutions may be more than one. Some optimal solutions may have the same number of rescued targets but with different rescue sequences. So they correspond to different rescue paths. In order to retain these optimal solutions, the strategy of updating $pbest$, given as follows, is different from the traditional one:

$$pbest(T+1) = \begin{cases} pbest(T), & F(X_i(T+1)) < F(pbest(T)) \\ \text{rand}(pbest(T), X_i(T+1)), & F(X_i(T+1)) = F(pbest(T)) \\ X_i(T+1), & F(X_i(T+1)) > F(pbest(T)). \end{cases} \quad (9)$$

If the newly generated solution is better than $pbest$, $pbest$ is updated to the newly generated one; if the newly generated solution is worse than $pbest$, $pbest$ remains fixed; otherwise, $pbest$ selects one from them at random.

4.4. Update of $gbest$. In traditional PSO, $gbest$ plays a very important role in guiding the evolution of particles. The

existing method selects the best one as $gbest$ of particles among all $pbests$ and the current $gbest$. However, given the fact that all particles select the same $gbest$, the swarm is prone to trap into local optima. In order to improve the capability of our algorithm in exploration, a new method of updating $gbest$ is presented based on the insertion and inversion operators.

For the problem investigated in this paper, there may exist the case that different particles have the same fitness. On this circumstance, all particles with the best fitness are first kept in a set, called the optional set. Then, two operators, that is, the insertion and inversion operators proposed in [34], are performed to generate $cbest$ based on the optional set. If $cbest$ is better than $gbest$, $cbest$ is selected as the new $gbest$; otherwise, $gbest$ remains unchanged. The following is the detailed operations about the insertion and inversion operators.

Insertion Operator. For a solution, $p = \{p_1, p_2, \dots, p_i, \dots, p_j, \dots, p_n\}$, first, randomly select an element and a position, denoted as p_i and j . Then, insert p_i into the position after j . As a result, a new solution, denoted as $p' = (p_1, p_2, \dots, p_{i-1}, \dots, p_j, p_i, \dots, p_n)$, can be obtained.

Inversion Operator. For a solution, $p = (p_1, p_2, \dots, p_i, \dots, p_j, \dots, p_n)$, first, randomly select two inverse positions. Then, inverse all the elements between them. Based on this, a new solution is generated, denoted as $p'' = (p_1, p_2, \dots, p_j, p_{j-1}, \dots, p_{i+1}, p_i, \dots, p_n)$, where i and j are the inverse positions and $i < j$.

Compared with the traditional method, the above strategy utilizes different $gbests$ when updating different particles, which is a benefit for the swarm to jump out from local optima of the problem.

4.5. Exchange Operator. Like most PSO-based algorithms, the convergence of our algorithm has a close relationship with the diversity of the swarm. In order to improve the diversity of the swarm, an exchange operator is proposed. For convenient illustration, the concept of entropy is first defined and then used to calculate the diversity of the swarm.

Definition 1 (particle entropy [35]). If the probability of the j th particle occurrence in generation T is ρ_j , its particle entropy is calculated as

$$e_j = -\rho_j \log_2(\rho_j), \quad j = 1, 2, \dots, n, \quad (10)$$

where $\rho_j = \text{count}(X_j(T))/n$ and $\text{count}(X_j(T))$ refers to the number of occurrences of $X_j(T)$ in generation T .

Definition 2 (swarm diversity [35]). The diversity of the swarm in generation T is calculated as follows:

$$E_T = -\sum_{j=1}^n \rho_j \log_2(\rho_j), \quad (11)$$

when all particles in the swarm are the same and the entropy is minimum, that is, $E_T = 0$. If there are no identical particles in the swarm, the entropy is maximum, that is, $E_T = \log_2 n$.

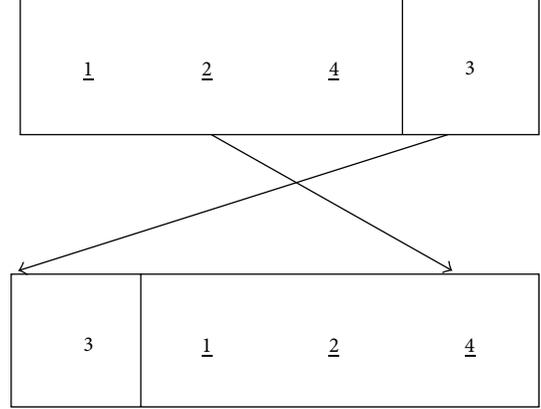


FIGURE 2: Exchange operation.

The diversity of the swarm can be reflected by the entropy value.

If the diversity of the swarm is smaller than a threshold, each particle is performing the exchange operator to generate a new particle. Taking particle $X_i(T) = (X_{i1}(T), X_{i2}(T), \dots, X_{iN}(T))^T$ as an example, first, an element of $X_i(T)$ is randomly selected. Then, the particle is divided into two parts. Finally, the left and the right parts are exchanged to generate a new particle.

Figure 2 shows an example with $X_i(T) = (1, 2, 4, 3)$. Suppose that the element randomly selected is 4; then the new particle after performing the exchange operator is $X_i(T) = (3, 1, 2, 4)$.

4.6. Algorithm Steps. Based on the above strategies, the PSO-based rescue path planning can be described as follows.

Step 1. Initialize the parameters, including the swarm size, n , the largest number of generations, T_{\max} , and the threshold of the diversity, ΔE . Randomly generate the initial swarm, calculate the fitness of each particle, and obtain $pbest$ and $gbest$.

Step 2. Calculate the diversity of the swarm. If the diversity is smaller than ΔE , use the method proposed in Section 4.5 to improve its diversity.

Step 3. Update each particle using the method proposed in Section 4.2.

Step 4. Perform the rapid local search operator proposed in [34] to further exploit the best solution in the neighborhood of each particle.

Step 5. Update $pbest$ of each particle by the method in Section 4.3.

Step 6. Select $gbest$ adopting the method proposed in Section 4.4.

Step 7. Judge whether the swarm evolves for T_{\max} generations. If yes, stop the evolution of the swarm, and output the optimal solution(s); otherwise, go to Step 2.

4.7. Algorithm Analysis. Solis and Wets [36] proposed a sufficient condition that a stochastic optimization algorithm converges to the global optimal solution with the probability of 1. For the purpose of analysis, the main conclusions are restated as follows.

Hypothesis 1. F s.t. $\{f(X^T)\}_{T=0}^{\infty}$ is an increasing function; that is, if $f(F(X, \xi)) \geq f(X)$ and $\xi \in U$, then $f(F(X, \xi)) \geq \max\{f(X), f(\xi)\}$. Where F is the iteration function which was used to generate new individuals, U is the feasible region for the variables.

Hypothesis 2. For a Borel subset of U , denoted as A , if its measure $\theta(A) > 0$, $\prod_{T=0}^{\infty} (1 - \mu_T(A)) = 0$ is held, where $\mu_T(A)$ is the probability of gaining the element(s) in A by adopting some strategies.

Lemma 3. Suppose that f is a measurable function, U a measurable subset, $\{X^T\}_{T=0}^{\infty}$ the resulting sequence generated by an algorithm, and Hypotheses 1 and 2 are held, then $\lim_{T \rightarrow \infty} p(X^T \in R_\varepsilon) = 1$ is held, where $p(X^T \in R_\varepsilon)$ is the probability of generating $X^T \in R_\varepsilon$ by the algorithm in the T th generation and R_ε is the set of global best solutions.

Lemma 4. Assume that the objective, f , that PSO optimizes is measurable, and its solution space, U , is a measurable set, then the modified PSO can converge to the global best solution with the probability of 1.

Proof. From the lemma, we need only to prove that the modified PSO can meet Hypotheses 1 and 2.

(1) The iteration function, F , of the modified PSO can be attributed to the following formula:

$$F(p_g(T), p_i(T)) = \begin{cases} p_g(T), & f(p_g(T) \geq p_i(T)), \\ p_i(T), & f(p_g(T) < p_i(T)), \end{cases} \quad (12)$$

where T is the number of generations and the resulting sequence of the algorithm is $\{p_g(T)\}_{T=0}^{\infty}$. It is clear that formula (12) meets Hypothesis 1.

(2) Set X_i^k as the k th result of particle X_i after the exchange, insertion, inversion, and the rapid local search operations. If these operations run all the time, the distribution of particles remains unchanged. From [36], the union set of particles must include U , that is, $U \subseteq \bigcup_{i=1}^n M_{i,k}$, where $M_{i,k}$ is the k th support set of particle X_i .

In the modified PSO, the velocity of a particle is convergent. These operations have no influences on the result in limited generations, so for $\forall A \subset U$, as long as its measure $\theta(A) > 0$, there is always a particle of the modified PSO that can reach A , which meets Hypothesis 2. \square

Lemma 5. The modified PSO is convergent to the global best solution with the probability of 1.

Proof. Since the insertion and inversion operators perform on the set that is used to supply the global best solution, they have the role of transforming the global best solution, so it is enough to prove that the modified PSO converges to the global best solution with the probability of 1, which is equivalent to formula (7) satisfying $V_i(T) = 0 (T \rightarrow \infty)$. Without loss of generality, the following assumption is given:

- (1) only one swarm evolves in the modified PSO;
- (2) the global best solution after the above operations is approximate to the current global best solution.

Based on the above assumptions, the velocity updating formula of particle can be written as follows:

$$V_{ij}(T+1) = wV_{ij}(T) + r_1c_1(p'_{ij}(T) - X_{ij}(T)) + r_2c_2(p_{gj}(T) - X_{ij}(T)). \quad (13)$$

While $p'_{ij}(T) \approx p_{gj}(T - a)$, and the iteration function of F is bounded and $f(p_g(T)) \leq f(p_g(T+1))$, so

$$\lim [p_g(T) - p_g(T - a)] = 0, \quad (14)$$

and then

$$p'_{ij}(T) \approx p_{gj}(T) + \varepsilon(T), \quad (15)$$

where $\lim_{T \rightarrow \infty} \varepsilon(T) = 0$, so we can get

$$V_{ij}(T+1) = wV_{ij}(T) + (r_1c_1 + r_2c_2)(p_{gj}(T) - X_{ij}(T)) + \varepsilon(T). \quad (16)$$

Now, obviously, the modified PSO algorithm can be approximated as the traditional PSO, and the particle's velocity of the traditional PSO model is convergent, so, the velocities of particles in the algorithm can converge to zero, if the number of generations is enough, i.e., $V_i(T) = 0 (T \rightarrow \infty)$. As a result, the modified PSO thus converges to the global best solution with the probability of 1. \square

5. Simulations

In order to verify the validity of the proposed method, simulations are done by using MATLAB software on a PC computer. The configuration of PC is P4 and 2.66 GHz, and the RAM memory is 512 M.

5.1. Parameters Setting. The related parameters of PSO are set as follows: $c_1 = c_2 = 2$, the linear inertia weight, $\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) * T/T_{\max}$, where $\omega_{\max} = 0.9$, $\omega_{\min} = 0.4$ [37], the swarm size is 200, and the largest number of generations is 500. The environment is a two-dimensional plane of $100 * 100$, the strength threshold is 1.0, the velocity of the robot is 3, ST is equal to 5, the initial strength (before the disaster), σ_o , is equal to 30, the position of the disaster source is $D(65, 70)$, and the maximum radius of the region influenced by the disaster is equal to 150.

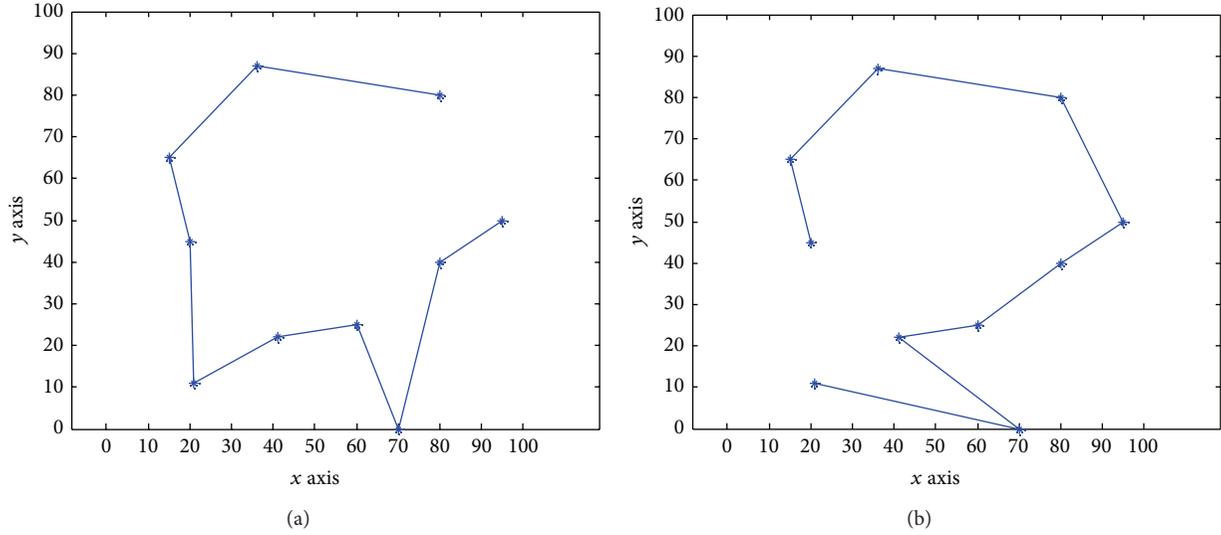


FIGURE 3: Two different rescue paths with the same number of rescued targets: (a) the rescue path corresponding to row one in Table 1, (b) the rescue path corresponding to row two in Table 1.

TABLE 1: Results of rescuing ten targets.

	1	2	3	4	5	6	7	8	9	10	F
1	4	7	6	5	10	8	2	1	3	9	7
2	5	6	7	4	9	3	2	8	1	10	7
3	6	7	4	9	3	2	8	1	5	10	7
4	7	6	5	8	3	2	9	1	4	10	7
5	7	6	5	10	8	3	2	1	4	9	7

Note: F means the number of rescued targets.

5.2. Simulation Results of Two Cases

(a) *Case One: Ten Targets Need to Be Rescued.* Consider the situation that ten targets need to be rescued, and their positions are (70, 0), (60, 25), (80, 40), (80, 80), (20, 45), (15, 65), (36, 87), (41, 22), (95, 20), and (2, 11). From formula (4), the initial strengths (after the disaster) of all these targets are 14.03, 9.05, 6.71, 3.61, 10.30, 10.05, 6.73, 10.73, 11.66, and 17.26, respectively. Employing the proposed method, five sequences of rescuing targets in this case are listed in Table 1, and the first two rescue paths are shown in Figure 3.

Table 1 reports that the robot can rescue seven targets at most in limited time, and there are five optimal rescue paths in total, which provide more than one selection when the secondary disaster occurs.

Figure 4 is the diversity of the swarm with and without the exchange operator. From the figure, at the beginning of the evolution, the two cases have a similar diversity. However, as the number of generations increases, the case with the exchange operator performs better than that without it. Furthermore, Table 2 shows the comparison between the above two cases. Owing to a better diversity, the case with the exchange operator has a larger number of rescue paths.

(b) *Case Two: 100 Targets Need to Be Rescued.* Consider that there exist 100 targets to be rescued, and their positions are

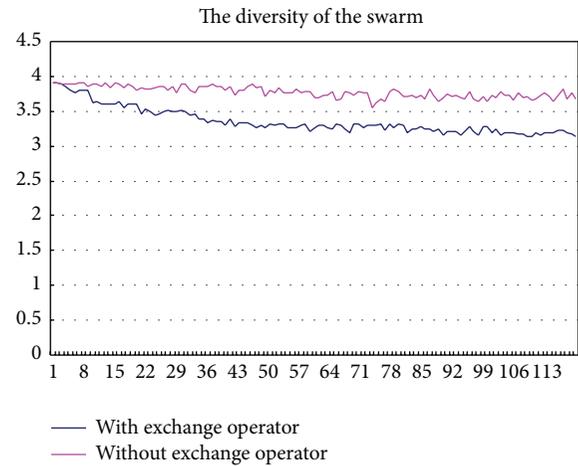


FIGURE 4: The diversity of the swarm with and without the exchange operator.

TABLE 2: Comparison between the cases with and without the exchange operator.

	F	Average number of rescue paths
With EO	7	6.3
No EO	7	2.6

Note: EO means the exchange operator, F refers to the number of rescued targets.

listed in Appendix. Their initial strengths (after the disaster) calculation method is the same as case one. Employing the proposed method, the results are listed in Table 3.

Table 3 reports that with the exchange operator, the robot can rescue 16 targets at most in limited time, and there is only 1.2 rescue path, whereas without the exchange operator, the maximum number of rescued targets is 11, and the average

TABLE 3: Comparison between the cases with and without the exchange operator.

	F	Average number of rescue paths
With EO	16	1.2
Without EO	11	11.9

TABLE 4: Comparison with and without the correction operator.

	F with the correction operator	F without the correction operator
Case 1	7	9
Case 2	16	21
Feasibility of solutions	Feasible	Infeasible

TABLE 5: Comparison between methods of updating the global best particle.

	F with the insertion and inversion operators	F without the insertion and inversion operators
Case 1	7	6
Case 2	16	14

number of rescue paths is 11.9. For the latter case, the number of rescue paths is larger; however, any path cannot complete the rescue task with the largest number of rescued targets.

5.3. Influences of Each Strategy on the Algorithm. Table 4 is the comparison between with and without the correction operation (described in Section 4.2). From Table 4, the maximum numbers of rescued targets for the above two cases are 7 and 16, respectively, whereas those without the correction operator are 9 and 21, respectively. It seems that without the correction operation is better for the two cases. However, some paths are infeasible since they have repeated targets, shown in Figure 5. In this figure, the rescue sequence for ten rescued targets without the correction operation is 10 8 9 1 7 6 7 3 2 10, suggesting that the robot rescues targets 7 and 10 twice. Therefore, it is essential to use the correction operation so as to get a feasible solution.

Table 5 is the comparison between with and without the insertion and inversion operators. Table 5 reports that the case with the insertion and inversion operators is better than its counterpart. The possible reason is that the insertion and inversion operators are added to the mechanism of updating $gbest$, and $gbest$ is selected from the optional set.

Table 6 is the comparison of the above two cases between with and without the rapid local search. From Table 6, the result of the proposed method is better, which is the result that the algorithm can effectively search the neighbor of an individual by using the rapid local search. However, getting a better solution requires more time consumption.

Tables 7 and 8 are the comparison between the rapid local search and the common local search. From these two tables, both methods can gain the best solution for the above two cases. However, in terms of the time consumption, the proposed method has a greater advantage.

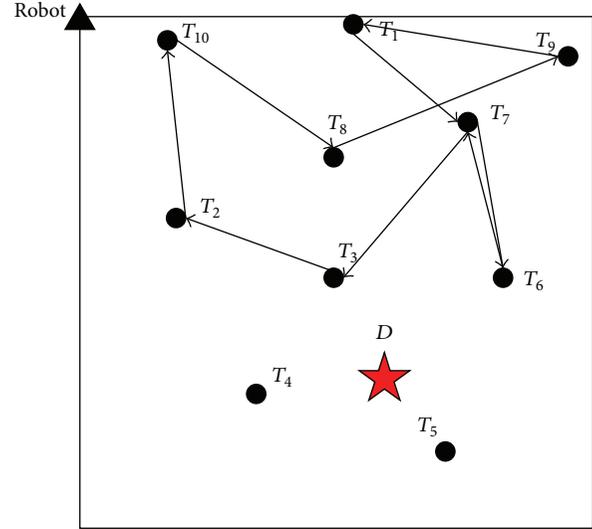


FIGURE 5: Rescue path without the correction operator for 10 targets.

TABLE 6: Comparison between with and without the rapid local search.

	F with the rapid local search	F without the rapid local search
Case 1	7	5
Case 2	16	13

TABLE 7: Comparison between local search methods.

	F of the proposed method	F of the common local search method
Case 1	7	7
Case 2	16	16

TABLE 8: Comparison of time consumption between local search methods (unit: s).

	Mean time of Case 1	Mean time of Case 2
Rapid local search	26.9	320.3
Local search	39.8	429.8

5.4. Algorithm Comparisons

(A) *Comparison with Integer PSO.* The proposed method and integer PSO are the same but the updating strategies, and their comparison results of the above two cases are shown in Table 9.

Table 9 tells that, in case one, the two algorithms are different in the number of rescued targets. The number of rescued targets of the proposed method is 7, whereas that of integer PSO is 6. In case two, their difference is 4, so the proposed method is better than integer PSO. Further, as the number of targets increases, the differences between these algorithms are larger.

(B) *Comparison with GA.* In order to verify the proposed method, the method proposed in this study is compared with

TABLE 9: Comparison between the proposed method and integer PSO.

	F of the proposed method	F of integer PSO
Case 1	7	6
Case 2	16	12

TABLE 10: Comparison between GA and the proposed method.

	F of the proposed method	F of GA
Case 1	7	5
Case 2	16	12

GA, which has the same encode and decode strategy, and the parameters of GA are set as follows: the mutation probability is 0.2, the crossover probability is 0.8, and other parameters are the same as the proposed PSO. These methods are run 30 times, and the average results are listed in Table 10.

From Table 10, the number of rescued targets of the proposed method is larger than that of GA in the two cases, since the proposed PSO algorithm preserves the optimal solution in each particle's memory and employs the rapid local search, whereas the optimal solution obtained by GA may be destroyed in the next generation.

6. Conclusions

For the problem of robot path planning for multisurvivor rescue in limited survival time after a disaster, a modified PSO is proposed to effectively solve it. When formulating the mathematical model of this problem, not only the number of rescued targets is considered but also the survivors' strength. In order to efficiently solve the established mathematical model by employing PSO, the formulas of updating a particle's velocity and position are designed. Also a new mechanism of updating g_{best} is presented, which can generate g_{best} with good performance. In addition, the rapid local search method is employed, with the purpose of helping a particle quickly search good solutions in its neighbor.

The proposed method is applied to two cases and compared with existing methods, and the simulation results show that the robot can quickly find better rescue paths by the proposed method.

The survivors' strength is considered when formulating the mathematical model of the problem, and it can reflect real-world rescue scenarios to some degree. However, some assumptions made in this study are too ideal; for example, the velocity of the robot is constant, the rescue environment remains unchanged before and after the disaster, and no obstacles exist in the rescue environment, to say a few, which would limit the application of the proposed method. So in the following research, it is essential to relax the above assumptions so as to better reflect a practical rescue scenario.

In addition, as a real-world rescue environment is much more complex than that considered in this study, some information and parameters are hard to be obtained, which makes the work done not mature enough to be published. It is another topic that needs to be further studied.

Appendix

The positions of the 100 targets in case two are listed as follows: (70, 0), (60, 25), (82, 40), (80, 80), (20, 45), (15, 65), (36, 87), (41, 22), (95, 50), (2, 11), (73, 10), (60, 35), (8, 50), (20, 90), (20, 55), (15, 76), (36, 70), (41, 12), (95, 52), (12, 17), (76, 8), (60, 15), (29, 30), (80, 70), (20, 35), (15, 55), (36, 47), (41, 12), (95, 40), (2, 31), (71, 20), (60, 70), (11, 90), (89, 20), (20, 2), (15, 40), (36, 65), (41, 6), (95, 40), (2, 15), (70, 27), (60, 10), (88, 40), (26, 60), (20, 30), (15, 32), (36, 65), (41, 60), (95, 80), (25, 10), (70, 35), (60, 42), (80, 28), (60, 0), (20, 45), (15, 50), (36, 15), (41, 10), (95, 9), (2, 22), (74, 28), (60, 18), (80, 69), (82, 28), (20, 40), (15, 18), (36, 23), (41, 94), (95, 21), (8, 14), (70, 22), (60, 23), (80, 24), (80, 30), (20, 26), (15, 75), (36, 76), (41, 15), (95, 71), (21, 85), (70, 14), (60, 63), (70, 55), (80, 80), (20, 14), (15, 99), (36, 60), (41, 11), (95, 100), (20, 52), (0, 19), (25, 71), (40, 51), (84, 80), (45, 45), (65, 65), (87, 87), (22, 22), (50, 50), and (11, 11).

Conflict of Interests

The authors declare no conflict of interests.

Acknowledgments

This research is jointly supported by the Fundamental Research Funds for the Central Universities (no. 2013XK09), the Jiangsu Planned Projects for Postdoctoral Research Funds (no. 1301009B), and the China Postdoctoral Science Foundation funded project (no. 2014T70557). Thanks to Dr. Edward C. Mignot, Shandong University, for linguistic advice.

References

- [1] S.-H. Qian, S.-R. Ge, Y.-S. Wang, and C.-Q. Liu, "Research status of the disaster rescue robot and its applications to the mine rescue," *Robot*, vol. 28, no. 2, pp. 350–354, 2006 (Chinese).
- [2] Z.-J. Cai, D.-B. Sun, Y.-Q. Qin, and N. Li, "Study of robot path planning based on framework space approach," *Computer and Digital Engineering*, vol. 4, no. 34, pp. 88–90, 2006.
- [3] N. Ghita and M. Kloetzer, "Trajectory planning for a car-like robot by environment abstraction," *Robotics and Autonomous Systems*, vol. 60, no. 4, pp. 609–619, 2012.
- [4] W. W. Charles, "Global path planning using artificial potential fields," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 316–321, Scottsdale, Ariz, USA, 1989.
- [5] D. Roy, "Study on the configuration space based algorithmic path planning of industrial robots in an unstructured congested three-dimensional space: an approach using visibility map," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 43, no. 2–4, pp. 111–145, 2005.
- [6] L. E. Kavradi, P. Švestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.
- [7] W. J. Yang, "Anytime synchronized-biased-greedy rapidly-exploring random tree path planning in two dimensional complex environments," *International Journal of Control, Automation and Systems*, vol. 9, no. 4, pp. 750–758, 2011.

- [8] Y.-Y. Han, Q.-K. Pan, J.-Q. Li, and H.-Y. Sang, "An improved artificial bee colony algorithm for the blocking flowshop scheduling problem," *The International Journal of Advanced Manufacturing Technology*, vol. 60, no. 9–12, pp. 1149–1159, 2012.
- [9] A. Z. Mohamed, S. H. Lee, H. Y. Hsu, and N. Nath, "A faster path planner using accelerated particle swarm optimization," *Artificial Life and Robotics*, vol. 17, no. 2, pp. 233–240, 2012.
- [10] M. F. Aly and A. T. Abbas, "Simulation of obstacles' effect on industrial robots' working space using genetic algorithm," *Journal of King Saud University—Engineering Sciences*, vol. 26, no. 2, pp. 132–143, 2014.
- [11] X. Chen, Y. Kong, X. Fang, and Q. Wu, "A fast two-stage ACO algorithm for robotic path planning," *Neural Computing and Applications*, vol. 22, no. 2, pp. 313–319, 2013.
- [12] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of IEEE International Conference on Neural Networks (ICNN '95)*, pp. 1942–1948, Perth, Western Australia, December 1995.
- [13] J. J. Liang, H. Song, B. Y. Qu, and Z. F. Liu, "Comparison of three different curves used in path planning problems based on particle swarm optimizer," *Mathematical Problems in Engineering*, vol. 2014, Article ID 623156, 15 pages, 2014.
- [14] H.-C. Huang, "FPGA-based parallel metaheuristic PSO algorithm and its application to global path planning for autonomous robot navigation," *Journal of Intelligent & Robotic Systems*, 2013.
- [15] Y.-K. Liu, M.-K. Li, C.-L. Xie, M.-J. Peng, and F. Xie, "Path-planning research in radioactive environment based on particle swarm algorithm," *Progress in Nuclear Energy*, vol. 74, pp. 184–192, 2014.
- [16] S. Kibler and D. Raskovic, "Coordinated multi-robot exploration of a building for search and rescue situations," in *Proceedings of the 44th IEEE Southeastern Symposium on System Theory*, pp. 159–163, Jacksonville, Fla, USA, 2012.
- [17] P. Mandal, R. K. Barai, M. Maitra, S. Roy, and S. Ghosh, "Autonomous robot coverage in rescue operation," in *Proceedings of the International Conference on Computer Communication and Informatics (ICCCI '12)*, pp. 1–5, Coimbatore, India, January 2013.
- [18] A. H. M. I. Ferdous, S. M. Masudur Rahamn, M. S. H. Nizami, M. M. Abdul Quadir, and M. W. Ullah, "Path planning for automated robotic rescue system," in *Proceedings of the 16th IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD '12)*, pp. 709–712, Wuhan, China, May 2012.
- [19] B. Doroodgar, M. Ficocelli, B. Mobedi, and G. Nejat, "The search for survivors: Cooperative human-robot interaction in search and rescue environments using semi-autonomous robots," in *Proceedings of IEEE International Conference on Robotics and Automation Anchorage Convention District (RAACD'10)*, pp. 2858–2863, Anchorage, Alaska, USA, May 2010.
- [20] Z.-J. Tian, L.-Y. Zhang, and W. Chen, "Improved algorithm for navigation of rescue robots in underground mines," *Computers and Electrical Engineering*, vol. 39, no. 4, pp. 1088–1094, 2013.
- [21] X. Zhang, M. Wu, J. Peng, and F. Jiang, "A rescue robot path planning based on ant colony optimization algorithm," in *Proceedings of the International Conference on Information Technology and Computer Science (ITCS '09)*, pp. 180–183, IEEE, Kiev, Ukraine, July 2009.
- [22] N. Basilico and F. Amigoni, "Exploration strategies based on multi-criteria decision making for searching environments in rescue operations," *Autonomous Robots*, vol. 31, no. 4, pp. 401–417, 2011.
- [23] S.-Y. Chien, H.-D. Wang, and M. Lewis, "Human vs. algorithmic path planning for search and rescue by robot teams," in *Proceedings of the 54th Human Factors and Ergonomics Society Annual Meeting (HFES '10)*, pp. 379–383, Jacksonville, Fla, USA, October 2010.
- [24] E. Masehian and D. Sedighzadeh, "Multi-objective PSO- and NPSO-based algorithms for robot path planning," *Advances in Electrical and Computer Engineering*, vol. 10, no. 4, pp. 69–76, 2010.
- [25] A. Adham and P.-A. Somnuk, "Applying area extension PSO in robotic swarm," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 58, no. 3-4, pp. 253–285, 2010.
- [26] T.-K. Lee, S.-H. Baek, Y.-H. Choi, and S.-Y. Oh, "Smooth coverage path planning and control of mobile robots based on high-resolution grid map representation," *Robotics and Autonomous Systems*, vol. 59, no. 10, pp. 801–812, 2011.
- [27] Y. Zhang, D.-W. Gong, and J.-H. Zhang, "Robot path planning in uncertain environment using multi-objective particle swarm optimization," *Neurocomputing*, vol. 103, pp. 172–185, 2013.
- [28] S. Liu, D. Sun, and C. Zhu, "Coordinated motion planning for multiple mobile robots along designed paths with formation requirement," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 6, pp. 1021–1031, 2011.
- [29] Q.-R. Tang and P. Eberhard, "A PSO-based algorithm designed for a swarm of mobile robots," *Structural and Multidisciplinary Optimization*, vol. 44, no. 4, pp. 483–498, 2011.
- [30] Y. Xue and H. Liu, "Optimal path planning in complex indoor environment based on improved PSO," *Journal of Computational Information Systems*, vol. 7, no. 6, pp. 2158–2165, 2011.
- [31] Y. Kuwata and S. Takada, "Rescue ability for earthquake causality during the 1995 KOBE earthquake," http://www.lib.kobe-u.ac.jp/handle_kernel/00231322.
- [32] J. Kennedy and R. C. Eberhart, "A Discrete binary version of the particle swarm algorithm," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, pp. 4104–4108, IEEE, Orlando, Fla, USA, October 1997.
- [33] J.-C. Bansal and K. Deep, "A modified binary particle swarm optimization for knapsack problems," *Applied Mathematics and Computation*, vol. 218, no. 22, pp. 11042–11061, 2012.
- [34] L. Wang, Q.-K. Pan, P. N. Suganthan, W.-H. Wang, and Y.-M. Wang, "A novel hybrid discrete differential evolution algorithm for blocking flow shop scheduling problems," *Computers & Operations Research*, vol. 37, no. 3, pp. 509–520, 2010.
- [35] K.-Z. Tang, X. Xiao, J.-H. Jia, and X. Xu, "Adaptive particle swarm optimization algorithm based on discrete stimate strategy of diversity," *Journal of Nanjing University of Science and Technology*, vol. 37, no. 3, pp. 344–349, 2013.
- [36] F. J. Solis and R. J. Wets, "Minimization by random search techniques," *Mathematics of Operations Research*, vol. 6, no. 1, pp. 19–30, 1981.
- [37] Y. H. Shi and R. C. Eberhart, "A modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation (ICEC '98)*, pp. 63–79, Alaska, Alaska, USA, May 1998.

Research Article

Location Prediction-Based Data Dissemination Using Swarm Intelligence in Opportunistic Cognitive Networks

Jie Li,^{1,2} Xingwei Wang,³ Jie Jia,³ Pengfei Wang,⁴ Yan Zhou,⁴ and Zhijie Zhao⁵

¹ Computing Center, Northeastern University, Shenyang 110819, China

² Key Laboratory of Networked Control System, The Chinese Academy of Sciences, Shenyang 110016, China

³ College of Information Science and Engineering, Northeastern University, Shenyang 110819, China

⁴ Software College, Northeastern University, Shenyang 110819, China

⁵ Information and Technology Center of China Mobile Group Liaoning Co., Ltd., Liaoning 110179, China

Correspondence should be addressed to Jie Li; lijie@mail.neu.edu.cn

Received 8 June 2014; Revised 8 August 2014; Accepted 11 August 2014; Published 25 September 2014

Academic Editor: Baozhen Yao

Copyright © 2014 Jie Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Swarm intelligence is widely used in the application of communication networks. In this paper we adopt a biologically inspired strategy to investigate the data dissemination problem in the opportunistic cognitive networks (OCNs). We model the system as a centralized and distributed hybrid system including a location prediction server and a pervasive environment deploying the large-scale human-centric devices. To exploit such environment, data gathering and dissemination are fundamentally based on the contact opportunities. To tackle the lack of contemporaneous end-to-end connectivity in opportunistic networks, we apply ant colony optimization as a cognitive heuristic technology to formulate a self-adaptive dissemination-based routing scheme in opportunistic cognitive networks. This routing strategy has attempted to find the most appropriate nodes conveying messages to the destination node based on the location prediction information and intimacy between nodes, which uses the online unsupervised learning on geographical locations and the biologically inspired algorithm on the relationship of nodes to estimate the delivery probability. Extensive simulation is carried out on the real-world traces to evaluate the accuracy of the location prediction and the proposed scheme in terms of transmission cost, delivery ratio, average hops, and delivery latency, which achieves better routing performances compared to the typical routing schemes in OCNs.

1. Introduction

Cognitive networks [1] have already been prototyped for many commercial and civilian applications. Combined with the social intelligence, cognitive networks promise to support services like citizen journalism, mobile social networking, environmental monitoring, and traffic monitoring by integrating ubiquitous sensing, large-scale data collection, and cloud computing. Opportunistic network [2] provides an ideal solution for promoting the evolution of communication among human beings and machines. Cognitive network technology can be applied to the communication system of opportunistic networks to provide heuristic schemes on the algorithm design and system implementation. Smart handheld devices (e.g., smart phone or PDA) carried by

a large number of participants contribute to opportunistic cognitive networks with their sensing and communication parts [3, 4].

Moreover, by including people in the loop, it is now possible to design applications that can dramatically improve daily lives of individuals and communities. The inherent mobility of participants provides unprecedented spatiotemporal coverage and also makes it possible to observe various events. CarTel [5] is a mobile sensor computing system designed to collect, process, deliver, and visualize data from sensors located on mobile units such as automobiles. PEIR [6] is an application that uses location data sampled from everyday mobile phones to calculate personalized estimates of environmental impact and exposure.

The opportunistic cognitive network is a kind of delay tolerant networks (DTNs), and it utilizes communication opportunities obtained from node movement to relay packets. Mobile nodes are enabled to communicate with each other even if a route connecting them never exists. The basic routing strategy of opportunistic networks is “store-carry-forward.” Messages are routed between the sender and the destination(s), and any possible node can be used as a relay, provided that it is likely to bring the message closer to the final destination. Message transmission is mainly dependent on intermediate relays, so relay node selection is critical for efficient data dissemination in opportunistic cognitive networks.

Since human mobility is mostly unpredictable, conventional routing algorithms for mobile ad hoc networks (MANETs) no longer perform well in the opportunistic cognitive networks. Therefore, new algorithms are required to overcome intermittent connectivity in the opportunistic networks.

Stochastic routing protocols, such as Epidemic [7], First Contact (FC) [8], and Direct Delivery (DD) [9], broadcast messages to any encountered node in order to increase the delivery ratio. Epidemic routing diffuses messages similar to the way viruses or bacteria spread in biology. Whenever encountering another node, a node replicates and transfers messages. After receiving messages, a node will move to other places and continuously replicate and deliver the messages to other encountered ones. First Contact sends messages to the first encountered node without copying the messages to other nodes. Direct Delivery comprises the trade-off between Epidemic and First Contact by finding an optimal number of message copies.

Creating more copies of a message increases the message delivery but decreases the network lifetime. These stochastic routing approaches usually consider the destinations of messages as nodes rather than locations.

Different from stochastic routing, current contact-based routing chooses the most appropriate nodes relaying messages to the destinations based on historical contact information, such as contact times, contact duration, and contact cycle.

Probabilistic Routing Protocol using History of Encounters and Transitivity (PRoPHET) [10] is a context-based routing protocol based on the history of encounters. PRoPHET estimates the delivery predictability for each known destination at each node before passing a message. The estimation is based on the history of encounters between nodes. SimBet [11] uses historical contacts to calculate two metrics, similarity, and betweenness. The similarity is calculated by how frequently a node and its destination have met. The betweenness is calculated by how many nodes which a node has met. However, if the utility metrics are equal, SimBet will prevent its forwarding behavior. To improve this flaw, BUBBLE [12] adds the knowledge of community structure to ensure message dissemination. In addition, the betweenness may be useless if the message is near its destination. SimBetAge [13], an improved version of SimBet, was proposed to address these shortcomings. Spray and Wait [14], that uses only a handful of copies per message, can achieve

comparable delays to an oracle-based optimal scheme that minimizes delay while using the lowest possible number of transmissions.

The work in [15] does routing by simply calculating the delivery probability for a node to be at a location in MobySpace, which is a high-dimensional Euclidean space based on the preknown mobility model. However, the required assumption of that each node has the knowledge about mobility patterns of other nodes in the network makes this work unpractical in realistic scenarios.

For the concerns of optimality, robustness, and flexibility, some routing protocols are developed by inspiration from biology [16–21]. Such routing schemes are also well applied in the transportation domain [22–26], which are based on prediction mechanism to improve the performance of transportation systems.

BeeHive is developed with inspiration of the foraging principles of honeybees [27]. Communication between real bees is modeled by designing intelligent bee agents, which are able to make routing decisions in large and complex topologies. The simulation results conclude that bee agents occupy smaller bandwidth and require less processor time. Artificial Bee Colony algorithm [17] is developed with scanning strategy for periodic vehicle routing problem. The work in [28] presents a biologically inspired discrete-event modeling approach for simulating alternative computer network protocols. Adaptation and probabilistic specifications are introduced into honeybee (BEE) and Routing Information Protocol (RIP) routing algorithms.

The ant colony optimization (ACO) methods have been inspired by operating principles of ants [29], which empower a colony of ants to perform complex tasks such as nest building and foraging [30].

Schoonderwoerd [31] first developed ACO-based approach for routing in telecommunication networks. The basic principle in the approach is about the use of stigmergy in multiagents interaction. Randomized ants traverse the network nodes probabilistically and select the highest probability path. The approach is shown to be sufficiently adaptive and demonstrates robustness across difficult network conditions. Yao et al. [19, 20] improved ACO for delivery routing problem and PROMETShop scheduling problems.

The aforementioned routing algorithms show good performance on message delivery. However geographic coordinates of nodes have little or no correlation with their contact times, so the algorithms do not perform well when each device frequently appears at different regions, as most people daily do.

In this paper, we propose the location-prediction and swarm-intelligence-based data dissemination (LOPSI) algorithm for opportunistic cognitive networks. The LOPSI algorithm is a probabilistic routing protocol combining location prediction and the ant colony optimization. It firstly predicts possible locations of relay nodes and destination(s) in successive time series. The mobile nodes calculate intimacy (contact frequency) with potential relay nodes using ACO and then make a forwarding decision based on node intimacy and probabilities of node mobility.

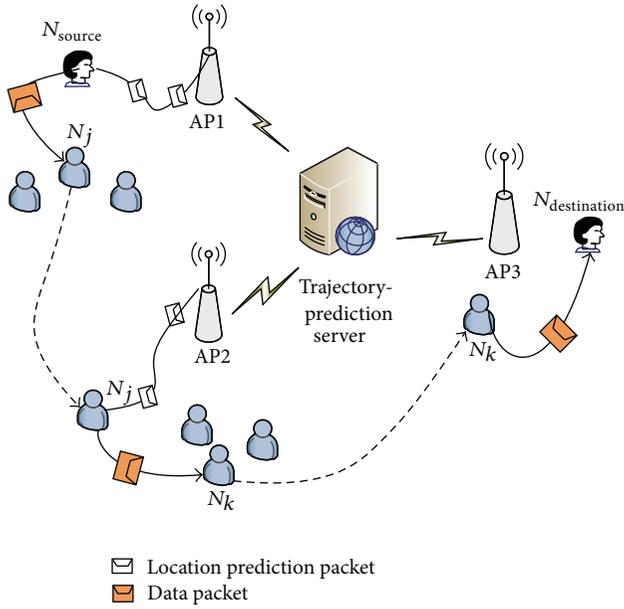


FIGURE 1: Network model of data dissemination in opportunistic cognitive networks.

The remainder of this paper is organized as follows. The system model is illustrated in Section 2. Section 3 specifies the data dissemination algorithm. Extensive simulations have been done for performance evaluation in Section 4. Section 5 concludes the paper.

2. Network Model

As mentioned in Section 1, the location-prediction and swarm-intelligence-based data dissemination (LOPSI) algorithm is a probabilistic routing protocol using location and intimacy information of potential relay nodes and destination nodes. The network is a mixture of an opportunistic network and a centralized infrastructure as shown in Figure 1. The centralized infrastructure consists of a number of wireless access points (APs) and a backbone connecting the APs.

Mobile nodes (carrying smart devices) can only access to the network when they are walking into the transmission range of any AP. Each AP periodically uploads connection records of mobile nodes to the location prediction server (LPS) which will eventually maintain a mobility database of all nodes.

The data exchanged in this scenario is assumed to be of high magnitude, and data transmission can only occurs between peer counterparts as in normal opportunistic networks. The LPS is merely responsible for storing node mobility records and predicting node location upon receiving query from mobile nodes via one of the APs. The accuracy of the localization prediction algorithm can increase the delivery ratio of the proposed approach.

In the proposed system, there exist two major entities: the mobile nodes and the LPS. The mobile nodes work in a decentralized manner, while the LPS is a powerful central unit which collects the trajectories of mobile users and performs

complex computation tasks to provide location prediction service to the request mobile users.

The mobile nodes only store the encountering information of the contact nodes in the local buffer. Based on the encountering information, each mobile node obtains the intimacy between the contact nodes and the destination node and then refines the forwarding probability to the destination node. The mobile nodes can forward the data message to the next hop by the optimal wireless channel according to its own system state. The mobile nodes can solve the data dissemination locally without the need of coordination with a central server or the other clients. Thus the data dissemination process is decentralized.

The LPS is applied to track the movement trajectories of all mobile nodes; thus it needs to collect global movement information predicting the encounter opportunities. Specifically, it uses the long-term trajectory information to construct the Markov chain of a mobile user and to determine the probable mobility trajectory based on partial encountering history. The computations in the server side and in the client side are independent, and their optimization results will not affect each other. Therefore, we have designed three data dissemination algorithms to adapt different infrastructure of the network environment. If the mobile nodes can connect to the LPS, it can obtain the location prediction information to optimize the accuracy of forwarding node selection, whereas the mobile nodes can use the local information, that is, intimacy, to estimate the delivery probability of the contact nodes.

The APs are deemed as living and working locations in the system. Nodes can communicate with each other through short-range communication media, for instance, ZigBee, Bluetooth, NFC, or WiFi Direct. Each node migrates from one location to another according to its own mobility model. When encountering other nodes, the mobile node dynamically calculates intimacy with them using ACO. The intimacy truly reflects historical contact information. It accumulates upon each contact and on the other hand decays over time.

Once a mobile node would like to send data to a destination node, it firstly consults a nearest AP for its neighbors' trend of movement. The LPS performs location prediction using Markov process inference and returns back to the query node an ordered list indicating probabilities of the neighbor nodes meeting the destination node in successive time series (usually more than one time slot).

The mobile node will compute forwarding probability of its neighbors by considering their intimacy with the destination node and then make a forwarding decision.

The mixture network model enhances traditional opportunistic networks with the centralized infrastructure which takes good advantage of existing AP assets but not burdens current network. The data forwarding is a probabilistic scheme guided by location prediction rather than stochastic or trivially probabilistic.

By this means, data will be delivered to the destination with higher probability and hence the network efficiency is improved.

3. Algorithm Design

In this section, we formulate a data dissemination problem in the network environment lacking contemporaneous end-to-end connectivity. To tackle the problem, we propose a contact-based probability routing algorithm, LOPSI, which implements the data dissemination by calculating the forwarding probability based on location prediction scheme and a swarm intelligence heuristic method. Therefore, LOPSI is fundamentally based on two routing schemes that are Location Prediction-Based Data Dissemination (LOPDAD) and Ant Colony Optimization- (ACO-) Based Data Dissemination (ACODAD). LOPDAD uses location prediction information to calculate the maximum probability of the location where the forwarding nodes and the destination node encounter, which is suitable for the opportunistic environment deploying a centralized infrastructure. ACODAD uses the swarm intelligence mechanism, ACO, to select the forwarding nodes according to the intimacy between the forwarding nodes and the destination node, which is suitable for the fully distributed data dissemination in opportunistic networks. LOPSI combines the merits of the two aforementioned algorithms. Depending on the requirements of applications, researchers can select the suitable algorithm to apply.

3.1. Location Prediction-Based Data Dissemination (LOPDAD). Studies on human mobility patterns have shown that people daily activities exist in a high degree of repeatability. People usually visit several fixed places regularly in each day and do activities in a relatively fixed period. According to mobile trajectories and regular behavior pattern of mobile nodes, it can be used to model the scene based on the location of mobile nodes and use relevant algorithms to predict the probability of the node arriving at a certain position and to estimate the location of the mobile node.

Markov chain algorithm is currently the most widely used in location prediction algorithm with high accuracy. Here we use the second-order Markov chain model to predict the location of the mobile node, which has higher accuracy than the first-order Markov chain model according to the simulation.

It can use Markov model to describe the application scenario such as Campus, where it is assumed that there are m locations. Location i is the i th status X_i of Markov process, and the state space is $E = \{X_1, X_2, \dots, X_m\}$. Thus scene mobility model is defined as $\{X, T\}$, and T is time series.

For each application scenario, Markov chain model can be used to predict the future location state of each mobile node. Specific modeling and forecasting process is as follows.

3.1.1. Preparation Process. Preparation before prediction process includes the following steps.

(1) *Determination of State Set.* According to the collection of mobile nodes trajectories from the system server, the location elements in the collecting data are counted, which is denoted as set L . As set L contains a number of location elements,

the locations of higher visiting frequency are chosen as state space of the system, denoted as set E , $E \subset L$.

(2) *Discretization of Data Set.* Statistical data of all users related to state set E is made. Then the dataset of each user is processed to be discrete set of the fixed time period, so the set after discretization is denoted as follows:

$$\{(t_k, X_i)\}, \quad k = 1, 2, 3, \dots, i \in \{1, 2, 3, \dots, m\}. \quad (1)$$

3.1.2. Location Prediction Algorithm Based on O2MM. Order-1 Markov chain model (O1MM) uses the state transition matrix and the initial distribution to predict, which is simple and intuitive [16]. However, as the indeterminacy of the state transition probabilities is unscientific division of the initial state of the system, the prediction result of this method tends to produce larger errors.

Unlike the first order, a Markov chain of higher order is a Markov model with memory, that is, a Markov chain that depends on not only the current state, but also on $n - 1$ states before, where n is the order and n is finite [17]. The chain is dependent on where it is right now and also where it was in the last occasion. Order-2 Markov chain model (O2MM) is used to improve the accuracy of prediction method. Compared with the prediction based on order-1 Markov chain, order-2 Markov chain model can be more complete and rational use of information and effectively integrated with correlation analysis, so as to improve prediction accuracy. O2MM depends on the current state and also the just visited state.

The finite state space of O2MM is

$$E = \{X_1, X_2, \dots, X_r, X_i, X_j, \dots, X_m\}, \quad i = 1, 2, \dots, m, \quad (2)$$

$$r = 1, 2, \dots, m, \quad j = 1, 2, \dots, m$$

and if the conditional probability is

$$P\{X(t_n) = X_j\} \quad (3)$$

$$= P\{X(t_n) = X_j \mid X(t_{n-1}) = X_i, X(t_{n-2}) = X_r\},$$

the transition probability of the node located at X_j at time slice t_n under the condition that the node is located at X_i at time slice t_{n-1} and X_r at time slice t_{n-2} is

$$P_{rij}\{X(t_n) = X_j\} \quad (4)$$

$$= \sum_{i=1, r=1}^m P\{X(t_n) = X_j \mid X(t_{n-1}) = X_i, X(t_{n-2}) = X_r\}.$$

Formula (4) approximately equals the frequency that the node visits the location X_j when the state space tends to infinity [18]:

$$P_{rij} = \frac{\sum_{r=1, i=1}^{r=m, i=m} C_{rij}}{\sum_{k=1}^m C_{rik}}, \quad (5)$$

- (1) **input:** State Space Set $E = \{X_i, i \in \{1, 2, 3, \dots, m\}\}$, Nodes Set $N = \{N_j, j \in \{1, 2, 3, \dots, n\}\}$, the initial probability distribution is $P(n-1, n-2) = \{p_{ri}, r, i \in \{1, 2, 3, \dots, m\}\}$;
- (2) **Discretization of data set:** Statistical data of all users related to state set E is made. Then the data set of each user is processed to be discrete set of the fixed time slice, so the set after discretization is denoted as follow: $\{(t_k, X_i), k \in N^+, i \in \{1, 2, 3, \dots, m\}\}$;
- (3) calculate the probability of the node to visit location X_j according to (5) where the location state of the node at current time slice and also the just visited state is respectively X_i and X_r .
- (4) Calculate one step transition probability matrix according to (6);
- (5) Calculate the probability of each state at time slice t_n
 - (a) $P(n) = P(n-1, n-2)P$
- (6) the location state at time slice t_n is
 - (b) $X_j = \arg \max \{P_j^{(n)}\}$
- (7) **return** X_j ;

 ALGORITHM 1: L -Markov(N_c, t_n): location prediction based on O2MM.

where c_{rij} is the number of times that the observation node visits location X_j by records statistics, $\sum_{k=1}^m c_{rik}$ is the total number of times that the node visits all the locations in E , and then the probability of the node to visit location X_j and the location state of the node at current time slice and also the just visited state is, respectively, X_i and X_r .

If there are m location states in state space set, one-step transition probability matrix is a $m \times m^2$ matrix:

$$P = \begin{bmatrix} P_{111} & P_{112} & \cdots & P_{11(m-1)} & P_{11m} \\ P_{211} & P_{212} & \cdots & P_{21(m-1)} & P_{21m} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{m11} & P_{m12} & \cdots & P_{m1(m-1)} & P_{m1m} \\ P_{m21} & P_{m22} & \cdots & P_{m2(m-1)} & P_{m2m} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{mm1} & P_{mm2} & \cdots & P_{mm(m-1)} & P_{mmm} \end{bmatrix} m^2. \quad (6)$$

The description of the location prediction algorithm based on O2MM is as shown in Algorithm 1.

According to Algorithm 1, given the initial probability distribution of the node at time slice $n-1, n-2$, we can recurrence the probability distribution at time slice t_n by O2MM. The state set $E = \{X_1, X_2\}$, $m = 2$, the initial state is $\{(t_{n-2}, X_1), (t_{n-1}, X_2)\}$, and the initial probability distribution is

$$\begin{aligned} &P(n-1, n-2) \\ &= (p_{11}^{(n-1, n-2)}, p_{12}^{(n-1, n-2)}, p_{21}^{(n-1, n-2)}, p_{22}^{(n-1, n-2)}) = (0, 1, 0, 0). \end{aligned} \quad (7)$$

The transition matrix for Markov chain of order two is

$$P = \begin{bmatrix} P_{111} & P_{112} \\ P_{121} & P_{122} \\ P_{211} & P_{212} \\ P_{221} & P_{222} \end{bmatrix}. \quad (8)$$

Here the first two numbers of the index representing the current state and the last number represent the next state. And the probability distribution at time slice n is

$$P(n) = P(n-1, n-2)P = \{p_1^{(n)}, p_2^{(n)}\}. \quad (9)$$

The location state is obtained by the location prediction server at time slice n is

$$X_j = \arg \max \{p_1^{(n)}, p_2^{(n)}\}. \quad (10)$$

In our system the location prediction server gathers the location data of the mobile nodes by the APs. The server executes Algorithm 1 to predict the trajectory of each node according to the discrete time slice.

3.1.3. Transmission Probability of LOPDAD. The data dissemination mechanism can use the result of the location prediction algorithm using O2MM. At a certain time slice, the data forwarding probability equals the probability of the forwarding node visiting the location where a destination node is, which is given by

$$p_L^d = p_{X_j^d}^{(n)}, \quad (11)$$

where X_j^d is the location state of destination node N_d at time slice t_n , and $p_{X_j^d}^{(n)}$ is the probability that node N_c visits the location X_j^d at time slice t_n .

Algorithm 2 describes the process of the location-based data dissemination algorithm, which obtained the forwarding probability p_L^d of N_c to the destination node N_d .

3.1.4. The Execution of Location Prediction. The LPS predicts the locations where the destination node and the forwarding nodes will encounter at the future time slices. The threshold of time slices is l . At time slice, t_n , when the LPS receives the service request information $\text{REQ}(C\{N_f\}, N_d)$ which includes the destination node N_d and the encounter nodes set $C\{N_f\}$, from the data carrier node N_s , the LPS calculates the

Input: State Space Set $E = \{X_i, i \in \{1, 2, 3, \dots, m\}\}$, N_d, N_c , the initial probability distribution is $P(n-1, n-2) = \{p_{ri}, r, i \in \{1, 2, 3, \dots, m\}\}$;

Output: p_L^d

- (1) $X_j^d = L_Markov(N_d, t_n)$
- (2-5) Algorithm 1 steps 1-5;
- (6) calculation of forwarding probability according to (11)
- (7) return p_L^d ;

ALGORITHM 2: $P_Markov(N_c, t_n)$: data dissemination probability based on location prediction.

- (1) $N_s \rightarrow LPS: REQ(C\{N_f\}, N_d) \ i = 0, j = 0, k = 0, l = 3$;
- (2) **for all** $n \in [1, l]$ **do**
- (3) $X(N_d, t_n) = L_Markov(N_d, t_n)$;
- (4) **for all** $N_f \in C\{N_f\}$ **do**
- (5) $X(N_f, t_n) = L_Markov(N_f, t_n)$; //calculate the location state of N_f at t_n
- (6) **if** $(X(N_f, t_n) == X_i(N_d, t_n))$ **then** // N_f and N_d encounter at t_n
- (7) $F.N[j++] = N_f$; // N_f is stored as the forwarding node in set $F.N$
- (8) $F.P[i++] = P_Markov(N_f, t_n)$; //The probability of N_f meets N_d is stored in set $F.P$
- (9) $F.T[k++] = n$; //the encounter time slice of N_f
- (10) **end if**
- (11) **end for**
- (12) **end for**
- (13) $LPS \rightarrow N_s: SEI(F\{N_f, P, T\}, \{X(N_d, t_n)\})$;

ALGORITHM 3: $LoP_Service(C\{N_f\}, N_d)$: the location prediction service in the LPS.

location state set $X = \{X(N_d, t_l)\}$ of N_d in the time slice $t_{n+1}, t_{n+2}, \dots, t_{n+l}$:

$$\{X(N_d, t_l)\} = \{(t_1, X_i), (t_2, X_j), \dots, (t_l, X_p)\} \subseteq \{(t_k, X_i)\},$$

$$k \in N^+, \quad i, j, p \in \{1, 2, 3, \dots, m\}. \quad (12)$$

And the LPS also calculates the forwarding probability F . P of each node N_f in the set $C\{N_f\}$. The forwarding nodes set $F.N$ in which the node N_f will visit the location where the destination node N_d locates at the same time slice, and the encounter time slice is recorded in $F.T$. Finally, the server sends the service information $SEI(F\{N_f, P, T\}, \{X(N_d, t_n)\})$ to the node N_s . The execution process is described in Algorithm 3.

3.1.5. The Selection Mechanism of the Forwarding Nodes Set. When the data carrier N_s receives the SEI from the predictive server, considering the cache management and the load of the network, the number of copies of the forwarding data is a fixed value, COPY, which is decided by the average buffer size of each node and the current load of the network. If N_s finds that the location of N_d is the same as N_s during the threshold time slice, it only transmits to the nodes which can encounter N_d at earlier time slice than N_s . Otherwise, if the number of nodes in the forwarding nodes set is less than COPY, N_s transmits the data to the nodes in the forwarding nodes set and delete the data in its own buffer. If the number of nodes

in the forwarding nodes set is more than COPY, N_s transmits the data only to the nodes having maximum probability no more than COPY, according to the following:

$$P_{L_{sf}}^d = \left\{ \frac{P_Markov(N_f, t_n)}{n}, N_f \in F, t_n \in T, n \in [1, l] \right\}, \quad (13)$$

where $P_Markov(N_f, t_n)$ is the forwarding probability which is equal to the probability that N_f encounters N_d at time slice t_n , where n is the number of time slice intervals when N_f meets N_d . The larger the number of time slice intervals is, the lower the probability of forwarding node is. And the forwarding nodes set F' which is selected by N_s :

$$F' \subseteq F, \quad F' \{N_f\} = \underset{f \leq COPY}{\operatorname{argmax}} \left\{ P_{L_{sf}}^d \right\}. \quad (14)$$

3.2. Swarm Intelligence Heuristic Data Dissemination (ACO-DAD). The inspiring source of ACO is the pheromone trail laying and following behavior of real ants which use pheromones as a communication medium. Artificial ants used in ACO are stochastic solution construction procedures that probabilistically build a solution by iteratively adding solution components to partial solutions by taking into account (i) heuristic information on the problem instance being solved, if available, and (ii) (artificial) pheromone trails which change dynamically at run-time to reflect the agents' acquired search experience [19]. In our algorithm,

ACO is improved to be applied in data dissemination in opportunistic cognitive networks.

3.2.1. Ant Colony Optimization- (ACO-) Based Data Dissemination in OCN (ACODAD). The pheromone in ACODAD is the intimacy between two nodes. The more frequency and continuous the contact between two nodes is, the higher the value of intimacy is. It means that the higher the pheromone is. The data carrier node tends to choose the node which has high intimacy value with the destination nodes to forward the data. It means that the data forwarding probability of the node having high intimacy with the destination node is high. The comparison of the characteristics between ACODAD and ACO is shown in Table 1.

3.2.2. Intimacy between Two Nodes in OCN. Each node in opportunistic cognitive networks maintains a relationship table with other nodes using the value of intimacy.

For example, at time slice t_s , the data carrier node N_s arrives at location X_i and senses all the other nodes within the communication range via the communication channels (such as ZigBee, Bluetooth, NFC, and other short-range communication protocols). Those nodes are added in encounter nodes set $C\{N_f\}$ of N_s and are recorded in the relationship table with the calculation results of intimacy.

The intimacy between two nodes N_i and N_j depends on the frequency of two nodes in connection based on contact times n_{ij} , lasting time of one connection ΔD_{ij}^c , and the encounter intervals between two contacts ΔI_{ij}^c . The mathematical description of intimacy is as follows, where $R_{ij}(t)$ is the intimacy of N_i and N_j at time slice t :

$$R_{ij}(t) = \begin{cases} \frac{n_{ij} \times \sum_{c=1}^{n_{ij}} \Delta D_{ij}^c}{\sum_{c=1}^{n_{ij}} \Delta I_{ij}^c}, & \forall \Delta I_{ij}^c < k \\ \frac{n_{ij} \times \sum_{c=1}^{n_{ij}} \Delta D_{ij}^c}{\sum_{c=1}^{n_{ij}} \Delta I_{ij}^c} \times (1 - \rho)^e, & \exists \Delta I_{ij}^c > k \\ 0, & \Delta I_{ij}^c > T, \text{ over the system running time,} \end{cases} \quad (15)$$

where ρ is the evaporation rate, k is the threshold of encounter time intervals of two nodes, and e is the times when the time intervals is more than k .

Although the intimacy is defined in consideration of encounter time intervals, if the two nodes are not in connection for a long time (more than a certain time threshold value k), it enables the intimacy evaporation mechanism to ensure the most frequently contact nodes with a high degree of intimacy. If the encounter time intervals are beyond the system running time T , the value of intimacy is zero.

When two nodes encounter, they, respectively, compute the intimacy based on records of encounter time t_s and the departure time t_e at the contact times c . Each node records

the time according to its own time clock. When $c = 1$, ΔI_{ij}^c equals the encounter time of the first contact. The algorithm of intimacy is as shown in Algorithm 4.

Algorithm 4 captures the essence of (15). The intimacy between two nodes is used to compute the forwarding probability in ACODAD.

The intimacy updates during the time duration Δt ; the variation of the intimacy is given by

$$R_{ij}(t + \Delta t) = R_{ij}(t) + \Delta R_{ij}(t),$$

$$\Delta R_{ij}(t) = \begin{cases} \left(\frac{(n_{ij} + \Delta n_{ij}) \times \sum_{c=1}^{(n_{ij} + \Delta n_{ij})} \Delta D_{ij}^c}{\sum_{c=1}^{(n_{ij} + \Delta n_{ij})} \Delta I_{ij}^c} - \frac{n_{ij} \times \sum_{c=1}^{n_{ij}} \Delta D_{ij}^c}{\sum_{c=1}^{n_{ij}} \Delta I_{ij}^c} \right), & \Delta n_{ij} > 0, \forall \Delta I_{ij}^c < k \\ -\rho \times \frac{n_{ij} \times \sum_{c=1}^{n_{ij}} \Delta D_{ij}^c}{\sum_{c=1}^{n_{ij}} \Delta I_{ij}^c}, & \Delta n_{ij} = 0, \Delta t > k, \forall \Delta I_{ij}^c < k \\ 0, & \text{otherwise.} \end{cases} \quad (16)$$

3.2.3. Transmission Probability of ACODAD. When the data carrier node N_s encounters the nodes in set $C\{N_f\}$ and chooses the forwarding nodes from $C\{N_f\}$ to the destination N_d based on the intimacy between the node in $C\{N_f\}$ and N_d , the forwarding probability is defined as the following equation:

$$P_{R_{sf}}^d = \begin{cases} 1 & j = d \\ 0 & \text{Intimacy}(N_f, N_d) < \text{Intimacy}(N_i, N_d) \\ \frac{[R_{fd}(t)]^\alpha \cdot [\eta_{fd}(t)]^\beta}{\sum_{N_f \in \text{allowd}_d} R_{jd}^\alpha \cdot \eta_{jd}^\beta} & N_f \in \text{allowd}_d, \end{cases} \quad (17)$$

where allowd_d is given by

$$\text{allowd}_d^t = F\{N_f \mid (R_{fd}(t) > R_{id}(t))\}, \quad (18)$$

$$\text{allowd}_d^t \in F\{N_f\}, \quad \text{allowd}_d = \text{allowd}_d^t - \text{tabu}_d,$$

where tabu_d is the node set including those nodes which have already carried the transmission data copy and also the node had the copy before. Such nodes will not be selected as the forwarding nodes.

The parameters α and β control the relative importance of the pheromone versus the heuristic information, η_{fd} , which is given by

$$\eta_{fd} = \frac{1}{L_{fd}}, \quad (19)$$

where L_{fd} is the time slice intervals that N_f and N_d will encounter which is an estimation value given by the records in the contact vector.

A heuristic value η , respectively, represents a priori information about the problem instance definition or run-time information provided by a source different from the ants.

TABLE 1: Comparison of characteristics between ACODAD and ACO.

Characteristics	ACODAD	ACO
Transmission process of data	Data dissemination based on the encounter of two nodes	Artificial ants move from one location to the neighbor one
Transition probability	The probability that the forwarding node encounters the destination node	The state transition probabilities that are from one location to the next
Pheromone	Intimacy between two nodes	Pheromone trail laying by ants
Path length	The interval of time slices that two nodes encounter	The distance between two locations
Pheromone evaporation	No contact between the two nodes during threshold time	As time goes by the pheromone evaporates

```

Input:  $N_i, N_j$ 
Output:  $R_{ij}(T)$ 
(1) define  $T$  the system running time
(2)  $L$  encounter time interval threshold
(3) Contact vector array  $\vec{M}_{ij}(c, t_s, t_e)$ 
(4) initialization:  $\vec{M}_{ij}[0] = (0, 0, 0)$ ,  $R_{ij}(0) = 0$ ,  $k = L$ ,  $flag = 1$ ;
(5) forall  $t = 0 : T$ ;
(6) if  $N_i$  recieve HELLO from  $N_j$ 
(7)  $n_{ij} = ++c$ ;
(8) for all  $c = 1 : n_{ij}$ 
(9)  $\Delta D_{ij}^c[c] = t_e - t_s$ ;
(10)  $D_{ij}^c += \Delta D_{ij}^c$ ;
(11)  $\Delta I_{ij}^c[c] = t_s - t$ ;
(12)  $I_{ij}^c += \Delta I_{ij}^c$ ;
(13)  $t = t_e$ ;
(14) if ( $\Delta I_{ij}^c[c] > k$ )
(15)  $e++$ ;
(16)  $flag = 0$ ;
(17) end if
(18) end for
(19) end if
(20) end for
(21) if ( $flag == 0$ )
(22) return  $R_{ij}(T) = ((n_{ij} \times D_{ij}^c) / I_{ij}^c) \times (1 - \rho)^e$ 
(23) else
(24) return  $R_{ij}(T) = (n_{ij} \times D_{ij}^c) / I_{ij}^c$ 
(25) end if

```

ALGORITHM 4: Intimacy(N_i, N_j) computes the intimacy value between two nodes at encounter time T .

```

(1) Initialization: node set  $N = \{N_i, i = 1, 2, \dots, k, k \in N^+\}$ , state space  $E = \{X_j, j = 1, 2, \dots, m, m \in N^+\}$ , time slice set  $T = \{t_n, n = 0, 1, 2, \dots, l, l \in N^+\}$  Maximum data copies: COPY;
(2) for  $n = 0 : l$  // the system running time
// the function executes the one hop data dissemination to the forwarding node set
(3) LOPSI_Sec( $N_s$ )
(4) for  $\forall N_f \in F^l\{N_f\}$ 
(5) execution of LOPSI_Sec( $N_f$ );
(6) end for
(7) end for

```

ALGORITHM 5: LOPSI(N_s, N_d).

In many cases η is the cost, or an estimate of the cost, of extending the current state. These values are used by the ants' heuristic rule to make probabilistic decisions on how to move on the graph [32].

In our algorithm, the heuristic value is defined as the distance between N_f and N_d . As the location state of nodes is discretized by the time slice. And the delay of data forwarding between the two nodes is not decided by the absolute path length or the distance between the two nodes but by calculating the number of time slice intervals between forwarding node and the destination node to meet with each other.

The data carrier node N_s chooses the forwarding nodes from set $C\{N_f\}$ according to the forwarding probability based on intimacy. Considering the cache management, our algorithms set the maximum copy quantity of a unique data message. If the data forwarding operation is executed, N_s will transmit the data to the nodes in set $C'\{N_f\}$, which is given by (20), and the quantity of nodes, denoted by f , is no more than COPY. Finally, N_s will delete the data stored in its buffer:

$$C' \subseteq C\{N_f\}, \quad C'\{N_f\} = \operatorname{argmax}_{f \leq \text{COPY}} \left\{ P_{R_{sf}}^d \right\}. \quad (20)$$

3.3. Location Prediction Based Data Dissemination Using Swarm Intelligence (LOPSI). The data dissemination algorithm LOPSI considers not only the intimacy between the forwarding node and the destination node, but also the location where the two nodes may encounter. According to the location prediction algorithm, the set of forwarding nodes, $F\{N_f\}$, which visit the location where the destination node will be during the threshold time slices, can be obtained. And the data carrier node N_s only sends message to nodes in $F\{N_f\}$ and compares the intimacy between N_d and $N_f \in F\{N_f\}$. Combined with the prediction results from LOPDAD and ACODAD, the transmission probability from node N_s to the forwarding node N_f is calculated by the weight formula

$$p_{sf}^d = \gamma * P_{R_{sf}}^d + \delta * P_{L_{sf}}^d, \quad \gamma + \delta = 1, \quad (21)$$

where $p_{R_{sf}}^d$ is the forwarding probability obtained from ACODAD, $p_{L_{sf}}^d$ is the forwarding probability obtained from LOPDAD, and parameters γ and δ are the weights of the two probabilities.

In our campus environment the mobile node can communicate to the location prediction server anywhere and anytime. The prediction server provides only location prediction service and no other services. Algorithm 5 describes the whole execution process of LOPSI. LOSI_Sec is part of Algorithm 5, which is a function that the data message transfers from the current node to the forwarding node set. When the source node N_s wants to send the data message to the destination node N_d , N_s will choose the potential nodes by location prediction schemes and then obtain the intimacy between the potential nodes and the destination node. Thus the forwarding node set will be determined. The execution process of LOPSI is described in Algorithm 5.

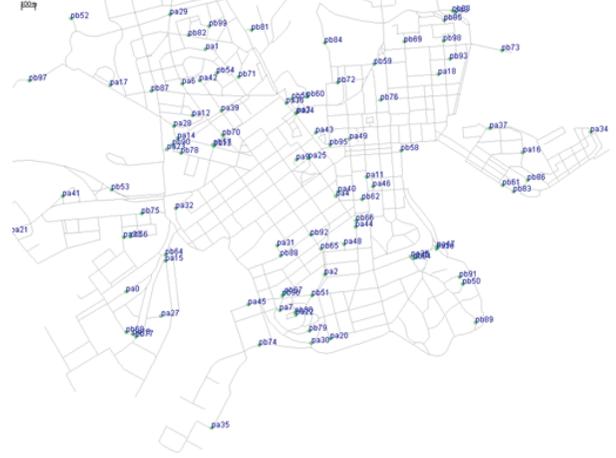


FIGURE 2: The simulation based on a realistic campus scenario.

TABLE 2: The accuracy of location prediction algorithms based on O1MM and O2MM.

	O1MM	O2MM
Prediction accuracy	0.5610	0.8030
Time complexity	$O(N)$	$O(N^2)$
Storage space	$O(N^2)$	$O(N^3)$

For any node N_i carrying data message, $\text{LOPSI.Sec}(N_i)$ will determine the forwarding node set. The description of $\text{LOPSI.Sec}(N_i)$ is given as shown in Algorithm 6.

Algorithm 5 describes the essence of LOPSI. Each data has a TTL, which indicates how long the data can live in the network. It is set by the provider at the time of data generation. In Algorithm 5 the length of time slices l equals TTL. The data carried by each node within TTL duration cannot be forwarded and then be automatically discarded.

4. Performance Evaluation

In this section, we present the simulation results to demonstrate the performance of proposed data dissemination algorithms. Note that the recent work studying the nature of human mobility has proved that suitable movement models can sufficiently present the behavior of human mobility [33]. The model of mobility pattern deployed in our platform is SPMBM model [34], which is a mobility model that integrates temporal and spatial relationships and selects the shortest path for the node randomly walking in the map area.

4.1. Simulation Settings. The data sets to evaluate the location prediction algorithm based on O2MM are obtained from wireless topology discovery (WTD) [35], which are employed in our simulation. The accuracy of the location prediction algorithms based on O1MM and O2MM can be obtained from our previous work [36], which is given in Table 2.

The simulation is based on a realistic campus scenario shown in Figure 2. There are 40 locations and each one installed a WiFi access point, which can cover the campus

```

(1)  $N_i$  senses the contact nodes set  $C\{N_f\}$ ;
(2) for  $\forall N_f \in C\{N_f\}$ 
(3)   update intimacy( $N_i, N_f$ );
(4) end for
    /* the source node deliver the data directly to the destination node if the destination node is in the contact
    nodes set  $C\{N_f\}$  and update the node set  $tabu_d^*$  /
(5) if  $N_d \in C\{N_f\}$ 
(6)    $N_i$  transmits data to  $N_d$ ;
(7)    $tabu_d \leftarrow \{N_i, N_d\}$ ;
(8)   break the Algorithm 5;
(9) else
(10)   $N_s$  sends REQ( $C\{N_f\}, N_d$ ) to the server;
(11)  the server executes LoP_Service( $C\{N_f\}, N_d$ );
(12)   $N_s$  receives SEI;
(13)   $N_s$  sends  $N_d$  to the nodes in  $F\{N_f\}$ ;
(14)  for  $\forall N_f \in F\{N_f\}$ 
(15)    Send intimacy( $N_f, N_d$ ) to  $N_s$ ;
(16)  end for
(17)   $N_s$  calculates  $p_{sf}^d$  according to (21);
(18)  if ( $f \leq COPY$ ) //  $f$  is the number of forwarding nodes
(19)     $F'\{N_f\} = F\{N_f\}$ ;
(20)  else
(21)     $F'\{N_f\} = \{\arg \max_{f \leq COPY} (p_{sf}^d)\}$ 
(22)  end if
(23)   $N_i$  sends data to  $F'\{N_f\}$ ;
(24)   $tabu_d \leftarrow \{N_i, F'\{N_f\}\}$ ;
(25)   $N_i$  delete the data copy in its buffer;
(26) end if

```

ALGORITHM 6: LOPSI_Sec(N_i): the function executes one-hop data dissemination to the forwarding node set.

TABLE 3: Simulation parameters.

	Parameter	Value
Scene features	Simulation time	12 h
	Field area	4500 m * 3400 m
	Scene	NEU Campus
	APs	40
Node features	Mobility model	SPMBM
	Movement speed for cars	2.7–13.9 m/s
	Movement speed for pedestrians	0.5–1.5 m/s
	Transmission rate	250 KB/s
	Maximum transmission range	10 m
	Transmission mode	Broadcast
	Cache size	10 MB/1 G
Message features	Packet size	500 KB–1 MB at random
	Frequency of creating packets	From 25 s to 35 s at random
	Number of copies	8
	TTL	5 hours

area. The mobile nodes can be cars and pedestrians with smart phones. The location prediction server can communicate with mobile nodes via WiFi and only provide location prediction service and no other services. The only way to exchange and obtain data is through the contact of two nodes. If the communication range increases and is out of the range

of APs, the mobile devices can exchange data by the ACO-DAD without using location prediction scheme by different wireless communication techniques including WiFi Direct, Bluetooth, and ZigBee. In order to evaluate the performance of the proposed data dissemination algorithms, we conduct a series of experiments under the parameters in Table 3. The

First In First Out is applied on buffer management. In order to avoid the heavy traffic load and cache load, each data is set a TTL, and the maximum copies of each data is a fixed constant in the system.

With the above settings, the three data dissemination algorithms proposed in our work LOPDAD, ACODAD, and LOPSI are evaluated and compared with well-known opportunistic routing protocols: PRoPHET and Spray and Wait.

4.2. Evaluation Metrics. Four metrics are used to evaluate the performance requirements of the aforementioned data dissemination algorithms: average hops, delivery ratio, average latency, and transmission cost.

Average Hop H . This hop-count metric is to assess the delivery cost in time and in cache. N denotes the total number of forwarding nodes of every transmission of data including both successful and failure delivery. Y is the total number of created unique data messages. H is given by

$$H = \frac{N}{Y}. \quad (22)$$

Delivery Ratio R . This metric is to evaluate the effectiveness and utility of the algorithm. S is the total number of successfully delivered unique data messages. R is given by

$$R = \frac{S}{Y}. \quad (23)$$

Average Latency L [35]. The average latency of a unique message is calculated by the following equation:

$$L = \frac{1}{Y} \sum_{i=1}^Y (T_{d_i} - T_{s_i}), \quad (24)$$

where T_{s_i} is the moment that a unique data message i is originated, and T_{d_i} is the time when the first replicate of unique message i arrives at the destination. The replicate is a copy of a unique message. The number of replicates depends on the methodology of the routing algorithm, single or multiple copies [35].

Transmission Cost C . It is a key metric to characterize the resource consumption and evaluate the data dissemination algorithms in OCNs. The total number of copies of unique data messages to deliver in the system, denoted by Q , is divided by the number of copies of successfully delivered messages, denoted by P :

$$C = \frac{Q}{P}. \quad (25)$$

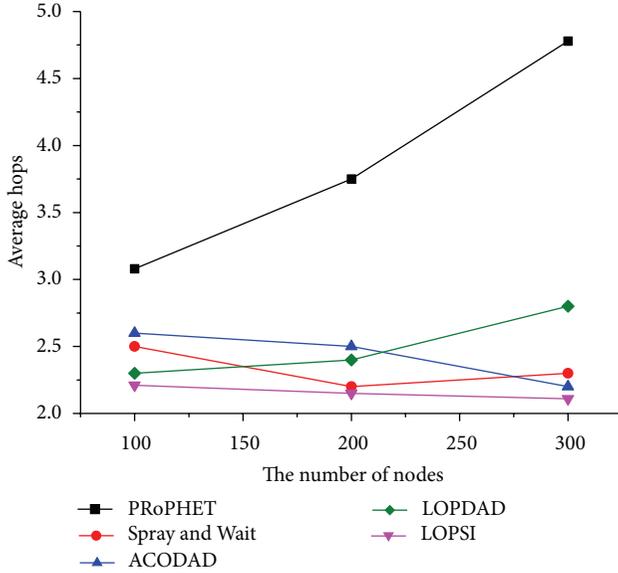
4.3. Influence on Performance with the Variation of Total Number of Nodes

Average Hops. As shown in Figure 3(a), with the increasing of numbers of nodes, due to the hop limitation of Spray and Wait, the average hops are relatively small; PRoPHET has no restrictions in this respect; the frequency of nodes encountering increases, resulting in an increase in the average hops. Our algorithms are proposed to consider the management of cache space and limit the maximum number of copies of the nodes in the network, which constrains the hops of data messages transmission. LOPDAD and LOPSI especially set a time threshold on executing location prediction, which guaranteed the data message delivery to the destination node within the time threshold. The location state is discrete by the time slices, so threshold of time slices corresponds to the number of hops. Therefore, the average hops of our algorithms are relatively small.

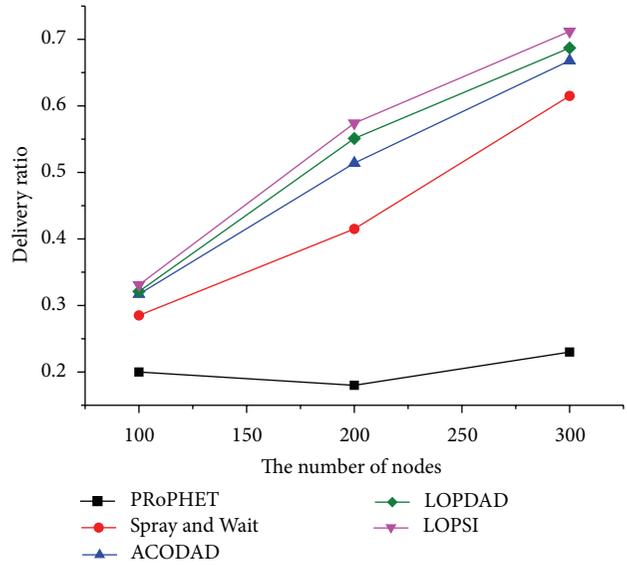
Delivery Ratio. As shown in Figure 3(b), whether for Spray and Wait, PRoPHET, and our data distribution algorithms, the delivery ratio significantly increases with the increase of the nodes. LOPDAD, ACODAD, LOPSI, and Spray and Wait have constrained the number of copies of the data messages transmitted in the network. Even if the number of nodes and the amount of data messages increases, the storage space and the network overhead maintain a good status, avoiding data transmission failure by the heavy load of cache and network resources depletion. Therefore, delivery ratio is better than the PRoPHET transmission. LOPDAD selects forwarding nodes which are most likely to complete the task based on location prediction. ACODAD selects the best forwarding nodes by high intimacy. LOPSI tends to make more “assertive” options to select the forwarding nodes based on the two aforementioned factors, so the delivery ratio has been significantly improved.

Average Latency. As shown in Figure 3(c), the average latency is reduced with the increase of the nodes, indicating that our data distribution algorithms are assertive to select the forwarding nodes which are more likely to contact with the destination node. The data transfer of LOPDAD occurs at the location that can connect with AP points, not at any other encounter places, so the average delay is longer than that of ACODAD and LOPSI. The forwarding nodes selection mechanism of LOPSI is better than that of ACODAD, which has less transmission operation but high delivery ratio, so the average latency is lower than that of ACODAD.

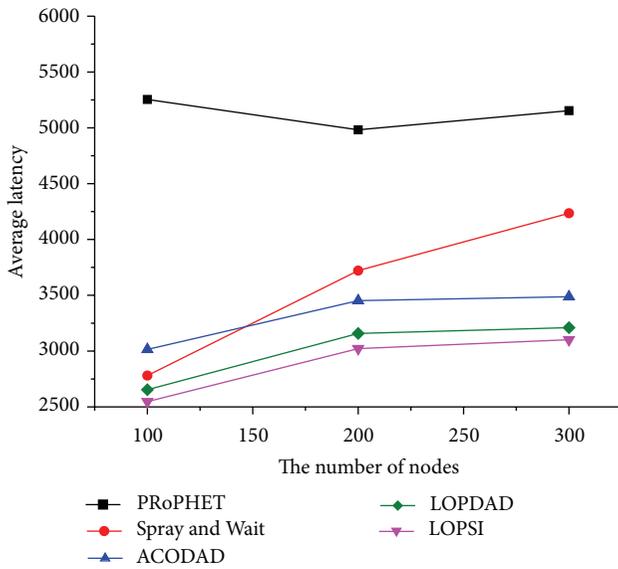
Transmission Cost. Figure 3(d) shows the transmission cost of our proposed algorithm and some existing algorithms. LOPSI has the lowest transmission cost since it only transfers messages to the nodes with the highest forwarding probability to the destination and the quantity of copies of the message is a constant which equals the hops estimated by the location prediction algorithm. With the increase of number of nodes, the opportunity of forwarding messages to the potential nodes increases, which leads to increase of the



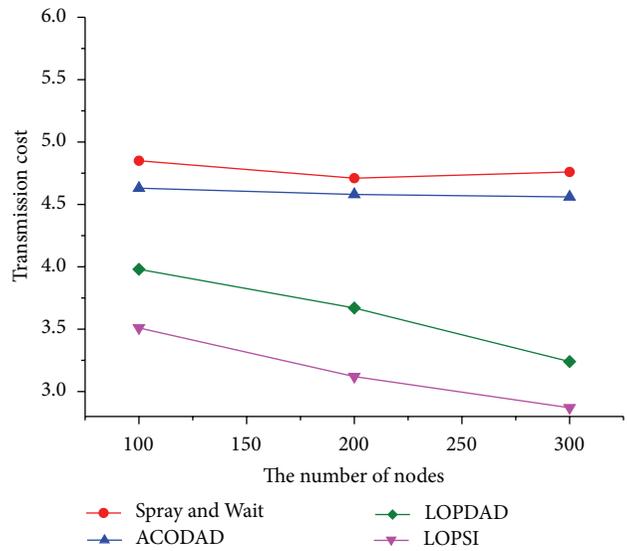
(a)



(b)



(c)



(d)

FIGURE 3: (a) Average hops variation with the increase of the number of nodes. (b) Delivery ratio variations with the increase of the number of nodes. (c) Average latency variations with the increase of the number of nodes. (d) Transmission cost variations with the increase of the number of nodes.

delivery ratio, and also the average hops decrease (as shown in Figure 3(a)). PRoPHET has the highest transmission cost since it has no consideration in the cache management. The transmission cost is at the value of more than 1100, much more than the transmission cost of the other four algorithms. Thus the curve of transmission cost of PRoPHET cannot be drawn in the scale of the graph. The transmission costs of the other three algorithms are lower than that of PRoPHET since they constrained the quantity of copies, but not better than LOPSI because the selection schemes of potential forwarding nodes are no better than that of LOPSI.

4.4. Influence on Performance with the Variation of TTL. In this scenario, in order to illustrate the influence on performance by the variation of TTL, the simulation parameter of the number of nodes is set to be 200.

Average Hops. As shown in Figure 4(a), with the increase of TTL, the data messages live long in the network, which will increase the load of cache and the network. However, LOPDAD and LOPSI have little change in the average hops since the most data messages are successfully delivered to

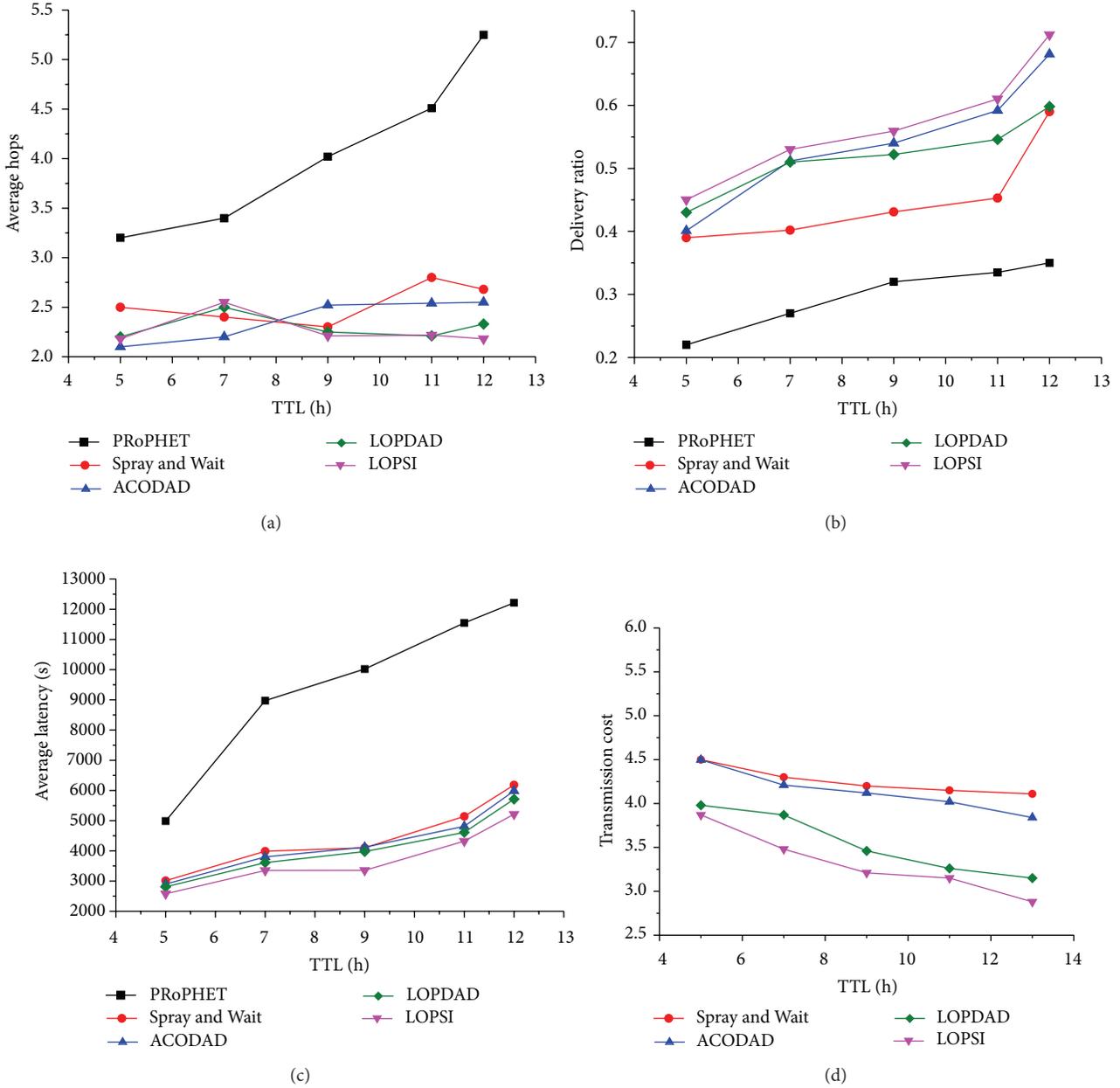


FIGURE 4: (a) Average hops variations with the increase of TTL. (b) Delivery ratio variations with the increase of TTL. (c) Average latency variations with the increase of TTL. (d) Transmission cost variations with the increase of TTL.

the destination in the time threshold according to the accuracy of O2MM being 80%. During the TTL, if the data dissemination is failure in the first time threshold period, then start the second time threshold to deliver the data message; the average hops may be doubled; however, the probability of that condition is less than 20%. The average hops of ACODAD are more than LOPDAD and LOPIS, as it has no consideration of location where the destination node will be, only by random encounter of two nodes. With the increase of the network load, the average hops will increase.

Delivery Ratio. As shown in Figure 4(b), with the increase of TTL, the delivery ratio of LOPDAD changes little. According

to the characteristic of Markov chain prediction, the future status of long-time prediction tends to be stabilized, which means that the prediction accuracy is reduced. If the time threshold is set too big, then delivery ratio will decrease. The delivery ratio of ACODAD and LOPIS increase since the copy of data message in the network is not changed but the opportunistic of encounter is increased.

Average Latency. As shown in Figure 4(c), with the increase of TTL, the average latency of those five algorithms will increase since the network load is heavy and the total number of data messages increases, which makes the buffer overcrowded. Since the algorithms except PRoPHET constrain the copies of messages to avoid the traffic loads, the latency is lower

than that of PROPHET which transfers the message to any potential nodes without copies constraint.

Transmission Cost. As shown in Figure 4(d), with the increase of TTL, the transmission cost of LOPSI and LOPDAD decreases since the delivery ratio is higher (as shown in Figure 4(b)) and the number of copies of a unique message is slightly changed. It results in that LOPSI and LOPDAD estimate the optimal path and then determine the number of copies of a unique message. Spray and Wait and ACODAD constrain the number of copies, so the variation of transmission cost is slight. As regards PROPHET, since it does not constrain the copies of a unique message, the transmission cost is much higher than the transmission cost of the other four algorithms and at a more than thousand value. Thus the curve of transmission cost of PROPHET cannot be drawn in the scale of the graph.

5. Conclusion and Future Work

In this paper we consider the efficient data dissemination mechanism in opportunistic cognitive networks. We propose a swarm intelligence heuristic data dissemination algorithm based on location prediction. The algorithm can select the efficient forwarding nodes with the maximum probability to encounter the destination node at the location where they most likely to encounter and a high value of intimacy with the destination node. Moreover the algorithm considers the cache management and has good performances in the transmission cost and delivery ratio and simultaneously decreases the average hops and delivery delay. The algorithm has good scalability, which can consider credibility, incentives, energy control, and buffer management mechanism in the future work. Furthermore, the swarm intelligence scheme can also be used in group construction of the mobile nodes, which can promote the application performance in opportunistic cognitive networks.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

The authors would like to thank all the editors of this paper. They read the manuscript very carefully and provided valuable feedbacks, which are helpful to improve the quality of the draft.

References

- [1] Q. H. Mahmoud, *Cognitive Networks: Towards Self-Aware Networks*, John Wiley & Sons, London, UK, 2007.
- [2] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp. 134–141, 2006.
- [3] R. Yu, P. Wang, and Z. Zhao, "NDI: node-dependence-based dynamic gaming Incentive algorithm in opportunistic networks," in *Proceedings of the 23rd International Conference on Computer Communications and Networks (ICCCN '14)*, pp. 581–588, Shanghai, China, 2014.
- [4] R. Yu, R. Liu, X. Wang, and J. Cao, "Improving data quality with an accumulated reputation model in participatory sensing systems," *Sensors*, vol. 3, pp. 5573–5594, 2014.
- [5] B. Hull, V. Bychkovsky, K. Chen et al., "CarTel: a distributed mobile sensor computing system," in *Proceedings of the 4th ACM International Conference on Embedded Networked Sensor Systems*, pp. 125–138, November 2006.
- [6] M. Mun, S. Reddy, K. Shilton et al., "PEIR, the personal environmental impact report, as a platform for participatory sensing systems research," in *Proceedings of the 7th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys '09)*, pp. 55–68, June 2009.
- [7] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," Tech. Rep., Department of Computer Science, Duke University, Durham, NC, USA, 2000.
- [8] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," in *Proceeding of the Conference on Computer Communications (ACM SIGCOMM '04)*, pp. 145–158, New York, NY, USA, September 2004.
- [9] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Single-copy routing in intermittently connected mobile networks," in *Proceedings of the 1st Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON '04)*, pp. 235–244, October 2004.
- [10] A. Lindgren and A. Droia, "Probabilistic routing protocol for intermittently connected networks," Internet Draft draft-lindgren-dtnrg-prophet-02, Work in Progress, 2006.
- [11] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant manets," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '07)*, pp. 32–40, ACM, New York, NY, USA, 2007.
- [12] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE rap: social-based forwarding in delay tolerant networks," in *Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08)*, pp. 241–250, May 2008.
- [13] J. Á. B. Link, N. Viol, A. Goliath, and K. Wehrle, "SimBetaAge: utilizing temporal changes in social networks for pocket switched networks," in *Proceedings of the 1st ACM Workshop on User-Provided Networking: Challenges and Opportunities (UNET '09)*, pp. 13–18, ACM, New York, NY, USA, December 2009.
- [14] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of the ACM SIGCOMM Workshops: Conference on Computer Communications*, pp. 252–259, August 2005.
- [15] J. Leguay, T. Friedman, and V. Conan, "DTN routing in a mobility pattern space," in *Proceedings of the ACM SIGCOMM Workshops: Conference on Computer Communications*, pp. 276–283, August 2005.
- [16] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research C*, vol. 43, pp. 233–248, 2014.
- [17] B. Yao, P. Hu, M. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem," *Simulation*, vol. 89, no. 6, pp. 762–770, 2013.

- [18] B.-Z. Yao, C.-Y. Yang, and J.-B. Yao, "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.
- [19] B. Yao, P. Hu, M. Zhang, and X. Tian, "Improved ant colony optimization for seafood product delivery routing problem," *PROMET—Traffic&Transportation*, vol. 26, no. 1, pp. 1–10, 2014.
- [20] B. Yao, C. Yang, J. Hu, J. Yao, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6-7, pp. 2127–2131, 2011.
- [21] B. Z. Yao, J. B. Yao, and M. H. Zhang, "Improved support vector machine regression in multi-step-ahead prediction for rock displacement surrounding a tunnel," *Scientia Iranica*. In press.
- [22] B. Yu, Z. Z. Yang, and K. Chen, "Hybrid model for prediction of bus arrival times at next station," *Journal of Advanced Transportation*, vol. 44, no. 3, pp. 193–204, 2010.
- [23] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus arrival time prediction using support vector machines," *Journal of Intelligent Transportation Systems*, vol. 10, no. 4, pp. 151–158, 2006.
- [24] B. Yu, Z. Z. Yang, and B. Z. Yao, "A hybrid algorithm for vehicle routing problem with time windows," *Expert Systems with Applications*, vol. 38, no. 1, pp. 435–441, 2011.
- [25] B. Yu, H. B. Zhu, W. J. Cai, N. Ma, and B. Z. Yao, "Two-phase optimization approach to transit Hub location—the case of Dalian," *Journal of Transport Geography*, vol. 33, pp. 62–71, 2013.
- [26] B. Yu, Z. Yang, and J. Yao, "Genetic algorithm for bus frequency optimization," *Journal of Transportation Engineering*, vol. 136, no. 6, pp. 576–583, 2010.
- [27] M. Farooq, *Bee-Inspired Protocol Engineering: From Nature to Networks*, Springer, New York, NY, USA, 2009.
- [28] A. Zengin, H. Sarjoughian, and H. Ekiz, "Discrete event modeling of swarm intelligence based routing in network systems," *Information Sciences*, vol. 222, pp. 81–98, 2013.
- [29] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, 1999.
- [30] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 26, no. 1, pp. 29–41, 1996.
- [31] R. Schoonderwoerd, *Collective intelligence for network control [M.S. thesis]*, Faculty of Technical Informatics, Delft University of Technology, 1996.
- [32] M. Dorigo and T. Stutzle, "The ant colony optimization metaheuristic: algorithms, applications, and advances," in *Handbook of Metaheuristics*, pp. 251–285, Springer, 2002.
- [33] V. D. Le, H. Scholten, P. J. M. Havinga, and H. Ngo, "Location-based data dissemination with human mobility using online density estimation," in *Proceedings of the 11th Annual IEEE Consumer Communications & Networking Conference*, pp. 747–754, Las Vegas, Nev, USA, November 2014.
- [34] A. Ahmed and K. Abu Bakar, "A simulation based study of well known routing protocols for delay tolerant network," *World Applied Sciences Journal*, vol. 28, no. 3, pp. 353–360, 2013.
- [35] M. McNett and G. M. Voelker, UCSD Wireless Topology Discovery Project [EB/OL], 2013, <http://www.sysnet.ucsd.edu/wtd/wtd.html>.
- [36] J. Li, X. Xing, R. Yu, X. Wang, and Y. Zhou, "Social relationship-based mobile node location prediction algorithm in opportunistic cognitive networks," *WIT Transactions on Information and Communication Technologies*, vol. 59, pp. 113–119, 2014.

Research Article

Quantum Behaved Particle Swarm Optimization Algorithm Based on Artificial Fish Swarm

Dong Yumin¹ and Zhao Li^{2,3}

¹ Network Center, Qingdao Technological University, Qingdao 266033, China

² Qingdao Hiser Medical Group, Qingdao 266033, China

³ Qingdao Technological University, Qingdao 266033, China

Correspondence should be addressed to Dong Yumin; dym1188@163.com

Received 2 May 2014; Revised 29 July 2014; Accepted 13 August 2014; Published 28 August 2014

Academic Editor: Fang Zong

Copyright © 2014 D. Yumin and Z. Li. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Quantum behaved particle swarm algorithm is a new intelligent optimization algorithm; the algorithm has less parameters and is easily implemented. In view of the existing quantum behaved particle swarm optimization algorithm for the premature convergence problem, put forward a quantum particle swarm optimization algorithm based on artificial fish swarm. The new algorithm based on quantum behaved particle swarm algorithm, introducing the swarm and following activities, meanwhile using the adaptive parameters, to avoid it falling into local extremum of population. The experimental results show the improved algorithm to improve the optimization ability of the algorithm.

1. Introduction

Quantum behaved particle swarm algorithm is first proposed by Sun et al. [1] in 2004. Quantum behaved particle swarm optimization algorithm introduces quantum computing into the particle swarm algorithm, starting from the mechanical point of view that the particle in the space has quantum behavior. The algorithm overcomes the disadvantages while preserving the advantages of particle swarm algorithm, which can effectively improve the performance of optimization algorithms.

Research on the quantum particle swarm optimization mainly focuses on the following three aspects: The first one is proof theoretic research, the second one is to improve the contraction expansion factor, and the third one is combined with other algorithms. In 2005, Moore and Venayagamoorthy [2] proposed quantum particle swarm algorithm for the combinational logic circuit. In 2006, Mikki and Kishk [3] proposed a quantum mechanical particle swarm algorithm based on electromagnetism and is used to optimize the electromagnetic aspects.

Quantum particle swarm optimization (QPSO) is in the field of medical image watermarking for copyright protection

and authentication [4]. The trade-off between the imperceptibility and robustness is one of most serious challenges in digital watermarking system. Image watermarking can be considered as an optimization problem by utilizing human visual system characteristics and QPSO algorithm in adaptive quantization index modulation and singular value decomposition in conjunction with discrete wavelet transform and discrete cosine transform. In the literature [5], a modified and efficient version of the QPSO combined with chaotic sequences (CQPSO) is proposed and evaluated.

An artificial fish swarm algorithm based on a filter methodology for trial solutions acceptance is analyzed for general constrained global optimization problems [6]. The preliminary numerical experiments with a well-known benchmark set of engineering design problems show the effectiveness of the proposed method.

In the literature [7], an improved dynamic clustering algorithm was presented, which combines the quantum particle swarm algorithm with k -means algorithm by improving the encoding of quantum particles and the introduction of new distance metric rules. The algorithm has a quantum behaved particle swarm global search capability. In order to accelerate the convergence speed, the k -means algorithm

is used to optimize every particle. Through the adjustment of the value of the fitness function, our algorithm can search for the optimal clustering number of clusters, so the number of clusters and centers is not subject to subjective factors. Extensive experiments verified the effectiveness of the algorithm.

Based on quantum evolutionary algorithm and particle swarm optimization, a quantum particle swarm evolutionary algorithm is proposed [8]. In this algorithm, quantum angle is used to represent the qubit, a new method learning from the idea of particle swarm algorithm which is presented to determine rotation angle. The gate is taken to prevent premature convergence. In the literature [9], analytical optimization techniques suffer from slow convergence in complex solution space. Heuristics-based swarm intelligence is an efficient alternative to analytical optimization techniques. The particle swarm optimization approach is utilized for better and efficient nanodevice modeling.

Artificial fish swarm algorithm (AFSA) is a kind of swarm intelligence algorithms which is usually employed in optimization problems [10]. There are many parameters to adjust in AFSA like visual and step. Evaluations of the proposed methods were performed on eight well-known benchmark functions in comparison with standard AFSA and particle swarm optimization (PSO). The overall results show that proposed algorithm can be effective surprisingly.

The paper based on the quantum behaved swarm algorithm, combined with the artificial fish swarm algorithm, proposed the new improved quantum particle swarm algorithm. In order to avoid the individual into local extremum, the new algorithm uses the swarm and following behavior, meanwhile it combines with the adaptive parameter adjustment.

2. Quantum Particle Swarm Algorithm

2.1. The Principle of the Algorithm. According to the principle of quantum mechanics, quantum particle swarm algorithm is proposed by Sun et al. With the help of DELTA potential well, the particle swarm optimization algorithm is applied to the quantum space. The quantum space particle used wave function to describe

$$|\Psi|^2 dx dy dz = Q dx dy dz. \quad (1)$$

Among them, $|\Psi|^2$ is the square of the module of wave function, representing the probability density of particles in a position to appear. Q is the probability density function and satisfies the normalization condition:

$$\int_{-\infty}^{+\infty} |\Psi|^2 dx dy dz = \int_{-\infty}^{+\infty} Q dx dy dz = 1. \quad (2)$$

Assume that the D dimension of (on behalf of the dimension of the variables associated with problem) quantum space has a population, which consists of n particles. Location of the i th particle is $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, and the particle through the history of the best location is $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$; after all the particles of the best historical position is $P_g = (p_{g1}, p_{g2}, \dots, p_{gD})$.

In quantum space, positions of particles after the particles get through stochastic simulation of Monte Carlo measurement:

$$x_{id} = p_{id} \pm \frac{L}{2} \ln \left(\frac{1}{u} \right) \quad (i = 1, 2, \dots, n) \quad (d = 1, 2, \dots, D). \quad (3)$$

Among them, u is the random number range of $[0, 1]$. L is obtained by the particle's current position and historical best position is $L = 2 \cdot \beta \cdot |p_{id} - x_{id}|$. Thus, get the update formula of quantum particle swarm optimization:

$$x_{id}(t+1) = p_{id} \pm \beta \cdot \left| p_{id} - x_{id}(t) \cdot \ln \left(\frac{1}{u} \right) \right|. \quad (4)$$

Among them, t is the iteration number of algorithm. β is the contraction expansion factor and is the only parameter of quantum particle swarm algorithm.

In order to avoid the premature convergence, Sun et al. improved quantum particle swarm algorithm, introducing $mbest$ in the algorithm; that is,

$$\begin{aligned} mbest(t) &= \frac{1}{n} \sum_{i=1}^n p_i(t) \\ &= \left[\frac{1}{n} \sum_{i=1}^n p_{i1}(t), \frac{1}{n} \sum_{i=1}^n p_{i2}(t), \dots, \frac{1}{n} \sum_{i=1}^n p_{iD}(t) \right], \end{aligned} \quad (5)$$

where p_i is the best position of i th particles and n is the number of particles. " $mbest$ " find the average best location of n particles and solve problems based on the dimension of the variable.

After the introduction of $mbest$, the individual update formula is

$$\begin{aligned} L &= 2 \cdot \beta \cdot |mbest - x_{id}|, \\ x_{id} &= p_{id} \pm \beta \cdot |mbest_d - x_{id}| \cdot \ln \left(\frac{1}{u} \right). \end{aligned} \quad (6)$$

Therefore, quantum particle swarm optimization particle updating formula can be described as

$$\begin{aligned} p_{id} &= \varphi \cdot p_{id} + (1 - \varphi) \cdot p_{gd}, \\ mbest(t) &= \frac{1}{n} \sum_{i=1}^n p_i(t) \\ &= \left[\frac{1}{n} \sum_{i=1}^n p_{i1}(t), \frac{1}{n} \sum_{i=1}^n p_{i2}(t), \dots, \frac{1}{n} \sum_{i=1}^n p_{iD}(t) \right], \\ X_i(t+1) &= p_i \pm \beta \cdot |mbest - X_i(t)| \cdot \ln \left(\frac{1}{u} \right). \end{aligned} \quad (7)$$

Among them, φ is the random number in $[0, 1]$; other parameters are same as the ones mentioned above.

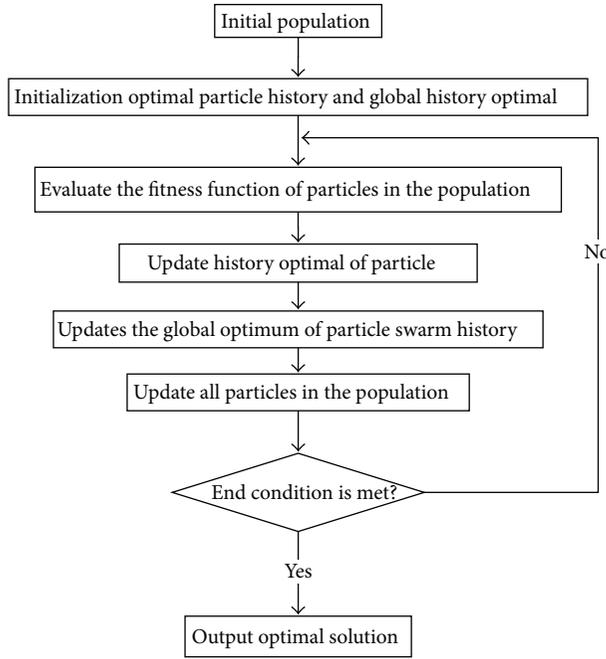


FIGURE 1: The flowchart of quantum particle swarm algorithm.

2.2. *The Algorithm Flow.* Quantum particle swarm algorithm flow chart in Figure 1 is specifically described as follows.

Step 1. Initialize algorithm parameters (population size n , particle dimension d , the maximum number of iterations MAXGEN), population initialization, initialization particles history p_i , and global history optimal value p_g .

Step 2. Evaluate individual fitness value.

Step 3. Update the optimal population in history. The particle's fitness is better than the particle history itself, with the current value of the replacement; otherwise, the history optimal particles remain unchanged.

Step 4. Update the history global optimal particle in a population, the best fitness value of all the particles in the population.

Step 5. Update particles by using quantum behaved particle swarm optimization algorithm formula, all the particles in space.

Step 6. If the algorithm reaches the maximum number of iterations, then output the optimal solution, and the algorithm terminates; otherwise, continue to implement the Step 2.

3. An Improved Quantum Behaved Particle Swarm Algorithm

For quantum particle swarm algorithm, due to the small parameter and the solving speed of algorithm, many scholars

have joined the ranks of the algorithm, but the premature convergence may also exist. In order to optimize the algorithm, put forward a quantum particle swarm optimization algorithm based on artificial fish swarm, abbreviated as QAFSP.

3.1. Improvement Strategies

3.1.1. *Swarm and Follow.* In the improved QAFSP algorithm, for the individual X_i updates, cluster operates to the optimal position of the history of P_i particles. Based on P_i and searching other individuals within the perceived distance together constitute a search region. Calculate and record the center position $X_{p\text{-cen}}$ and value $f(X_{p\text{-cen}})$ to the small area. If the $f(X_{p\text{-cen}})$ is better and the center around the position is not too crowded, so the individual X_i moves to the $X_{p\text{-cen}}$ direction, and the moving formula is as follows:

$$X_{\text{new}i} = X_i + \text{rand}() \cdot \text{step} \cdot \frac{X_{p\text{-cen}} - X_i}{\|X_{p\text{-cen}} - X_i\|}. \quad (8)$$

If the mobile requirement is not met, the individual X_i performing quantum particle swarm particle updating formula gets the new state of particles.

The introduction of polygroup activities which can help particle has been or is about to fall into the local extremum to escape, moving toward the optimal direction, to improve the quantum particle swarm algorithm local convergence. Similarly, the rear-end activities are also conducted for the individual X_i and the current particle best position P_i to be carried out. Take the minimum value problem as an example, rear-end activity is based on the P_i , searches for the minimum value of individual $X_{p\text{-min}}$ among its sensing range. The all individuals within the perceived distance of $X_{p\text{-min}}$ together form a small region. If the fitness value of $X_{p\text{-min}}$ is better, and the small region is not too crowded, then the individual X_i moves towards the direction of $X_{p\text{-min}}$ position:

$$X_{\text{new}i} = X_i + \text{rand}() \cdot \text{step} \cdot \frac{X_{\text{min}} - X_i}{\|X_{\text{min}} - X_i\|}. \quad (9)$$

If the mobile requirement is not met, then the particles X_i perform the updating formula of the quantum particle swarm algorithm and get the new state of particles.

Introduction of rear-end activities and enabling space particle to move toward better accelerate the speed of the individuals to be close to better solutions. In addition, this can help the individual to jump out of local minimum, to prevent it falling into local extreme and to stop the search.

3.1.2. *Adaptive Adjustment.* The shrinkage factor β is the only parameter of quantum particle swarm algorithm; how to choose can affect the overall performance of the proposed algorithm. If the contraction factor is too large, the algorithm convergence has long search time and too slow time; if it is too small, this can make the algorithm into a local optimal solution.

In order to avoid the defects of shrinkage factor fixed, use adaptive shrinkage factor to improve QAFSP algorithm. The

shrinkage factor is defined as a function associated with the evolution algebra:

$$\beta = (1 - 0.5) \cdot \frac{\text{MAXGEN} - t}{\text{MAXGEN}} + 0.5. \quad (10)$$

Among them, MAXGEN is the maximum number of iterative algorithms; t is the evolution algebra.

As can be seen, the shrinkage factor in the initial stage of algorithm is greatly easy to large range for group search. Accompanied by optimization algorithm, the shrinkage factor decreases gradually, the individual searching scope reduces. And it starts to fine search among the local area.

Another parameter to be automatically adjusted is the moving step in the clusters and rear end. Similar to the shrinkage factor, step size has a direct impact on the performance of the proposed algorithm. The individuals with larger step size expand those searching scope, and the individuals with smaller step size start to fine search among the local area.

In the improved quantum particle swarm algorithm QAFSP, use the adaptive step size adjustment. Based on fitness value of $X_{p\text{-cen}}$ in the cluster or $X_{p\text{-min}}$ in rear end, the ratio of X_i and individual fitness between, automatically adjust the step size, in particular to:

$$\left| 1 - \frac{f(X_{p\text{-cen}})}{f(X_i)} \right| \cdot \text{step} \quad (11)$$

or

$$\left| 1 - \frac{f(X_{p\text{-min}})}{f(X_i)} \right| \cdot \text{step}.$$

Therefore, when the individual X_i is closer to $X_{p\text{-cen}}$ or $X_{p\text{-min}}$, step of smaller values is more suitable for fine search in a small range. On the contrary, the adaptive step of larger size is suitable for moving in large scope. Adaptive step size selection can balance the local and global search ability, improving the ability of the algorithm.

3.2. Quantum Behaved Particle Swarm Optimization Algorithm Based on Artificial Fish Swarm. The proposed quantum particle swarm optimization algorithm based on artificial fish swarm, the basic process shown in Figure 2, can be described as follows.

Step 1. Initialize algorithm parameters (population size n , the number of iterations MAXGEN, perceived distance, the maximum number of temptation, the maximum step size, and crowding factor) and population initialization.

Step 2. Evaluate individual fitness value of the population. Do records particle history optimal and populations of the globally best.

Step 3. Implement quantum particle swarm optimization particle updating formula; the particle X_i is updated.

Step 4. Update the optimal particle itself and update the globally best population.

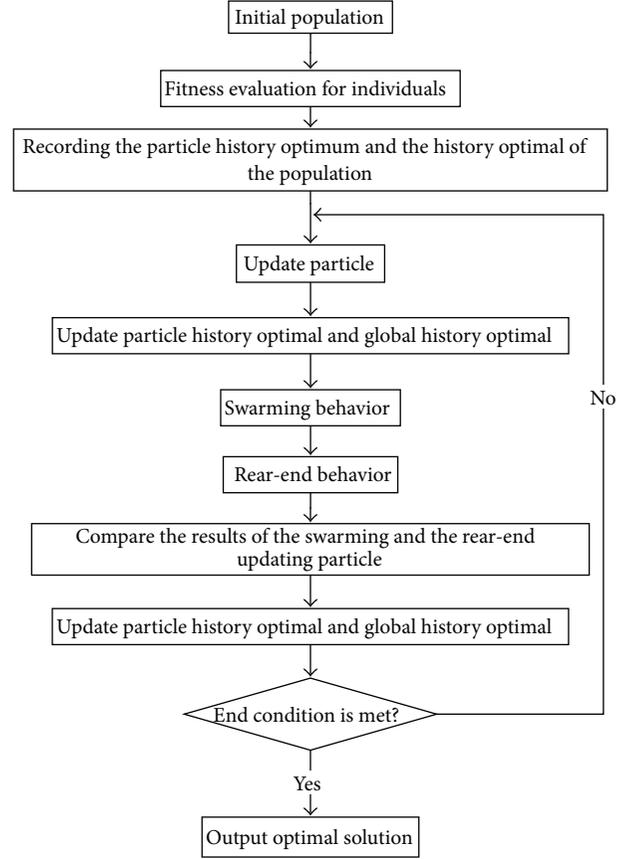


FIGURE 2: Flowchart of the quantum particle swarm algorithm based on artificial fish.

Step 5. For polygroup activities, according to the history of optimal P_i particles X_i , particles are perceived within the scope of the center P_i . If it meets the mobile condition, P_i move to the center position direction; if not satisfied, then perform particle update.

Step 6. For rear-end activity, according to the historical optimum P_i particle X_i , get the minimum fitness value of individual particles in the P_i perception. If it meets the mobile condition, P_i move to the minimum fitness position direction of individual; otherwise, perform particle update.

Step 7. To compare Step 6 to Step 7, newly produced particles, choose those better fitness values, updated X_i .

Step 8. Update the particle's own history and the globally best optimal of population.

Step 9. If the maximum number of iterations is reached, then output the optimal solution, and the algorithm terminates; otherwise, continue to implement Step 3.

Compared with quantum particle swarm algorithm, the improved algorithm adopts swarming and rear-end activities

TABLE I: The parameters of the algorithm.

		Population scale	Maximum number of iterations	Individual test times	Individual perception of distance	Congestion factor	The maximum step size	Code length
f_1	QAFSP	20	50	5	50	1.618	0.1	—
	QPSO	20	50	—	—	—	—	—
	QGA	100	100	—	—	—	—	20
	AFSA	100	100	5	50	11.618	1.25	—
f_2	QAFSP	100	100	5	150	1.618	0.1	—
	QPSO	100	100	—	—	—	—	—
	QGA	150	100	—	—	—	—	20
	AFSA	150	100	5	150	11.618	1.25	—
f_3	QAFSP	10	50	5	100	1.618	0.1	—
	QPSO	10	50	—	—	—	—	—
	QGA	200	100	—	—	—	—	20
	AFSA	200	100	10	100	11.618	1.25	—
f_4	QAFSP	20	50	5	5	1.618	0.1	—
	QPSO	20	50	—	—	—	—	—
	QGA	50	50	—	—	—	—	20
	AFSA	20	50	5	5	11.618	1.25	—
f_5	QAFSP	200	100	5	10	1.618	1.25	—
	QPSO	200	100	—	—	—	—	—
	QGA	200	100	—	—	—	—	20
	AFSA	200	100	20	10	11.618	11.25	—
f_6	QAFSP	150	100	5	200	1.618	0.1	—
	QPSO	200	200	—	—	—	—	—
	QGA	200	100	—	—	—	—	20
	AFSA	200	200	50	200	61.618	21.25	—
f_7	QAFSP	50	100	5	300	1.618	0.1	—
	QPSO	200	100	—	—	—	—	—
	QGA	200	100	—	—	—	—	5
	AFSA	200	100	20	300	61.618	21.25	—
f_8	QAFSP	100	100	5	100	1.618	0.1	—
	QPSO	200	100	—	—	—	—	—
	QGA	200	100	—	—	—	—	5
	AFSA	200	100	20	100	11.618	11.25	—

and basis on the best position of particle. In addition, the swarming and rear end use adaptive step length adjusting operation and dynamically change the length of the particles moving step; in update formula of the particles, adaptive shrinkage factor is also used, according the size of iterative search to automatic adjust.

4. Function Optimization and Simulation Experiment

4.1. Test Function. The improved algorithm is applied to a test function for function optimization. Use the following test functions:

(1)

$$f_1(x) = x^6 - 15x^4 + 27x^2 + 250 \quad x \in [-100, 100] \quad (12)$$

(2)

$$f_2(x, y) = \frac{\sin^2 \sqrt{x^2 + y^2} - 0.5}{[1 + 0.001(x^2 + y^2)]^2} - 0.5 \quad (13)$$

$$x, y \in [-100, 100]$$

(3)

$$f_3(x, y) = \left(4 - 2.1x^2 + \frac{x^4}{3}\right)x^2 + xy + (-4 + 4y^2)y^2 \quad x, y \in [-100, 100] \quad (14)$$

TABLE 2: The test function optimization results.

		The optimal solution	Average value	The average operation time (seconds)	The number of successful	The success rate
f_1	QAFSP	7	7	0.21	50	1
	QPSO	7	7.0004	0.03	45	0.9
	QGA	7	7.0009	2.07	40	0.8
	AFSA	7	7.0019	3.54	37	0.74
f_2	QAFSP	-1	-0.99902	4.16	45	0.9
	QPSO	-1	-0.99756	0.31	35	0.7
	QGA	-1	-0.99137	5.59	19	0.38
	AFSA	-1	-0.95641	9.03	0	0
f_3	QAFSP	-1.0316	-1.0316	0.10	50	1
	QPSO	-1.0316	-1.0315	0.02	49	0.98
	QGA	-1.0316	-0.8293	7.58	14	0.28
	AFSA	-1.0316	20.1389	11.83	12	0.24
f_4	QAFSP	-6.5511	-6.5511	0.22	50	1
	QPSO	-6.5511	-6.5457	0.03	49	0.98
	QGA	-6.5511	-6.4807	0.95	8	0.16
	AFSA	-6.5511	-6.5502	0.23	34	0.68
f_5	QAFSP	$1.8838e - 10$	$1.4904e - 5$	13.26	50	1
	QPSO	0.00012216	0.13664	0.64	6	0.12
	QGA	$2.365e - 8$	3.7042	18.55	4	0.08
	AFSA	0.23427	5.9101	16.87	0	0
f_6	QAFSP	$1.8495e - 7$	0.00021545	9.65	49	0.98
	QPSO	$6.4956e - 5$	0.00076334	1.52	42	0.84
	QGA	0.079839	5.7882	37.15	0	0
	AFSA	13.6422	25.9938	39.69	0	0
f_7	QAFSP	$1.1224e - 6$	$9.4379e - 6$	2.05	50	1
	QPSO	0.026063	0.07969	0.75	0	0
	QGA	14464.0999	27368.9906	27.81	0	0
	AFSA	1364.2993	8233.3037	17.91	0	0
f_8	QAFSP	$1.0984e - 5$	$3.4666e - 5$	6.45	50	1
	QPSO	0.034844	0.1822	0.90	0	0
	QGA	3.1575	3.2964	27.93	0	0
	AFSA	9.5439	12.1968	26.76	0	0

(4)

$$f_4(x, y) = 3(1-x)^2 e^{-x^2-(y+1)^2} - 10\left(\frac{x}{5} - x^3 - y^5\right) e^{-x^2-y^2} - \frac{1}{3} e^{-(x+1)^2-y^2} \quad (15)$$

$$x, y \in [-3, 3]$$

(5)

$$f_5(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10) \quad (16)$$

$$x_i \in [-5.12, 5.12]$$

(6)

$$f_6(x) = \sum_{i=1}^{n-1} (x_i^2 + x_{i+1}^2)^{0.25} \left[\sin^2 \left(50(x_i^2 + x_{i+1}^2)^{0.1} \right) + 1 \right] \quad (17)$$

$$x_i \in [-100, 100]$$

(7)

$$f_7(x) = \sum_{i=1}^n x_i^2 \quad x_i \in [-100, 100] \quad (18)$$

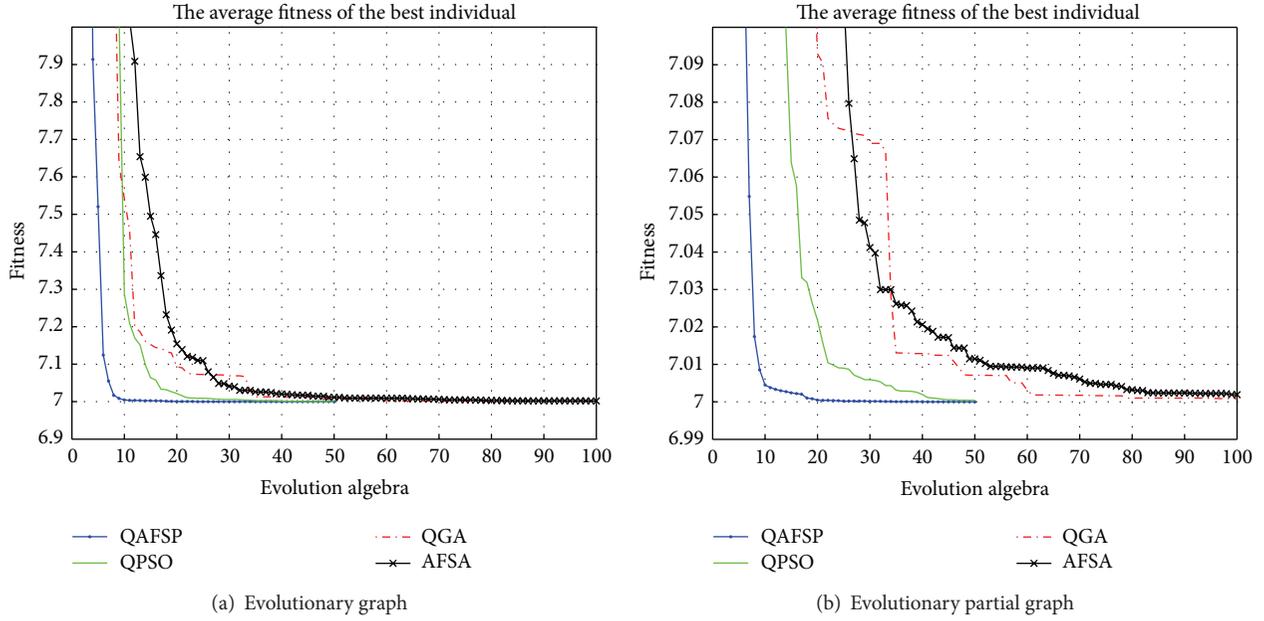


FIGURE 3: Convergence curve test function f_1 .

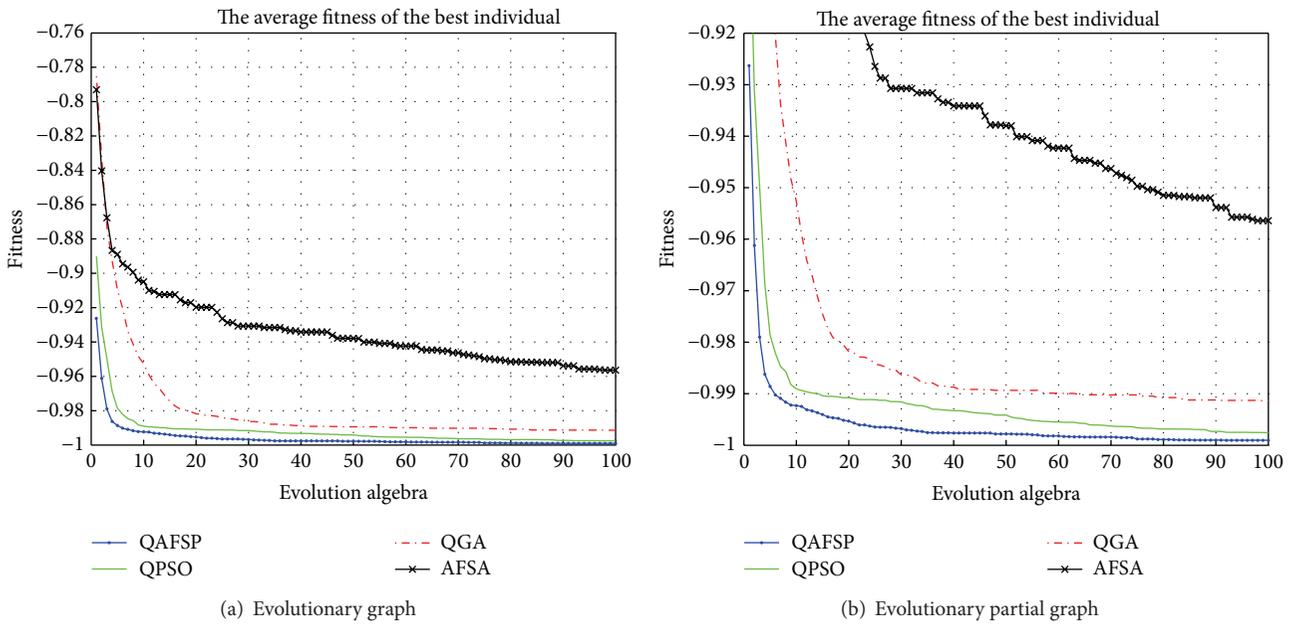


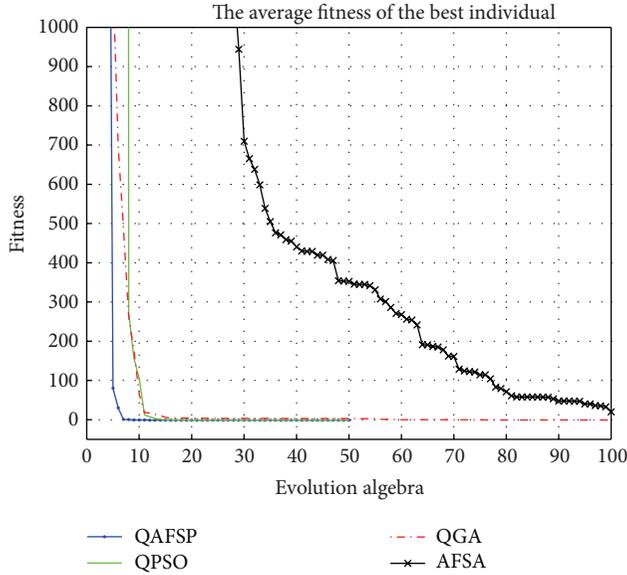
FIGURE 4: Convergence curve test function f_2 .

(8)

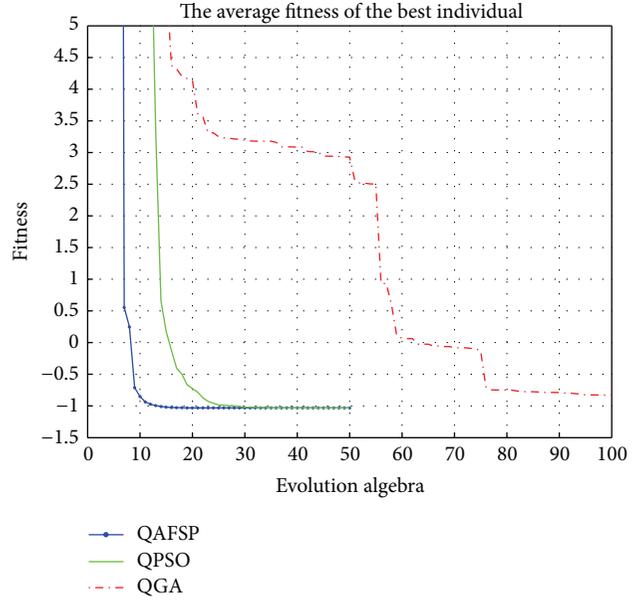
$$\begin{aligned}
 f_8(x) = & -20 \exp\left(-0.2 \sqrt{\frac{1}{30} \sum_{i=1}^n x_i^2}\right) \\
 & - \exp\left(\frac{1}{30} \sum_{i=1}^n \cos(2\pi x_i)\right) + 20 + e \quad (19) \\
 & x_i \in [-32, 32].
 \end{aligned}$$

Minimum theoretical test functions $f_5 \sim f_8$ are 0. The theory of $f_1 \sim f_4$ minimum, respectively, is 7, -1, -1.0316, -6.5511.

4.2. The Experimental Design. The improved quantum particle swarm optimization algorithm based on artificial fish algorithm(QAFSP), quantum particle swarm algorithm (QPSO), quantum genetic algorithm (QGA), and artificial fish swarm algorithm (AFSA), of four kinds of algorithms,



(a) Evolutionary graph



(b) Evolutionary partial graph

FIGURE 5: Convergence curve test function f_3 .

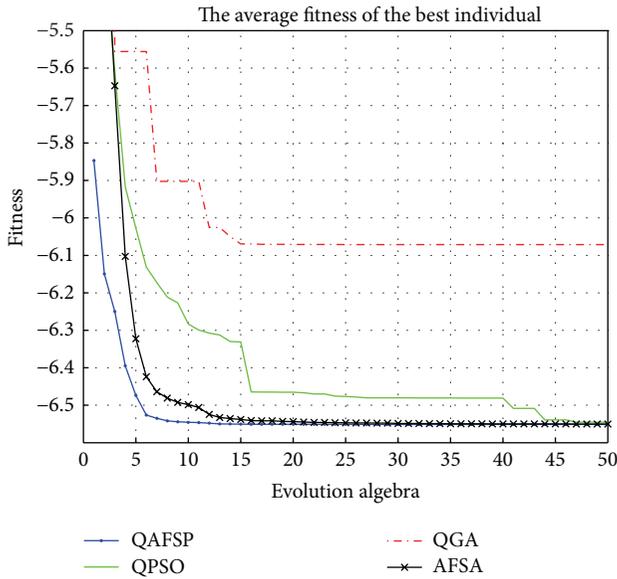


FIGURE 6: Convergence curve test function f_4 .

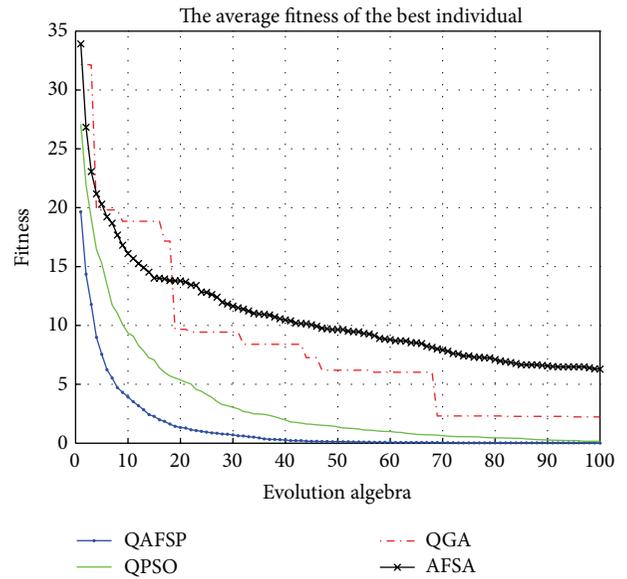


FIGURE 7: Convergence curve test function f_5 .

was applied to the eight minimum test functions. Experiments were carried out in environment of MATLAB R2008b; the computer processor is Intel(R), Core(TM)2, Quad CPU Q9400 @2.66 GHz (4 CPUs), and the RAM is 2048 MB. Each algorithm for each test function is repeated in 50 independent experiments.

Algorithm of QAFSP, QPSO, and AFSA adopts real coding and QGA coding using qubits. For multidimensional functions, f_5 take dimension $n = 5$, f_6 take dimension $n = 10$,

and f_7 and f_8 take dimension $n = 30$. The specific parameter of each algorithm is shown in Table 1.

4.3. *Experimental Results and Analysis.* The accuracy of the function is 10^{-3} , when the difference between optimal value from algorithm and the theoretical minimum value is less than 10^{-3} , denoted by algorithm for success. Define the algorithm success rate, the number of times for successful and

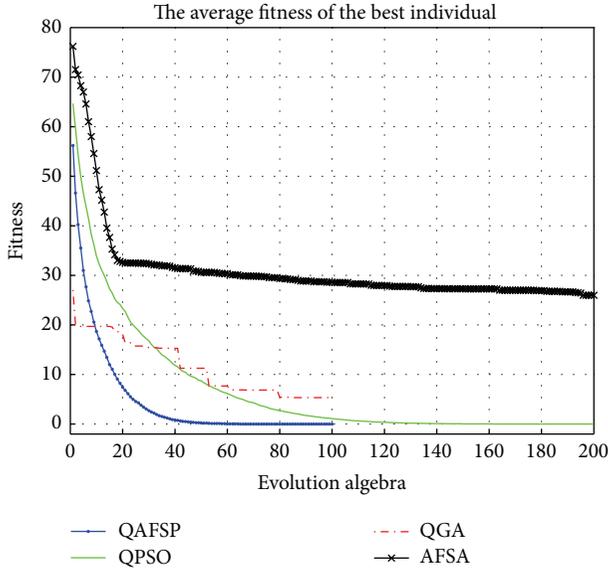


FIGURE 8: Convergence curve test function f_6 .

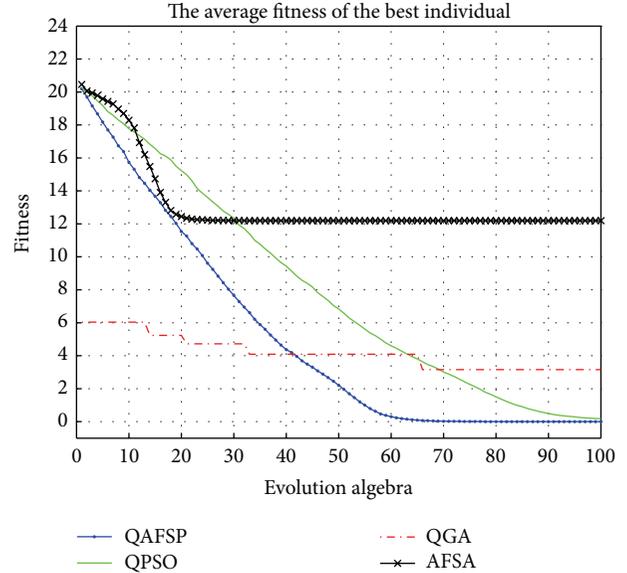


FIGURE 10: Convergence curve test function f_8 .

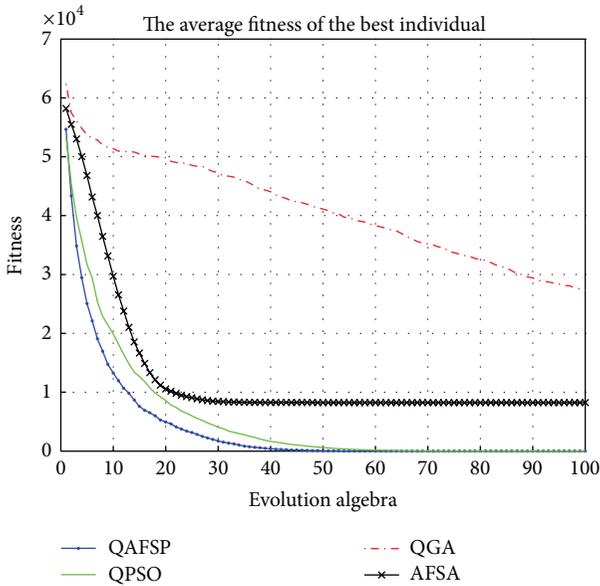


FIGURE 9: Convergence curve test function f_7 .

independent experiment ratio. Four kinds of algorithms on eight benchmark functions result in Table 2.

As can be seen from the table, strong optimization ability of quantum particle swarm algorithm (QAFSP) is based on artificial fish school. Using the same even smaller population size, the number of iterations, compared with the other three algorithms, QAFSP solves the optimal solution closer to the theoretical minimum function. In separate experiments 50 times, the number of successful optimization QAFSP is more and the success rate is close to 1. Experimental results show that the improved algorithm can be relatively stable to meet the accuracy requirements of searching the global optimum. In addition, the test functions include

higher dimension complex functions, although the other three algorithms to optimize the effect of low-dimensional test function better, the high-dimensional functions lose the optimization capabilities gradually, meanwhile QAFSP Shows better optimization ability to high-dimensional functions or low-dimensional functions.

In Figures 3, 4, 5, 6, 7, 8, 9, and 10 four algorithms on eight test function optimization iterations convergence plan were given. Where “fitness” represents the result value of test functions during the algorithm runs, “evolution algebra” is the number of iterations performed for solving test functions.

From the convergence graph functions, you can see that quantum particle swarm algorithm based on artificial fish (QAFSP) has better convergence. Compared with the other three optimization algorithms, QAFSP is fast and stable convergence. Experimental results show that the improved quantum particle swarm optimization improved quantum particle swarm premature convergence and effectively improved the performance of the algorithm.

5. Conclusion

On the basis of the quantum particle swarm optimization, combinations of artificial fish clusters and rear-end activities, a new improved algorithm is provided. The new algorithms use a dynamic parameter adjustment which is conducive to jump out of local optima and move to global extreme. Through the optimization of test functions, the experimental results show that the improved algorithm can improve the algorithm for solving ability and can better solve the global optimum to meet the accuracy requirements.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This paper was supported by the National Natural Science Foundation of China (61173056).

References

- [1] J. Sun, B. Feng, and W. B. Xu, "Particle swarm optimization with particles having quantum behavior," in *Proceedings of the Congress on Evolutionary Computation (CEC '04)*, pp. 325–331, June 2004.
- [2] P. Moore and G. K. Venayagamoorthy, "Evolving combinational logic circuits using a hybrid quantum evolution and particle swarm inspired algorithm," in *Proceeding of the NASA/DoD Conference on Evolvable Hardware (EH '05)*, pp. 97–102, July 2005.
- [3] S. M. Mikki and A. A. Kishk, "Quantum particle swarm optimization for electromagnetics," *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 10, pp. 2764–2775, 2006.
- [4] M. M. Soliman, A. E. Hassanien, and H. M. Onsi, "An adaptive medical images watermarking using Quantum Particle Swarm Optimization," in *Proceedings of the 35th International Conference on Telecommunications and Signal Processing (TSP '12)*, pp. 735–739, Prague, Czech Republic, July 2012.
- [5] L. dos Santos Coelho, F. A. Guerra, B. Pasquim, and V. Cocco Mariani, "Chaotic quantum-behaved particle swarm optimization approach applied to inverse heat transfer problem," in *Proceedings of the 5th International Joint Conference on Computational Intelligence (IJCCI '13)*, pp. 97–102, 2013.
- [6] A. M. A. C. Rocha and M. F. P. Costa, "An artificial fish swarm filter-based method for constrained global optimization," in *Proceedings of the 12th International Conference Computational Science and Its Applications (ICCSA '12)*, M. G. P. Fernandes, Ed., vol. 7335 of *Lecture Notes in Computer Science*, pp. 57–71, 2012.
- [7] F. Q. Liu and H. W. Zhang, "Dynamic clustering based on quantum-behaved particle swarm optimization," *Advanced Materials Research*, vol. 798, pp. 808–813, 2013.
- [8] H. Li and S. Li, "Quantum particle swarm evolutionary algorithm with application to system identification," in *Proceedings of the International Conference on Measurement, Information and Control (MIC '12)*, vol. 2, pp. 1032–1036, May 2012.
- [9] S. Deb, N. B. Singh, and S. K. Sarkar, "Particle swarm approach for parameter optimization of quantum well nano structure," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12999–13004, 2011.
- [10] D. Yazdani, A. N. Toosi, and M. R. Meybodi, "Fuzzy adaptive artificial fish swarm algorithm," in *AI 2010: Advances in Artificial Intelligence*, vol. 6464 of *Lecture Notes in Computer Science*, pp. 334–343, 2011.

Research Article

Research of Ant Colony Optimized Adaptive Control Strategy for Hybrid Electric Vehicle

Linhui Li, Haiyang Huang, Jing Lian, Baozhen Yao, Yafu Zhou, Jing Chang, and Ning'an Zheng

State Key Laboratory of Structural Analysis for Industrial Equipment, School of Automotive Engineering, Faculty of Vehicle Engineering and Mechanics, Dalian University of Technology, Dalian 116024, China

Correspondence should be addressed to Jing Lian; lianjing80@126.com

Received 6 June 2014; Revised 24 July 2014; Accepted 2 August 2014; Published 28 August 2014

Academic Editor: Rui Mu

Copyright © 2014 Linhui Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Energy management control strategy of hybrid electric vehicle has a great influence on the vehicle fuel consumption with electric motors adding to the traditional vehicle power system. As vehicle real driving cycles seem to be uncertain, the dynamic driving cycles will have an impact on control strategy's energy-saving effect. In order to better adapt the dynamic driving cycles, control strategy should have the ability to recognize the real-time driving cycle and adaptively adjust to the corresponding off-line optimal control parameters. In this paper, four types of representative driving cycles are constructed based on the actual vehicle operating data, and a fuzzy driving cycle recognition algorithm is proposed for online recognizing the type of actual driving cycle. Then, based on the equivalent fuel consumption minimization strategy, an ant colony optimization algorithm is utilized to search the optimal control parameters "charge and discharge equivalent factors" for each type of representative driving cycle. At last, the simulation experiments are conducted to verify the accuracy of the proposed fuzzy recognition algorithm and the validity of the designed control strategy optimization method.

1. Introduction

Combined with the feature of traditional gasoline vehicle and pure electric vehicle, hybrid electric vehicle (HEV) improves the fuel economy and emission performance while sustaining enough travel distance, and it has become an important development direction of automotive industry [1]. If the energy management control strategy of HEV can realize the reasonable distribution between the vehicle-mounted multiple energy power sources, the fuel economy and emission would be improved, the lifetime of power battery would be extended, and the vehicle maintenance cost would be minimized under the requirement for vehicle dynamic performance [2, 3].

Early energy management control strategy of HEV includes rule-based strategies and optimization-based strategies [4]. Most of them are usually based on the fixed control parameters that could not adapt the dynamic driving cycles [5–7]. The actual energy saving effect seems to be unsatisfactory. Later, researchers find that the fuel consumption and

emissions are sensitive to the driving cycle variation, and they start to study driving cycle recognition and adaptive control strategies in two aspects. One is using global position system (GPS), car navigation system, car to car communication, and other approaches to acquire the future road and traffic information such as average vehicle speed, road grade, and turning radius and then obtain the approximate global optimal energy distribution principles through the dynamic programming or other optimization algorithms [8]. But this kind of method needs a complex hardware implementation, and the global optimization needs large calculating quantity which may lead to a poor real-time performance. There is difficulty in its popularization and application. Another aspect of research is utilizing pattern recognition technology to identify the current type of driving cycle, according to the vehicle state parameters in the past period of time, such as average speed, idle time, and maximum acceleration [9]. It is relatively easy to implement, and this research method is selected to develop the adaptive control strategy in this paper.

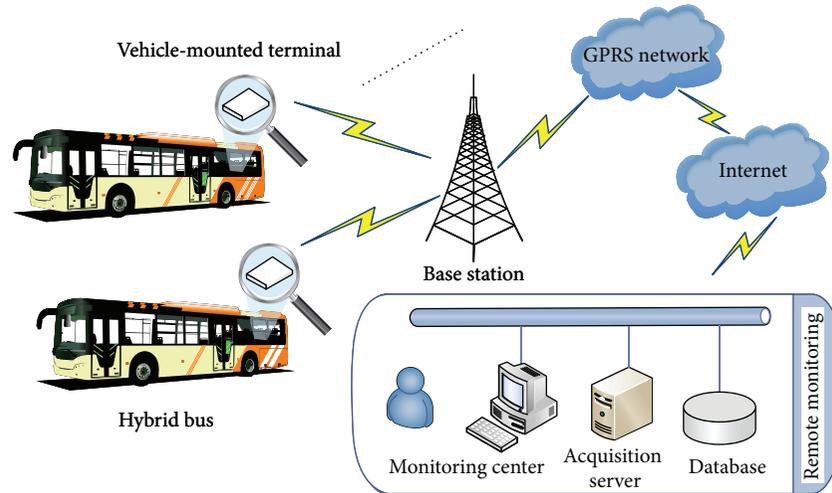


FIGURE 1: Remote data acquisition and monitoring system for HEV.

At present, the research on driving cycle recognition mainly contains neural network, support vector machine (SVM) and other predictive methods. Engström and Victor [10] proposed a statistical pattern recognition framework to analyze the collected vehicle operating data and utilized feed-forward neural network to classify the actual driving cycle into four types which are highway, arterial road, suburban, and urban, respectively. But the neural network-based method required a large number of suitable training samples to obtain a relatively accurate recognition result. Watanabe and Katsura [11] proposed an SVM driving cycle recognition method; it was suitable for the two-class classification problem, but it was more difficult to solve the multiclass classification problems. Gong et al. [12] proposed an iterative Markov chain approach for generating velocity profiles, which represent the specific driving pattern well based on the comparison of the phase plot to the typical real driving cycle. In this paper, a fuzzy driving cycle recognition algorithm is proposed to lay the foundation for the control strategy's adaptive adjustment.

The optimal control parameters in different types of driving cycles need to be determined after realizing the driving cycle recognition. The control parameters in present researches are mainly selected discretely according to the engineering experience, and then a relatively optimal solution is obtained through the simulating calculation; the parameter optimization result has the potential to be improved. Generally the optimization algorithm would be used to solve this kind of parameter optimization problem. As HEV is a strongly nonlinear complicated system, it takes a large amount of time for calculating the objective function according to the vehicle model; the optimization algorithm should have a fast convergence speed. Therefore, intelligent optimization algorithms such as genetic algorithm [13, 14], particle swarm optimization [15, 16], and simulated annealing algorithm [17] are introduced to solve such parameter optimization problem. In recent years, swarm intelligence algorithm did well in solving travelling salesman problem

(TSP) and other NP-complete problems [18–21], and it also has some applications in the field of optimal control research [22]. In this paper, utilizing the feature of automatic gain and accumulating the knowledge about search-space, we introduce an ant optimization algorithm to solve the HEV optimal control parameters in each type of driving cycle.

The rest of the paper is structured as follows. In Section 2, four types of representative driving cycles are constructed based on the actual vehicle operating data and a fuzzy driving cycle recognition algorithm is proposed for online recognizing the type of actual driving cycle. Section 3 introduces basic equivalent fuel consumption minimization strategy and studies the off-line control parameter optimization in different driving cycles based on the ant colony optimization method. Section 4 presents the simulation experiment results of the designed adaptive control strategy. Section 5 concludes the presented work.

2. Driving Cycle Classification and Recognition

To achieve the objective of making control strategy being able to adaptively adjust according to different types of driving cycles, the driving cycle classification and recognition should be implemented in advance. In this section, driving cycles are classified into four types and representative cycle for each type is built to reflect the geography and traffic features in different regions. Then, a fuzzy clustering center matrix and the corresponding relative membership degree function are defined to realize the driving cycle recognition.

2.1. Construction of Representative Driving Cycles. The classification and construction of four types of driving cycles are based on the independently developed remote data acquisition and monitoring system [23]; it has been operating for nearly five years, as shown in Figure 1. The original vehicle data are gathered from the hybrid electric buses on the Dalian Energy Efficient and New Energy Vehicle

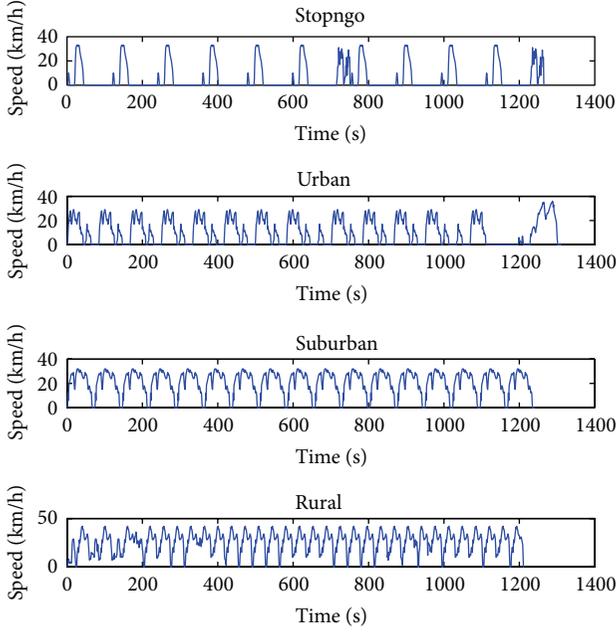


FIGURE 2: Four representative driving cycles.

Demonstration Project. A large number of reliable vehicle real-time operating data are collected from the Controller Area Network (CAN) bus through vehicle mounted terminal. The data can be divided into four types according to the city structure features; they are stopngo to represent the traffic jam cycles in the downtown, urban to represent the low speed flow in urban areas, suburban to represent the medium speed flow, and rural to represent the high speed flow in rural areas.

The steps for constructing four types of representative driving cycles are as follows. First, the microtrips are divided from the original vehicle speed data (microtrips are defined as a small driving trip segment from a vehicle idle point to the next idle point), and the database of the microtrips can be obtained from the original speed statistic database. Second, the characteristic parameters of each separate microtrip are calculated; the principal components and the corresponding contribution rate of these characteristic parameters can be obtained by the principal component analysis. Thus the cluster analysis technique is able to classify the microtrips into four cluster centers easily by these principal components, and the microtrips which are nearest to the cluster center are chosen to construct the representative driving cycles. At last, four types of representative driving cycles are obtained by the linear combination of these selected microtrips. The results of the constructed four representative cycles are shown in Figure 2.

2.2. Fuzzy Recognition of Driving Cycle. The driving cycle recognition is based on the statistic characteristic parameters of actual driving cycle [24]; the commonly used cycle characteristic parameters are as follows: cycle average speed, average driving speed, maximum speed, mean acceleration,

mean deceleration, maximum acceleration, maximum deceleration, idle time, the percentage of idle time, number of stops, and so on. In this paper, average cycle speed V_m , average driving speed V_{mr} , average deceleration A_d , average acceleration A_a , and percentage of idle time η are selected as the characteristic parameters for fuzzy recognition of driving cycle.

The sample driving cycle segment characteristic vector need to be recognized is expressed as $x = \{V_m, V_{mr}, \eta, A_a, A_d\}^T$. Together with four groups of parameters of the typical driving cycle constructed in Section 2.1, the characteristic vector is assembled as a matrix $X_{5 \times 5}$ to be identified:

$$X = \begin{bmatrix} V_m & 5.69 & 10.66 & 20.67 & 24.55 \\ V_{mr} & 19.59 & 16.55 & 23.22 & 27.27 \\ \eta & 0.7093 & 0.3562 & 0.1096 & 0.0998 \\ A_a & 1.34 & 0.70 & 0.71 & 0.94 \\ A_d & -0.88 & -0.76 & -0.65 & -1.13 \end{bmatrix} \quad (1)$$

$$= (x_{ij})_{5 \times 5}.$$

As characteristic vector elements V_m , η , A_a are different in physical dimensions, the matrix to be recognized $X_{5 \times 5}$ should be normalized; the elements to be identified in the normalized matrix are expressed as $(r_{ij})_{5 \times 5}$, which are calculated as follows:

$$r_{ij} = \frac{x_{ij} - x_{i \min}}{x_{i \max} - x_{i \min}}. \quad (2)$$

Five index characteristic values of the four representative driving cycles make up the clustering center of the driving cycle class. After normalization it can be expressed as standard fuzzy clustering center matrix $(s_{ih})_{5 \times 4}$ in the fuzzy recognition:

$$S = \begin{bmatrix} 1.0000 & 0.7365 & 0.2057 & 0.0000 \\ 0.7478 & 1.0000 & 0.3778 & 0.0000 \\ 0.0000 & 0.5793 & 0.9839 & 1.0000 \\ 0.0000 & 1.0000 & 0.9844 & 0.6250 \\ 0.2500 & 0.7708 & 1.0000 & 0.0000 \end{bmatrix} = (s_{ih})_{5 \times 4}. \quad (3)$$

As characteristic parameters have different impact in the process of driving cycle fuzzy recognition, different weights of characteristic parameters need to be considered. So a characteristic indicator weight vector is defined as $W = (w_1, w_2, \dots, w_5)$ which should satisfy the constraint conditions of $w_1 + w_2 + w_3 + w_4 + w_5 = 1$. In this paper, the characteristic indicator weight vector is selected as $W = (0.44, 0.34, 0.10, 0.09, 0.03)$.

After the initialization above, the relative membership degree u_{hj} of sample x_j to the category h ($h = 1, 2, 3, 4$) is calculated as follows:

$$u_{hj} = \frac{1}{\sum_{k=1}^c \left(\sum_{i=1}^m [w_i(r_{ij} - s_{ih})]^2 / \sum_{i=1}^m [w_i(r_{ij} - s_{ik})]^2 \right)}. \quad (4)$$

So the sample driving cycle segment x_j is recognized as the driving cycle type h which has the maximum relative membership degree u_{hj} .

3. Control Parameter Optimization Based on Ant Colony Algorithm

3.1. Basic Equivalent Fuel Consumption Minimization Strategy. The basic energy management control strategy to be optimized in this paper is ECMS. Its main idea is to multiply the battery electricity consumption by an equivalent factor α and transfer this electric energy to an equivalent vehicle instantaneous fuel consumption. In every computing interval t , the total equivalent fuel consumption to be calculated is the sum of drive motor's electric equivalent fuel consumption and actual engine fuel consumption, which is shown as follows:

$$\dot{m}_{\text{eq}} = \dot{m}_m + \dot{m}_e, \quad (5)$$

where \dot{m}_{eq} is the total equivalent fuel consumption mass flow, in kg/s, and \dot{m}_e is the actual engine fuel consumption mass flow which can be calculated as follows:

$$\dot{m}_e = \frac{P_e}{\eta_e \cdot Q}, \quad (6)$$

where P_e is the engine output power, η_e is the engine working efficiency, and Q is the fuel low caloric value, in J/kg.

\dot{m}_m is battery's electric equivalent fuel consumption mass flow; as the actual power consumption of battery is electricity,

it should be converted to the equivalent fuel consumption through the following equations:

$$\dot{m}_m = \begin{cases} \alpha_{\text{dis}} \cdot \frac{P_m}{Q} \cdot \frac{1}{\eta_{\text{dis}}} \cdot \frac{1}{\eta_m}, & P_b > 0 \\ \alpha_{\text{chg}} \cdot \frac{P_m}{Q} \cdot \eta_{\text{chg}} \cdot \eta'_m, & P_b < 0, \end{cases} \quad (7)$$

where P_m is the motor output power, when motor works as a generator it was a negative value; η_{dis} is the battery discharge efficiency; η_{chg} is the battery charge efficiency; η_m is the motor drive efficiency; η'_m is the motor generating efficiency; α_{dis} is the discharge equivalent factor; α_{chg} is the charge equivalent factor.

In (5) the calculated battery equivalent fuel consumption is not related to the current battery State of Charge (SOC); the strategy cannot ensure the battery SOC maintaining around a nominal value and get an acceptable battery efficiency to preserve battery life. Therefore the motor equivalent fuel consumption needs to be penalized with a nonlinear function to control the fluctuation range of SOC and ensure the battery charge balance. Firstly, SOC value in every simple time t needs to be normalized as follows:

$$x_{\text{SOC}}(t) = \begin{cases} -1, & \text{SOC}(t) \leq \text{SOC}_{\text{min}} \\ \frac{\text{SOC}(t) - ((\text{SOC}_{\text{max}} + \text{SOC}_{\text{min}})/2)}{((\text{SOC}_{\text{max}} - \text{SOC}_{\text{min}})/2)}, & \text{SOC}_{\text{min}} < \text{SOC}(t) < \text{SOC}_{\text{max}} \\ 1, & \text{SOC}(t) \geq \text{SOC}_{\text{max}}, \end{cases} \quad (8)$$

where SOC_{max} and SOC_{min} are the battery SOC working range. To maintain the SOC balance, if the SOC is in a lower stage, penalty function should enlarge the motor equivalent fuel consumption \dot{m}_m to increase the cost of battery discharge and decrease the battery charge cost. Therefore the penalty function of SOC is selected as S-shape high order polynomials:

$$\beta(\text{SOC}) = 1 + 1.2(x_{\text{SOC}}(t))^4 - 2(x_{\text{SOC}}(t))^5. \quad (9)$$

Finally, the basic equivalent fuel consumption minimization control strategy can be simplified to an optimization problem in each instantaneous sample time:

$$J_{\text{min}} = \min(\beta \cdot \dot{m}_m + \dot{m}_e). \quad (10)$$

s.t.

$$\begin{aligned} P_{\text{req}}(t) &= (P_e(t) + P_m(t)) \eta_t, \\ P_{m.\text{min}}(t) &\leq P_m(t) \leq P_{m.\text{max}}(t), \\ P_{e.\text{min}}(t) &\leq P_e(t) \leq P_{e.\text{max}}(t), \\ P_{b.\text{min}}(t) &\leq P_b(t) \leq P_{b.\text{max}}(t), \end{aligned} \quad (11)$$

where $P_{\text{req}}(t)$ is the driver's instantaneous power demand; η_t is the drivetrain's working efficiency; motor output power $P_m(t)$

and engine output power $P_e(t)$ should be within a certain range constrained by the constraint of current operating speed and battery SOC. The electric motor output power $P_m(t)$ is selected as a control variable. Thus, the $P_m(t)$ which satisfy the constraint conditions and make the objective function minimum can be solved to determine the motor and the engine working point.

3.2. Parameter Optimization Problem. In the above ECMS methods, selection of discharge equivalent factor α_{dis} and charge equivalent factor α_{chg} will directly affect the vehicle's utilization of electric energy and eventually have an impact on vehicle fuel economy. For example, in a congested urban driving cycle, a smaller equivalent factor will make vehicles tend to consume electric energy and reduce the engine working at a low speed with lower efficiency and higher emission. Therefore, according to the driving cycle recognition result, the optimal charge and discharge equivalent factors need to be determined under each type of driving cycle. It means that parameters of the basic control strategy need to be optimized to adaptively adjust for a better fuel economy without losing power performance.

Currently, the widely used control parameter calculation method is realized by setting a number of disperse

experimental values and then adjusting the parameter with trial and error method. And this calculation method always relies on the engineering experience. Although this method is practical, it cannot achieve the best efficiency of the power system. Therefore, it is necessary to introduce an optimization method to optimize HEV control parameters.

However, classic optimization methods generally require the objective function to be continuous and differentiable, and the sawtooth phenomenon leading to a slow convergence may occur while approaching the optimal solution. HEV is a complicated nonlinear system; in its control parameter optimization problem, it is difficult to find the optimal solution rapidly by using these classic optimization methods. Thus, this paper proposed an ant colony HEV control parameter optimization method to optimize the charge and discharge equivalent factor in each type of driving cycle, respectively. It has features of parallelism and positive feedback, and it can make full use of the vehicle model information.

This HEV control parameter optimization problem is to search the best solution for the objective function which minimizes the overall fuel energy consumption in every representative driving cycles constructed in Section 2. But as the charge and discharge equivalent factors are varied in the parameter optimization, the electric energy consumption under different parameters should be assessed according to a unified standard. Therefore we use the battery average efficiency $\bar{\eta}_b$, motor average efficiency $\bar{\eta}_m$, and transmission system average efficiency $\bar{\eta}_t$ to convert the battery energy consumption to the oil consumption; it can be expressed as follows:

$$\dot{m}_b = \frac{\Delta P_b}{\bar{\eta}_b \bar{\eta}_m \bar{\eta}_t}. \quad (12)$$

Then the objective function, total equivalent fuel energy consumption can be expressed as

$$\min F(X) = \int_0^t (\dot{m}_e + \dot{m}_b) dt, \quad (13)$$

where X is the control parameters "charge and discharge equivalent factors" to be optimized under each type of driving cycle; generally its value range is from 2 to 3.5. This optimization problem should satisfy the constraint of driving cycle's real-time speed requirement. The objective function value is the integration of total equivalent fuel consumption mass flow in (5) at every sample moment of the driving cycle to be optimized. Rules for ant colony parameter optimization are as follows.

(1) *Ant Initialization*. Randomly distribute the given number of M ants in a certain position of the variable definition domain $[a, b]$; each ant i is positioned as follows:

$$x(i, k) = a(k) + \frac{b(k) - a(k)}{M} (i - 1 + \text{rand}), \quad (14)$$

where rand is a random number between $[0, 1]$.

The initial pheromone quantity of each ant's position can be expressed as

$$\Delta\tau_{X(i)} = e^{-F(X_i)}, \quad (15)$$

where $F(X_i)$ is the objective function value of ant i .

(2) *Ant Travelling Rules*. After all the ants accomplish a searching process, one of them will find an optimal position in the current loop which will be the transfer guide for the rest ants' travelling in the next loop. So the ants transfer can be divided into two parts, one is the global search for the ants that have not found the optimal solutions moving towards the optimal solution $X(\text{Best})$, its transfer probability and step length is related to the amount of pheromone and relative positions of $X(i)$ and $X(\text{Best})$; the transfer probability is calculated as follows:

$$P_{i,\text{Best}} = \frac{e^{(\tau_{X(\text{Best})} - \tau_{X_i})}}{e^{\tau_{X(\text{Best})}}}. \quad (16)$$

When ant i moves to a large quantity of pheromone information position, it may find a better optimized solution. Thus the transfer step length of ant i is defined in (16) when it is moving to the best position $X(\text{Best})$:

$$X_i = \begin{cases} X_i + \lambda (X_{\text{Best}} - X_i), & P_{i,\text{Best}} < P_o \\ X_i + \text{rand}(-1, 1) * \frac{b-a}{M}, & \text{otherwise,} \end{cases} \quad (17)$$

where $0 < \lambda < 1$, $0 < P_o < 1$.

Another part of ants transfer is local search of the Best ant. It randomly searches for a better solution in a small determined neighborhood. The search radius decreases with the increase of iterations to find a more accurate solution in the later search period. The rules for local search are shown as follows:

$$X_{\text{Best}} = \begin{cases} X_{Is}, & F(X_{Is}) < F(X_{\text{Best}}) \\ X_{\text{Best}}, & \text{otherwise,} \end{cases}$$

$$X_{Is} = \begin{cases} X_{\text{Best}} + \omega \cdot r, & \text{rand}(1, 1) < 0.5 \\ X_{\text{Best}} - \omega \cdot r, & \text{otherwise,} \end{cases} \quad (18)$$

$$\omega = 1 - 0.4 \cdot \frac{i_{\text{ter}}}{n_{\text{max}}},$$

where i_{ter} is the current number of iterations and n_{max} is the maximum iteration number.

(3) *Pheromone Update Rules*. When the global and local search are finished, the pheromone information of every ants' position needs to be updated as follows:

$$\tau_{X(i)} = \rho \cdot \tau_{X(i)} + \Delta\tau_{X(i)}, \quad (19)$$

where ρ is the pheromone volatilization coefficient and $0 < \rho < 1$, $\Delta\tau_{X(i)}$ is calculated in (14).

(4) *Solve Procedure of Parameter Optimization*. The HEV control parameter optimization solving procedure is as follows.

- (a) Determine the maximum iteration numbers n_{max} , the ant colony size M , and the value range of control parameter X .
- (b) Initialize current ant colony position and the corresponding pheromone quantity according to (14)-(15).

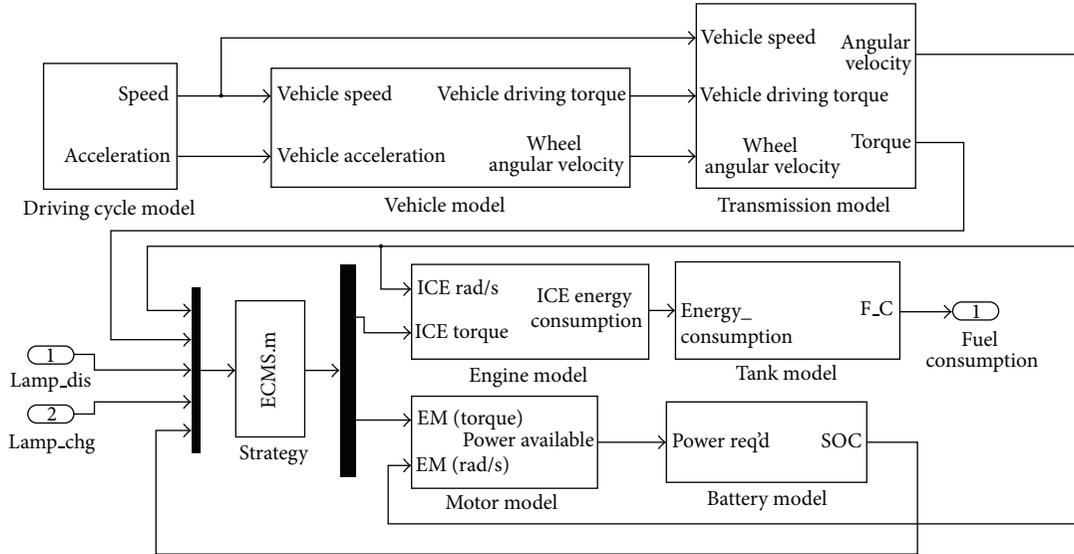


FIGURE 3: The hybrid electric bus fuel consumption simulation model.

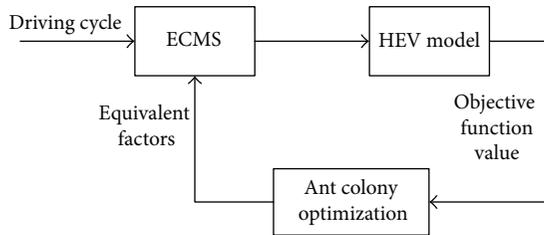


FIGURE 4: The parameter optimization process.

- (c) Determine the ant at the optimal position X_{Best} according to the object function.
- (d) The ants that did not find the optimal solutions conduct the global search to update ants' position according to (16)-(17).
- (e) The ant at the optimal position X_{Best} does the local search and updates optimal position according to (18).
- (f) Update the pheromone information with (19).
- (g) If the termination condition is satisfied which means $i_{ter} \geq n_{max}$, then finish the loop and output the optimal solution; otherwise go to step (c).

4. Simulation and Analysis

The ant colony control parameter optimization algorithm is realized by a Matlab m-language program. The relevant parameter settings are as follows: maximum iteration numbers $n_{max} = 15$, ant colony size $M = 30$, control variables charge, and discharge equivalent factors value range $X_1, X_2 \in [1.5, 3.5]$, $\lambda = 0.3$, $\rho = 0.95$.

The optimized object is a parallel hybrid electric bus in this paper; its basic power component's technical parameters are shown in Table 1.

TABLE 1: The parameters of HEV.

Engine	DUETZ BF6M2012-22E4
	Speed (rated/peak): 3600/6500 rpm
Motor (PMSM)	Power (rated/peak): 30/90 kW
	Torque (rated/peak): 80/240 N·m
	Capacity: 40 Ah
Ni-MH battery	Number of modules: 28
	Nominal voltage: 12 (volt/module)
AMT	5 speed, ratio: 5.785, 3.038, 1.623, 1.00, 0.773
Vehicle	Curb weight: 12000 kg
	Size: 11996 × 2550 × 3174 (mm)

The objective function Equation (12) is evaluated through a Matlab/Simulink model. According to the features of hybrid powertrain and ECMS control method, the vehicle fuel consumption simulation calculating model is constructed, as shown in Figure 3.

The input of this simulation model is the four representative driving cycles constructed in Section 2, respectively, and the control parameters charge and discharge equivalent parameters are adjusted by the ant colony optimization algorithm. While the algorithm's termination condition is satisfied, the objective fuel consumption calculated by the model will be minimized under the corresponding driving cycle. The parameter optimization process is shown in Figure 4.

With the control parameters varied in the iteration, the objective function equivalent oil consumption convergence procedure in four different types of driving cycles are shown in Figure 5. In this procedure, the ant algorithm adjusts the power distribution between engine and motor by changing the value of charge and discharge equivalent factors, so as to release and recover the electric energy more reasonable and effective, and to optimize the engine working range while making the objective function converge to the optimal value.

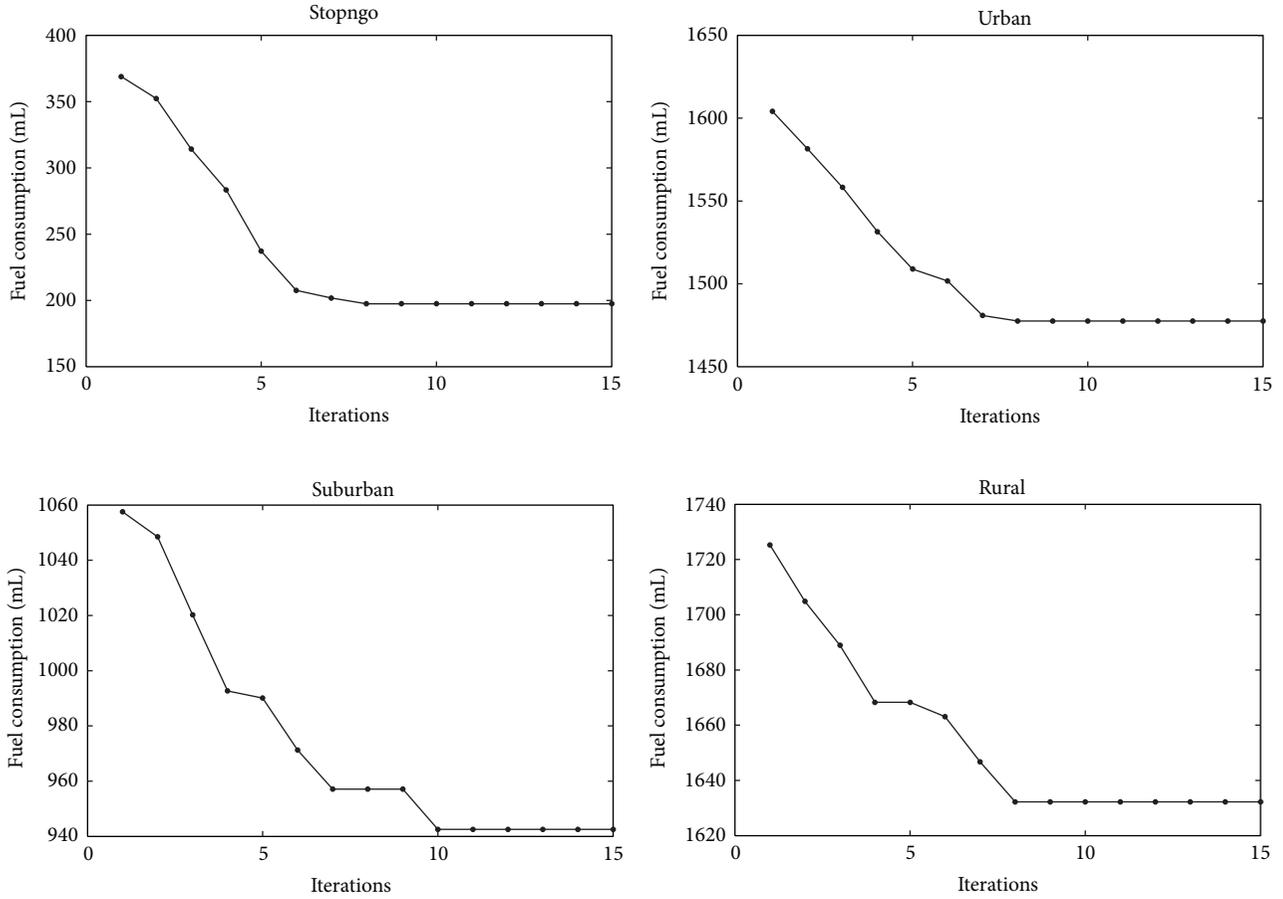


FIGURE 5: The objective function converge procedure in ant colony optimization.

TABLE 2: The optimal control parameters corresponding to each typical driving cycle.

The name of driving cycles	Charge equivalent factor	Discharge equivalent factor
Stopngo	1.52	2.27
Urban	1.56	2.52
Suburban	1.91	3.28
Rural	2.56	3.31

The final optimal value of charge and discharge equivalent factors under each driving cycle is shown in Table 2.

In order to achieve the purpose of making basic ECMS have the ability of adaptive adjustment with driving cycle and greatly improving the vehicle performance, an adaptive scheme of optimized ECMS is designed in this paper as shown in Figure 6.

The optimal control parameters and driving cycle recognition part are added into the hybrid electric bus Simulink strategy model (as shown in Figure 3) to test the control strategy’s performance. The driving cycle tested in the process of simulation is the Dalian cycle (as shown in Figure 7(a)). It is constructed based on a real-time operating database

which was collected from the hybrid electric buses in Dalian for four years by the remote vehicle-mounted data acquisition system for new energy vehicle. The total cycle time is 1235 s and it can reflect the actual geographical and traffic features of the Dalian area.

While Dalian driving cycle is inputted to the HEV Simulink model, the strategy module could receive vehicle speed information from the driving cycle module. The codes in the strategy module’s Matlab m-file which realized the driving cycle recognition algorithm proposed in Section 2 calculate the relative membership degree u_{hj} . Thus the driving cycle type of microtrips in the recognition period can be identified, and the optimal equivalent factors can be updated in the strategy module realizing the adaptive ECMS scheme as shown in Figure 6. The recognition period is set to be 10 s in this paper. Recognition results are shown in Figure 7(b). Types 1, 2, 3, and 4 represent stopngo, urban, suburban and rural, respectively. From the recognition results, we can see that the driving cycle type can be identified well; speed and other characteristic parameter values are in accordance with the corresponding recognized representative cycle. The real driving conditions in the Dalian area can be reflected.

Basic ECMS control method is simulated as a contrast to the optimized adaptive control strategy. This paper is mainly studying the effect of charge and discharge equivalent

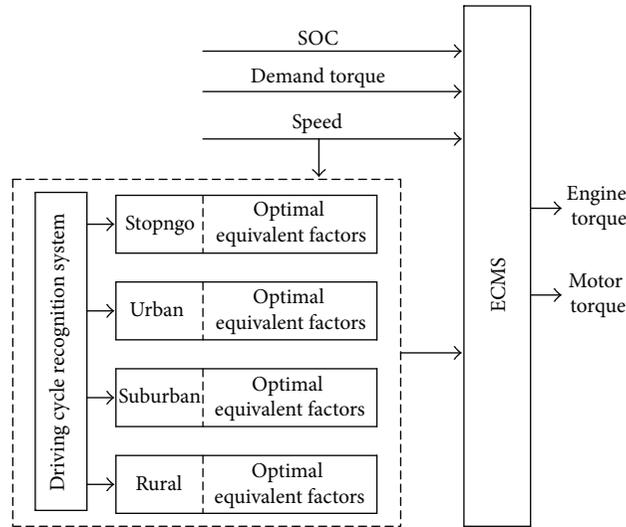


FIGURE 6: The adaptive ECMS scheme.

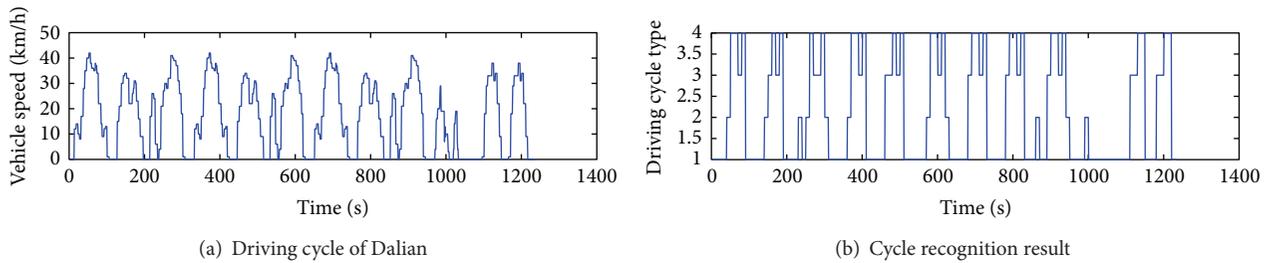


FIGURE 7: The driving cycle of Dalian and its recognition result.

factors on the fuel economy, and the vehicle fuel economy performance results are determined by the energy power distribution between engine and motor, so the battery power curve in the whole cycle is the best way to express the energy distribution and the utilization of electric power. The contrast of the battery power curve between basic ECMS and optimized adaptive strategy is shown in Figure 8. From the result we can see that for the adaptive control strategy battery charge and discharge are fewer under middle high speed than those frequent battery charge and discharge in the basic ECMS. It is because that the charge equivalent factor is smaller. The discharge equivalent factor is larger in the middle high speed urban and suburban driving cycles under the optimized adaptive ECMS, and the cost to charge and discharge is large for the objective function of ECMS. As a result, the vehicle tends to use more engines to power the vehicle and reduce the battery charge and discharge. On the other hand, the adaptive control strategy tends to discharge more at a lower speed and charge more at a higher speed; it is also the consequence of the equivalent factor's adjustment under rural and stopngo driving cycles.

The fuel consumption and SOC variation curves are shown in Figure 9; it shows that the adaptive control strategy proposed in this paper has a better fuel economy with a bit lower final SOC; the engine oil consumption is 14% lower than the basic ECMS.

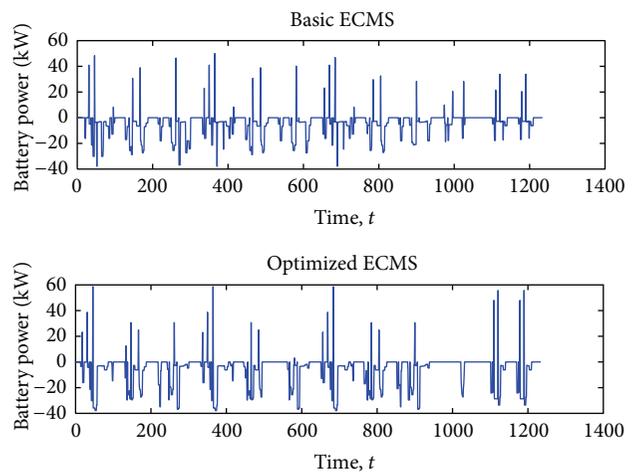


FIGURE 8: Battery power comparison of two control strategies.

5. Conclusion

In this paper, control strategy of HEV is further researched on the basis of remote data acquisition and monitoring system. Then an adaptive control strategy based on the ant colony parameter optimization for HEV is proposed. It can

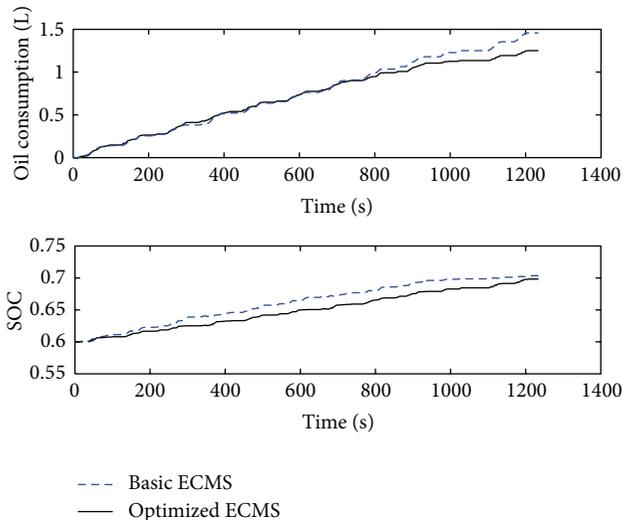


FIGURE 9: Oil consumption and SOC comparison curve.

adaptively adjust the control parameters according to the real driving cycle, and it is effective in improving vehicle fuel economy of hybrid electric vehicle. The main work of this paper includes: four representative driving cycles are constructed according to the vehicle operating data for the past five years; a fuzzy driving cycle recognition algorithm based on a relative membership degree function is proposed; for online recognizing the type of actual driving cycle; for each type of driving cycle, the optimal control parameters corresponding to each type of driving cycle are determined by using an ant colony optimization method which can effectively shorten the control parameter's adjustment time in the HEV road test; the validity and accuracy of the algorithm are verified by the simulation experiments at last. The results show that according to the on-line driving cycle recognition, vehicle controller is adjusted to the corresponding optimal control parameters, which realized the control strategy adaptive adjust with the variation of actual driving cycles, and the proposed control method improves vehicle fuel consumption effectively.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This project is supported by the National Natural Science Foundation of China (Grant nos. 51107006, 61203171, and 61473057) and China Postdoctoral Science Foundation (Grant nos. 2012M510799 and 2013T60278).

References

- [1] R. Dosthosseini, A. Z. Kouzani, and F. Sheikholeslam, "Direct method for optimal power management in hybrid electric vehicles," *International Journal of Automotive Technology*, vol. 12, no. 6, pp. 943–950, 2011.
- [2] S. Brown, D. Pyke, and P. Steenhof, "Electric vehicles: the role and importance of standards in an emerging market," *Energy Policy*, vol. 38, no. 7, pp. 3797–3806, 2010.
- [3] J. Lian, H. Han, L. H. Li, Y. F. Zhou, and J. Feng, "Research on optimal control method of hybrid electric vehicle," *Simulation-Transactions of the Society for Modeling and Simulation International*, vol. 89, no. 9, pp. 1137–1146, 2013.
- [4] K. Çağatay Bayindir, M. A. Gözüküçük, and A. Teke, "A comprehensive overview of hybrid electric vehicle: powertrain configurations, powertrain control techniques and electronic control units," *Energy Conversion and Management*, vol. 52, pp. 1305–1313, 2011.
- [5] N. Sadegh, B. Khan, and J. Meisel, "Optimization of the fuel consumption of a parallel hybrid electric vehicle, advanced intelligent mechatronics," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 128–133, 2005.
- [6] J.-P. Gao, G.-M. Zhu, E. G. Strangas, and F.-C. Sun, "Equivalent fuel consumption optimal control of a series hybrid electric vehicle," *Proceedings of the Institution of Mechanical Engineers D: Journal of Automobile Engineering*, vol. 223, no. 8, pp. 1003–1018, 2009.
- [7] D. Wang, X. Lin, and Y. Zhang, "Fuzzy logic control for a parallel hybrid hydraulic excavator using genetic algorithm," *Automation in Construction*, vol. 20, no. 5, pp. 581–587, 2011.
- [8] N. Cui, J. Fan, C. Zhang, and J. Wu, "Research on predictive control based energy management strategy for Hybrid Electric Vehicle," in *Proceeding of the 3rd IEEE International Symposium on Power Electronics for Distributed Generation Systems (PEDG '12)*, pp. 642–646, Aalborg, Denmark, June 2012.
- [9] A. Ravey, B. Blunier, S. Lukic, and A. Miraoui, "Control strategy of fuel cell hybrid electric vehicle based on driving cycle recognition," in *Proceedings of the IEEE Transportation Electrification Conference and Expo (ITEC '12)*, pp. 1–5, June 2012.
- [10] J. Engström and T. Victor, "System and method for real-time recognition of driving patterns," U.S. Patent 6,879,969, 2005.
- [11] T. Watanabe and S. Katsura, "On-line recognition of driving road condition using support vector machine," in *Proceedings of the IEEE International Conference on Industrial Technology (ICIT '11)*, pp. 405–410, March 2011.
- [12] Q. Gong, S. Midlam-Mohler, V. Marano, and G. Rizzoni, "An iterative markov chain approach for generating vehicle drive cycles," *SAE International Journal of Engines*, vol. 4, no. 1, pp. 1035–1045, 2011.
- [13] H. S. Hamut, I. Dincer, and G. F. Naterer, "Analysis and optimization of hybrid electric vehicle thermal management systems," *Journal of Power Sources*, vol. 247, pp. 643–654, 2014.
- [14] B. Yu, Z. Z. Yang, and J. B. Yao, "Genetic algorithm for bus frequency optimization," *Journal of Transportation Engineering*, vol. 136, no. 6, pp. 576–583, 2010.
- [15] S. Caux, D. Wanderley-Honda, D. Hissel, and M. Fadel, "On-line energy management for HEV based on particle swarm optimization," in *Proceedings of the IEEE Vehicle Power and Propulsion Conference (VPPC '10)*, pp. 1–7, September 2010.
- [16] J. Wu, C. H. Zhang, and N. X. Cui, "PSO algorithm-based parameter optimization for HEV powertrain and its control strategy," *International Journal of Automotive Technology*, vol. 9, no. 1, pp. 53–59, 2008.
- [17] Y. Deng, K. Chen, and J. E, "Parameters optimization for HEV based on simulated annealing particle swarm algorithm," *Automotive Engineering*, vol. 34, no. 7, pp. 580–584, 2012.

- [18] B. Yao, C. Yang, J. Hu, J. Yao, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6-7, pp. 2127–2131, 2011.
- [19] B. Z. Yao, P. Hu, M. H. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem," *Simulation*, vol. 89, no. 6, pp. 762–770, 2013.
- [20] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research C*, vol. 43, pp. 233–248, 2014.
- [21] B. Yu, Z. Z. Yang, and B. Z. Yao, "A hybrid algorithm for vehicle routing problem with time windows," *Expert Systems with Applications*, vol. 38, no. 1, pp. 435–441, 2011.
- [22] J. van Ast, R. Babuška, and B. de Schutter, "Ant Colony Optimization for optimal control," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '08)*, pp. 2040–2046, June 2008.
- [23] J. Lian, Y. Zhou, T. Ma, X. Shen, and J. Li, "Research on data monitoring system of Hybrid Electric Vehicle," *Sensor Letters*, vol. 9, no. 5, pp. 2012–2016, 2011.
- [24] D. Savvidis, H. Li, G. Andrews, and C. Ioakimidis, "Analysis of various driving parameters and emissions for passenger cars driven with and without stops at intersections under different test cycles," *Diesel Engine*, vol. 2011, pp. 8–19, 2011.

Research Article

Damage Identification of Bridge Based on Modal Flexibility and Neural Network Improved by Particle Swarm Optimization

Hanbing Liu, Gang Song, Yubo Jiao, Peng Zhang, and Xianqiang Wang

College of Transportation, Jilin University, No. 5988 Renmin Street, Changchun 130025, China

Correspondence should be addressed to Yubo Jiao; jiaoyubo86@163.com

Received 3 June 2014; Accepted 20 July 2014; Published 14 August 2014

Academic Editor: Bin Yu

Copyright © 2014 Hanbing Liu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An approach to identify damage of bridge utilizing modal flexibility and neural network optimized by particle swarm optimization (PSO) is presented. The method consists of two stages; modal flexibility indices are applied to damage localizing and neural network optimized by PSO is used to identify the damage severity. Numerical simulation of simply supported bridge is presented to demonstrate feasibility of the proposed method, while comparative analysis with traditional BP network is for its superiority. The results indicate that curvature of flexibility changes can identify damages with both single and multiple locations. The optimization of bias and weight for neural network by fitness function of PSO algorithm can realize favorable damage severity identification and possesses more satisfactory accuracy than traditional BP network.

1. Introduction

As important components of transportation infrastructure, bridges are essential for normal operation of transportation system. However, their loading capacities are threatened by external environment (such as vehicles and temperature), which reduces the level of security service. Therefore, it is necessary to periodically monitor the health status of structure [1]. Damage identification is one of the major challenges in bridge health monitoring. It can help to evaluate the safety condition, prevent catastrophic collapse, and also provide information for maintenance [2].

The damages of bridge can be defined as any deviation in structure's original geometric and material properties [3]. The reasons include reduction of structural stiffness, material cracks, and fatigue failure. A number of methods have been proposed in the past two decades to detect and assess the damage condition of bridge. Nondestructive methods such as ultrasonic waves, X-ray, and stress waves have been widely applied in practice considering their convenience and simplicity. Most of these methods, however, rely on a presumption of the determination of suspected damage regions and are restricted for local detection. When applied to large structures, these methods are time-consuming and costly [4, 5].

Vibration-based damage identification methods are active in this research area, which, as compared to local non-destructive methods, does not require the a priori knowledge of damage locations. Damage can be detected based on changes in natural frequencies and mode shapes, which can be regarded as global methods. The techniques are based on the theoretical basis where damages alter the physical properties of bridge (e.g., mass, stiffness, and damping). Furthermore, the dynamic characteristics (e.g., mode shape, frequency, and damping ratios) will change. Therefore, damage location and severity can be assessed by analyzing the structure's dynamic characteristics [6–8].

Natural frequencies are widely used as damage indicators in the early stage of dynamic-based damage detection as they are easy to obtain. However, they are verified to be insensitive to damage and susceptible to environmental changes [3, 8]. Mode shapes contain more information of damage location and are more suitable for damage identification than natural frequencies. However, it is difficult to realize the damage localizing that relies only on mode shapes data [6]. Modal flexibility has shown itself to be a promising damage indicator due to its high sensitivity to damage, because it contains the information of both natural frequency and mode shape. Pandey and Biswas [9] proposed a damage identification

approach using changes in modal flexibility for the first time. In reality, it is impractical to identify all of the modes. However, the flexibility matrix can be accurately estimated based on the first few modes because it is inversely proportional to the squares of natural frequencies. Li et al. [10] applied the flexibility approach to damage identification of cantilever-type structures. Stutz et al. [11] presented a flexibility-based continuum damage detection method. Reynders and de Roeck [12] proposed a local flexibility-based approach which allowed determining the local stiffness variations directly from modal properties. It was verified by numerical simulation of damaged isostatic and hyperstatic beam and also experiments of a reinforced concrete beam. Catbas et al. [13] adopted modal flexibility for identifying structural behavior after damage which could be evaluated by inspecting the deflected shapes.

Most of dynamic-based damage identification methods can be used to identify damage presence and locations. But it is difficult to assess the damage severity. Furthermore, the computational time and costs are other challenges in damage detection. The applications of different computational intelligence approaches such as artificial neural networks (ANNs), genetic algorithm (GA), and particle swarm optimization (PSO) have been proved to be effective in damage severity assessment and improving the computational efficiency [14–16]. ANNs [17, 18] are information processing systems which mimic the network structure of actual human brain. Mehrjoo et al. [19] presented a method for assessing the damage intensities of joints for truss bridges using BP network. Natural frequencies and mode shapes were adopted as input variables. Numerical simulation was used to demonstrate the accuracy and efficiency. Min et al. [17] proposed an innovative ANNs-based pattern analysis tool which can identify damage-sensitive frequency and realize the identification of damage type and severity. However, ANNs have some drawbacks, such as the low convergence speed, overfitting, and locally optimal solutions [20].

PSO is a population based stochastic optimization technique developed by Eberhart and Kennedy in 1995, which is a new swarm intelligence technique inspired by social behavior of bird flocking or fish schooling [21]. Comparing with other soft computing tools, PSO is more efficient and requires fewer number of function evaluations, which leads to better or the same quality of results [22]. In past several years, PSO has been successfully applied in many research and application areas. Sharafi and Elmekawy [23] proposed a novel approach for optimal design of hybrid renewable energy systems and PSO-simulation was used to solve the multiobjective optimization problem. Chen et al. [24] developed a method for reliability-based design of composite structures which was based on PSO and finite element analysis. Examples revealed that the proposed method had good stability.

In this paper, a two-stage damage identification method is proposed. Firstly, the damage location can be detected through modal flexibility indices. Secondly, ANNs are optimized by PSO for damage severity identification and the modal flexibility changes are treated as input variables. Numerical simulation is used to verify its feasibility of the proposed method.

2. Theoretical Background

2.1. Modal Flexibility. For a bridge system with n degrees of freedom, the flexibility matrix F can be calculated by [25]

$$F = [f_{i,j}] = \sum_{i=1}^n \frac{\phi_i \phi_i^T}{\omega_i^2}, \quad (1)$$

where ω_i is the i th natural frequency, ϕ_i is the i th mass normalized mode shape, and $f_{i,j}$ is the modal flexibility component.

As can be seen from (1), the modal contribution to the flexibility matrix decreases rapidly as the frequency increases. Therefore, it can be obtained by the first few lower modes in practice.

The flexibility change matrix ΔF due to damage can be obtained by

$$\Delta F = F_D - F_I, \quad (2)$$

where F_D and F_I are flexibility matrices for damaged and intact structure, respectively.

For each DOF j , the maximum absolute value of elements in j th column of ΔF can be acquired as

$$\delta_j = \max_i |\Delta f_{i,j}|. \quad (3)$$

δ_j can be used to locate damage for bridge structures.

The curvature of flexibility changes can be calculated through second order central difference, which can be expressed as

$$MFC_i = \frac{\delta_{i-1} - 2\delta_i + \delta_{i+1}}{\Delta l^2}, \quad (4)$$

where Δl is the length of structural element.

2.2. Neural Network Optimized by PSO

2.2.1. Artificial Neural Networks. ANNs can process complex logic operations and achieve nonlinear mapping adaptively through learning. The process contains the forward propagation and back propagation. Neural network with three layers is shown in Figure 1.

A three-layer network typically consists of an input layer, a hidden layer, and an output layer. Each neuron possesses a bias, a transfer function, and an output. Neurons in both previous and subsequent layers are connected with each other. The most widely used learning algorithm is back propagation (BP) algorithm, which is a supervised learning style. In this algorithm, data are transferred forward, while errors are backward. In the process of forward, input information can be calculated through the hidden layers and finally delivered to the output layers. If the calculated results are not consistent with the expected one, then back propagation begins. Weights and bias are adjusted by prediction errors. Thereby, output results continue close to the expected ones.

2.2.2. Particle Swarm Optimization. In PSO algorithm, each solution is based on concept of the bird flock and is referred

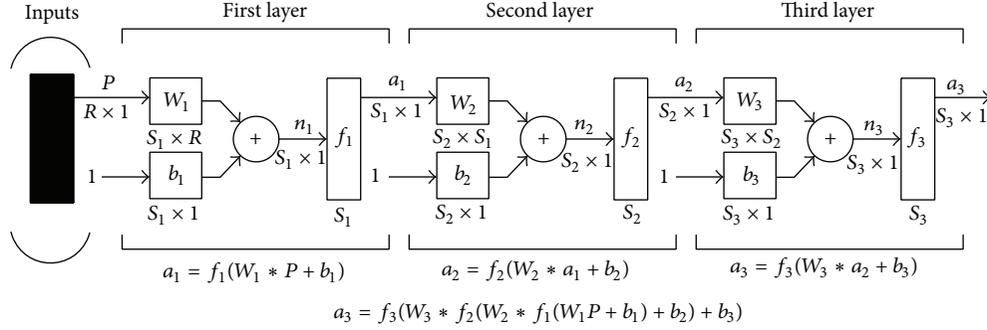


FIGURE 1: Structural diagram of ANNs.

to as a particle. In this framework the birds, besides having individual intelligence, also develop some social behavior and coordinate their movement towards a destination [22, 23, 26, 27].

Overall scale of particle swarm is assumed to be n , and the vector of coordinate position for each particle in D -dimension space can be expressed by

$$\vec{X}_i = (x_{i1}, x_{i2}, \dots, x_{id}, \dots, x_{iD}). \quad (5)$$

The velocity vector can be denoted by

$$\vec{V}_i = (v_{i1}, v_{i2}, \dots, v_{id}, \dots, v_{iD}). \quad (6)$$

And the best position of individual particle is

$$\vec{P}_i = (p_{i1}, p_{i2}, \dots, p_{id}, \dots, p_{iD}). \quad (7)$$

While the best position for particles swarm can be expressed by

$$\vec{P}_g = (p_{g1}, p_{g2}, \dots, p_{gd}, \dots, p_{gD}). \quad (8)$$

The iteration of best position for individual particle can be realized by

$$\begin{aligned} P_{i,t+1}^d &= x_{i,t+1}^d, & f(X_{i,t+1}) < f(P_{i,t}), \\ P_{i,t+1}^d &= P_{i,t}^d, & f(X_{i,t+1}) \geq f(P_{i,t}). \end{aligned} \quad (9)$$

The optimal position for particle swarm is the best one for individual particle, and the iteration of velocity and position can be calculated by

$$\begin{aligned} v_{i,t+1}^d &= v_{i,t+1}^d + c_1 * \text{rand} * (p_{i,t}^d - x_{i,t+1}^d) \\ &+ c_2 * \text{rand} * (p_{g,t}^d - x_{i,t+1}^d), \\ x_{i,t+1}^d &= x_{i,t}^d + v_{i,t+1}^d. \end{aligned} \quad (10)$$

The calculation flowchart for PSO algorithm is shown in Figure 2.

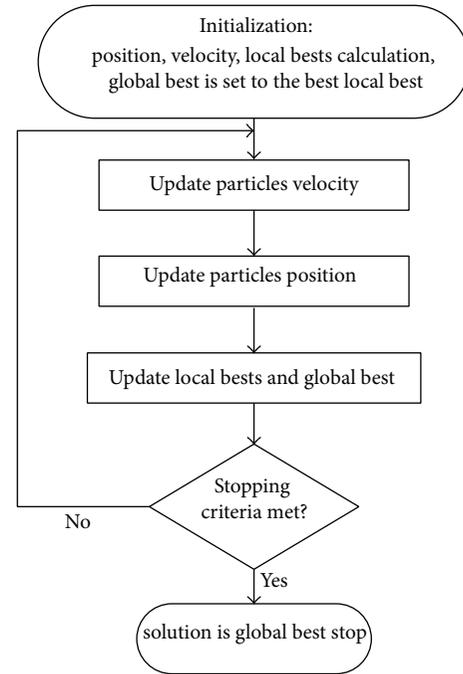


FIGURE 2: PSO algorithm flowchart.

2.3. PSO-BP Hybrid Algorithm. ANNs suffer from slow convergence and getting stuck in local minima. Therefore, PSO is used to optimize ANNs in order to achieve the combination between global optimization of PSO and instructive search of ANNs. The hybrid algorithm can not only avoid the local convergence phenomenon but also improve the performance of networks [28]. The bias and weight can be adjusted through the fitness function of PSO algorithm, and its optimization process is shown in Figure 3.

3. Numerical Simulation

3.1. Modeling of Bridge. Simply supported bridge with rectangular cross-section as shown in Figure 4 is adopted as numerical model to verify effectiveness of the proposed method. The length L is 20 m, the sectional width B is 0.8 m, and height

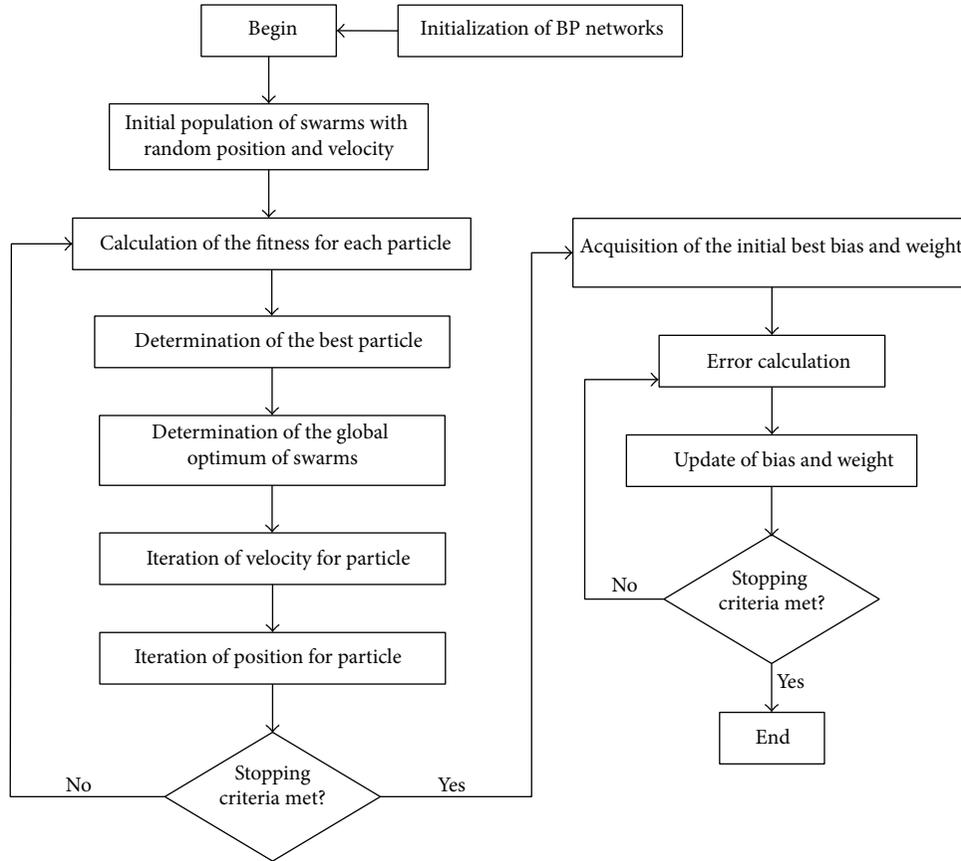


FIGURE 3: Flowchart of ANNs optimized by PSO.

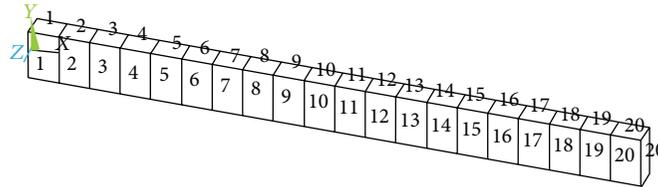


FIGURE 4: Simply supported beam bridge.

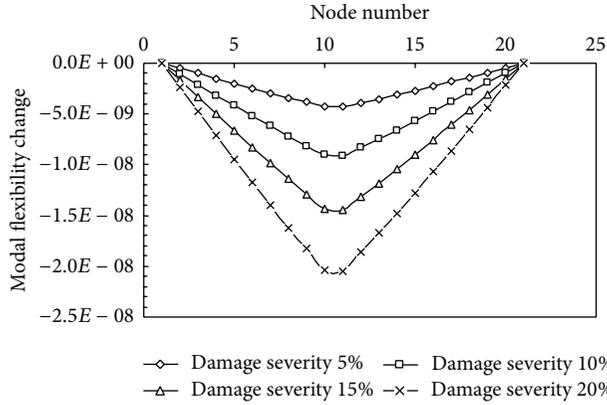
H is 1.7 m. The model is divided into 20 elements, the elastic modulus $E = 3.25e10$ N/m², Poisson's ratio $\mu = 0.167$, and the material density $\rho = 2600$ kg/m³.

In numerical simulation, damage of structure is represented by reduction in element stiffness. In this paper, the identifications with single damage location and multiple locations are conducted, respectively. As for damage identification with single location, stiffness of element 10 is assumed to drop 5%, 10%, 15%, and 20%. As for the multiple one, stiffnesses of elements 5 and 8 drop 5%, 10%, 15%, and 20% simultaneously. Natural frequencies and mode shapes for damaged and intact structure can be obtained through Lanczos modal analysis method, and the first four vertical bending modes are calculated. Natural frequencies for undamaged structure are listed in Table 1.

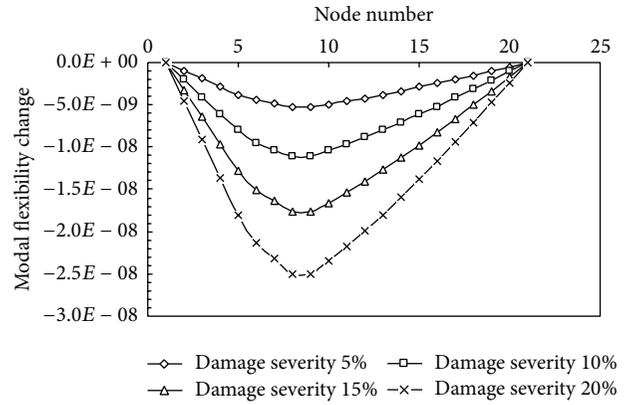
TABLE 1: Natural frequencies for undamaged structure.

Modal order	1	2	3	4
Natural frequencies (Hz)	6.7982	26.955	59.788	104.26

3.2. Damage Localization Based on Modal Flexibility Indicators. The modal flexibility matrices for structure before and after damage can be calculated by natural frequencies and mode shapes according to (1). And modal flexibility changes can be obtained by (2). The maximum absolute values of elements in each column of matrices of modal flexibility changes are displayed for damage localization. In order to illustrate the relationship between number of modes and identification effects, modal flexibilities are calculated by the first order mode and first four-order modes, respectively.

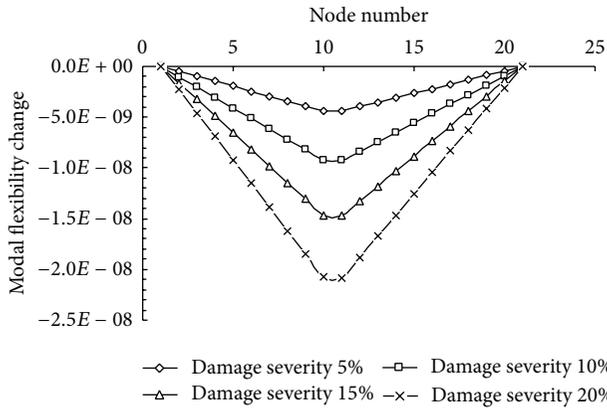


(a) Single damage location (element 10)

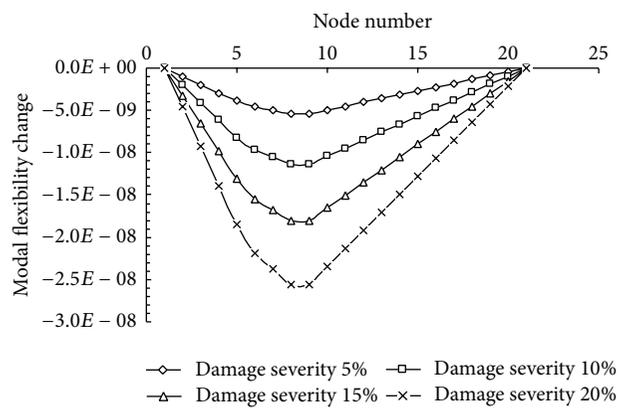


(b) Multiple damage locations (elements 5 and 8)

FIGURE 5: Damage localization based on modal flexibility changes using the first order mode.



(a) Single damage location (element 10)



(b) Multiple damage locations (elements 5 and 8)

FIGURE 6: Damage localization based on modal flexibility changes using the first four-order modes.

3.2.1. *Modal Flexibility Changes.* The damage localization results of modal flexibility changes using the first order mode are shown in Figure 5.

Damage localization results of modal flexibility changes using the first four-order modes are shown in Figure 6.

As can be seen from Figures 5 and 6, modal flexibility changes can identify the damage presence but cannot be able to identify the accurate damage locations. Curves calculated by the first order mode are consistent with that calculated by the first four-order modes.

3.2.2. *Curvature of Modal Flexibility Changes.* Damage localization results for curvature of modal flexibility changes using the first order mode are shown in Figure 7.

Damage localization results for curvature of modal flexibility changes using the first four-order modes are shown in Figure 8.

As can be seen from Figures 7 and 8, curvature of modal flexibility changes calculated by the first order mode and the first four-order modes can identify the damage locations and also qualitatively determine the damage severity. Curvature

curves obtained by the first four-order modes are smoother than that by the first order mode, and oscillation at undamaged locations is smaller. However, it is sufficient for damage localization based on the curvature calculated by the first order mode, and it is more feasible in practical applications.

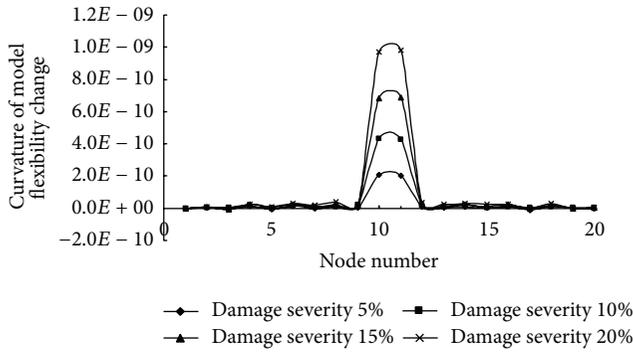
3.3. Damage Severity Identification Based on Neural Network Optimized by PSO

3.3.1. *Determination of Damage Indicator.* Considering the simplicity of modal flexibility changes, the normalized vectors are used as damage indicators and inputs of neural networks optimized by PSO. The normalized vector can be calculated by

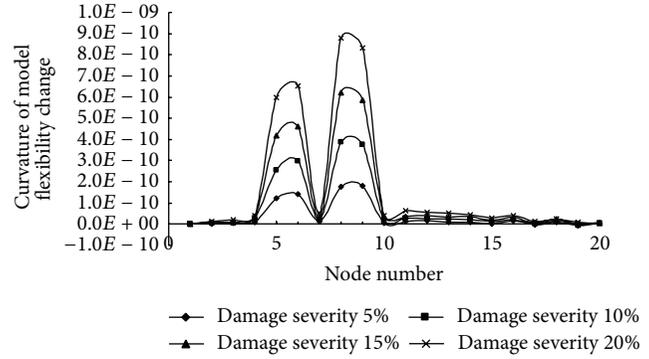
$$\text{Input} = \{\overline{\delta}_1, \overline{\delta}_2, \dots, \overline{\delta}_n\},$$

$$\overline{\delta}_j = \frac{\delta_j - \delta_{\min}}{\delta_{\max} - \delta_{\min}}, \quad (11)$$

where δ_j is the maximum absolute value of elements in j th column of ΔF ; δ_{\max} and δ_{\min} are maximum and minimum

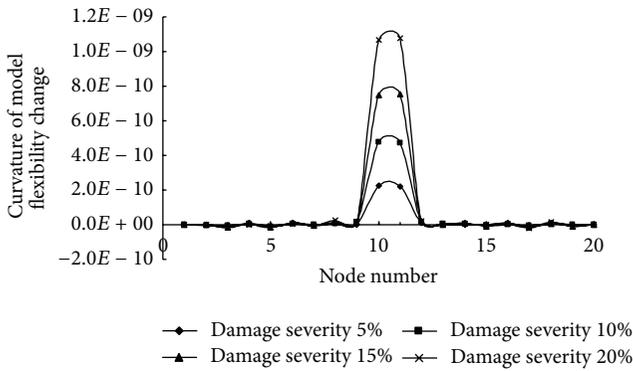


(a) Single damage location (element 10)

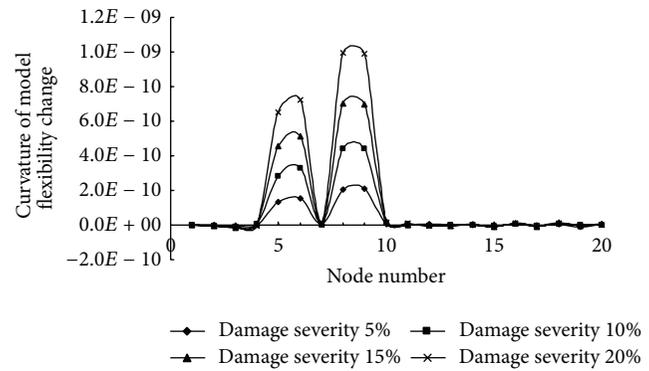


(b) Multiple damage locations (elements 5 and 8)

FIGURE 7: Damage localization based on curvature of modal flexibility changes using the first order mode.



(a) Single damage location (element 10)



(b) Multiple damage locations (elements 5 and 8)

FIGURE 8: Damage localization based on curvature of modal flexibility changes using the first four-order modes.

values of vector δ ($j = 1, 2, \dots, n$), respectively; $\bar{\delta}_j$ is the normalized δ_j .

3.3.2. Severity Identification with Single Damage Location. Taking the damage identification of element 10, for example, damage severities with 5%, 10%, 15%, and 20% are selected as training samples, while 7%, 12%, and 18% are testing ones. The normalized vectors of modal flexibility changes are used as input variables of neural network optimized by PSO. The training samples for damage identification are listed in Table 2.

Testing samples are used to verify the feasibility of optimized neural network, and the identification results are listed in Table 3.

As can be seen from Table 3, the maximum relative errors for neural network optimized by PSO are 2.57%. It reveals that the identification accuracy is favorable and can meet the practical requirements. The neural network optimized by PSO is suitable for damage severity identification with single damage location of bridge.

3.3.3. Severity Identification with Multiple Damage Locations. Taking damage identification of elements 5 and 8,

for example, damage severities simultaneously with 5%, 10%, 15%, and 20% are used as training samples, and simultaneously 7%, 12%, and 18% are treated as testing ones. The corresponding training samples are listed in Table 4, and the identification results for testing samples are shown in Table 5.

As can be seen from Table 5, the maximum relative error is 6.22%. It is larger than that of identification with single damage location. The reasons are that it is more complex for the severities identification with multiple damage locations and also associated with the number and rationality of training samples.

3.3.4. Comparative Analysis with Traditional BP Neural Network. In order to verify the superiority of the proposed method, comparative analysis between PSO-BP and traditional network is conducted. The identification results for severity identification with single and multiple damage locations are listed in Table 6.

As can be seen from Table 6, the maximum relative errors of BP neural network for severities identification with single and multiple locations are 4.00% and 12.00%, respectively. Comparative results show that the calculation accuracy of neural network optimized by PSO is better than traditional BP network.

TABLE 2: Training samples for damage severity identification of element 10.

Node number	Damage severity (%)			
	5	10	15	20
2	0.01	0.01	0.01	0.01
3	0.14	0.14	0.14	0.14
4	0.27	0.27	0.27	0.27
5	0.40	0.40	0.40	0.40
6	0.52	0.52	0.52	0.52
7	0.64	0.64	0.64	0.64
8	0.77	0.76	0.76	0.76
9	0.88	0.88	0.88	0.88
10	0.99	0.99	0.99	0.99
11	1.00	1.00	1.00	1.00
12	0.90	0.90	0.90	0.90
13	0.80	0.80	0.80	0.80
14	0.69	0.69	0.69	0.69
15	0.58	0.58	0.58	0.58
16	0.47	0.47	0.47	0.47
17	0.35	0.35	0.35	0.35
18	0.24	0.24	0.24	0.24
19	0.12	0.12	0.12	0.12
20	0.00	0.00	0.00	0.00

TABLE 3: Damage severity identification results for testing samples of element 10.

Expected outputs (%)	Identification results (%)	Relative errors (%)
7	7.18	2.57
12	12.28	2.28
18	18.35	1.94

4. Conclusions

Damage identification is one of the major challenges in bridge health monitoring. We have proposed a two-stage strategy based on modal flexibility and neural network optimized by PSO for damage location and severity identification. Simply supported bridge with rectangular cross-section is selected as numerical model to verify the effectiveness of the proposed method.

As for damage localization based on modal flexibility indicators, modal flexibility changes can identify the damage presence but cannot achieve acceptable identification of damage locations. The curvature of flexibility changes is more favorable for damage localizing and it is sufficient to be calculated by the first order mode.

With regard to the damage severity identification of neural network optimized by PSO, normalized vector of modal flexibility changes is adopted as input variable. Training samples for identification with single and multiple damage locations are constructed, respectively. The identification results for single damage location reveal that the maximum relative error is 2.57%, while it is 6.22% for multiple damage

TABLE 4: Training samples for damage severity identification of elements 5 and 8.

Node number	Damage severity (%)			
	5	10	15	20
2	0.10	0.10	0.10	0.10
3	0.30	0.30	0.30	0.30
4	0.50	0.50	0.50	0.50
5	0.69	0.69	0.69	0.69
6	0.83	0.83	0.83	0.83
7	0.92	0.92	0.92	0.92
8	1.00	1.00	1.00	1.00
9	1.00	1.00	1.00	1.00
10	0.93	0.93	0.93	0.93
11	0.85	0.85	0.85	0.85
12	0.78	0.77	0.77	0.77
13	0.69	0.69	0.69	0.69
14	0.60	0.60	0.60	0.60
15	0.50	0.51	0.51	0.51
16	0.41	0.41	0.41	0.41
17	0.31	0.31	0.31	0.31
18	0.21	0.21	0.21	0.21
19	0.10	0.10	0.10	0.10
20	0.00	0.00	0.00	0.00

TABLE 5: Damage severity identification results for testing samples of elements 5 and 8.

Expected outputs (%)	Identification results (%)	Relative errors (%)
7, 7	6.57, 7.35	6.14, 5.00
12, 12	11.38, 12.68	5.17, 5.67
18, 18	19.12, 18.95	6.22, 5.28

TABLE 6: Damage identification using BP network.

Damage category	Single damage location	Multiple damage locations
Expected outputs (%)	7 12 18	7, 7 12, 12 18, 18
Actual results (%)	7.28 12.38 17.59	7.84, 7.79 11.13, 12.89 16.83, 16.98
Relative errors (%)	4.00 3.17 2.28	12.00, 11.29 7.25, 7.42 6.50, 5.67

locations. The results manifest that it is feasible for damage severity identification using neural network optimized by PSO. Comparative analysis with traditional BP network is conducted to verify its superiority. The calculation accuracy of optimized neural network is more favorable.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors express their appreciation for the financial support of National Natural Science Foundation of China under Grant nos. 51378236 and 51278222 and “985 Project” of Jilin University.

References

- [1] N. Kulprapha and P. Warnitchai, “Structural health monitoring of continuous prestressed concrete bridges using ambient thermal responses,” *Engineering Structures*, vol. 40, pp. 20–38, 2012.
- [2] J. Kim, J. Park, and B. Lee, “Vibration-based damage monitoring in model plate-girder bridges under uncertain temperature conditions,” *Engineering Structures*, vol. 29, no. 7, pp. 1354–1365, 2007.
- [3] S. C. Mohan, D. K. Maiti, and D. Maity, “Structural damage assessment using FRF employing particle swarm optimization,” *Applied Mathematics and Computation*, vol. 219, no. 20, pp. 10387–10400, 2013.
- [4] U. Dackermann, *Vibration-based damage identification methods for civil engineering structures using artificial neural networks [Ph.D. thesis]*, Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia, 2010.
- [5] B. H. Kim, T. Park, and G. Z. Voyiadjis, “Damage estimation on beam-like structures using the multi-resolution analysis,” *International Journal of Solids and Structures*, vol. 43, no. 14-15, pp. 4238–4257, 2006.
- [6] S. W. Doebling, C. R. Farrar, and M. B. Prime, “A summary review of vibration-based damage identification methods,” *Shock and Vibration Digest*, vol. 30, no. 2, pp. 91–105, 1998.
- [7] Y. J. Yan, L. Cheng, Z. Y. Wu, and L. H. Yam, “Development in vibration-based structural damage detection technique,” *Mechanical Systems and Signal Processing*, vol. 21, no. 5, pp. 2198–2211, 2007.
- [8] P. Cawley and R. D. Adams, “The location of defects in structures from measurements of natural frequencies,” *The Journal of Strain Analysis for Engineering Design*, vol. 14, no. 2, pp. 49–57, 1979.
- [9] A. K. Pandey and M. Biswas, “Damage detection in structures using changes in flexibility,” *Journal of Sound and Vibration*, vol. 169, no. 1, pp. 3–17, 1994.
- [10] G. Li, K. Hao, Y. Lu, and S. Chen, “A flexibility approach for damage identification of cantilever-type structures with bending and shear deformation,” *Computers and Structures*, vol. 73, no. 6, pp. 565–572, 1999.
- [11] L. T. Stutz, D. A. Castello, and F. A. Rochinha, “A flexibility-based continuum damage identification approach,” *Journal of Sound and Vibration*, vol. 279, no. 3–5, pp. 641–647, 2005.
- [12] E. Reynders and G. de Roeck, “A local flexibility method for vibration-based damage localization and quantification,” *Journal of Sound and Vibration*, vol. 329, no. 12, pp. 2367–2383, 2010.
- [13] F. N. Catbas, M. Gul, and J. L. Burkett, “Damage assessment using flexibility and flexibility-based curvature for structural health monitoring,” *Smart Materials and Structures*, vol. 17, no. 1, pp. 1–12, 2008.
- [14] J. J. Lee, J. W. Lee, J. H. Yi, C. B. Yun, and H. Y. Jung, “Neural networks-based damage detection for bridges considering errors in baseline finite element models,” *Journal of Sound and Vibration*, vol. 280, no. 3–5, pp. 555–578, 2005.
- [15] M. Vakil-Baghmisheh, M. Peimani, M. H. Sadeghi, and M. M. Etefagh, “Crack detection in beam-like structures using genetic algorithms,” *Applied Soft Computing Journal*, vol. 8, no. 2, pp. 1150–1160, 2008.
- [16] F. Kang, J. Li, and Q. Xu, “Damage detection based on improved particle swarm optimization using vibration data,” *Applied Soft Computing Journal*, vol. 12, no. 8, pp. 2329–2335, 2012.
- [17] J. Min, S. Park, C. Yun, C. Lee, and C. Lee, “Impedance-based structural health monitoring incorporating neural network technique for identification of damage type and severity,” *Engineering Structures*, vol. 39, pp. 210–220, 2012.
- [18] Y. B. Jiao, H. B. Liu, Y. C. Cheng, and X. Q. Wang, “Fuzzy neural network-based damage assessment of bridge under temperature effect,” *Mathematical Problems in Engineering*, vol. 2014, Article ID 418040, 9 pages, 2014.
- [19] M. Mehrjoo, N. Khaji, H. Moharrami, and A. Bahreinnejad, “Damage detection of truss bridge joints using Artificial Neural Networks,” *Expert Systems with Applications*, vol. 35, no. 3, pp. 1122–1131, 2008.
- [20] R. Ganguli, “A fuzzy logic system for ground based structural health monitoring of a helicopter rotor using modal data,” *Journal of Intelligent Material Systems and Structures*, vol. 12, no. 6, pp. 397–407, 2001.
- [21] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [22] M. A. Mohandes, “Modeling global solar radiation using Particle Swarm Optimization (PSO),” *Solar Energy*, vol. 86, no. 11, pp. 3137–3145, 2012.
- [23] M. Sharafi and T. Y. Elmekawy, “Multi-objective optimal design of hybrid renewable energy systems using PSO-simulation based approach,” *Renewable Energy*, vol. 68, pp. 67–79, 2014.
- [24] J. Q. Chen, Y. F. Tang, R. Ge, Q. L. An, and X. W. Guo, “Reliability design optimization of composite structures based on PSO together with FEA,” *Chinese Journal of Aeronautics*, vol. 26, no. 2, pp. 343–349, 2013.
- [25] S. Kazemi, A. Fooladi, and A. R. Rahai, “Implementation of the modal flexibility variation to fault identification in thin plates,” *Acta Astronautica*, vol. 66, no. 3-4, pp. 414–426, 2010.
- [26] O. Begambre and J. E. Laier, “A hybrid Particle Swarm Optimization—Simplex algorithm (PSOS) for structural damage identification,” *Advances in Engineering Software*, vol. 40, no. 9, pp. 883–891, 2009.
- [27] A. Khare and S. Rangnekar, “A review of particle swarm optimization and its applications,” *Applied Soft Computing*, vol. 13, pp. 2997–3006, 2013.
- [28] H. S. Wang, Y. N. Wang, and Y. C. Wang, “Cost estimation of plastic injection molding parts through integration of PSO and BP neural network,” *Expert Systems with Applications*, vol. 40, no. 2, pp. 418–428, 2013.

Research Article

A Swarm Random Walk Based Method for the Standard Cell Placement Problem

Najwa Altwaijry and Mohamed El Bachir Menai

Department of Computer Science, College of Computer and Information Sciences, King Saud University,
P.O. Box 51178, Riyadh 11453, Saudi Arabia

Correspondence should be addressed to Najwa Altwaijry; ntwaijry@ksu.edu.sa

Received 12 March 2014; Accepted 19 July 2014; Published 13 August 2014

Academic Editor: Baozhen Yao

Copyright © 2014 N. Altwaijry and M. E. B. Menai. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The standard cell placement (SCP) problem is a well-studied placement problem, as it is an important step in the VLSI design process. In SCP, cells are placed on chip to optimize some objectives, such as wirelength or area. The SCP problem is solved using mainly four basic methods: simulated annealing, quadratic placement, min-cut placement, and force-directed placement. These methods are adequate for small chip sizes. Nowadays, chip sizes are very large, and hence, hybrid methods are employed to solve the SCP problem instead of the original methods by themselves. This paper presents a new hybrid method for the SCP problem using a swarm intelligence-based (SI) method, called SwarmRW (swarm random walk), on top of a min-cut based partitioner. The resulting placer, called sPL (swarm placer), was tested on the PEKU benchmark suite and compared with several related placers. The obtained results demonstrate the effectiveness of the proposed approach and show that sPL can achieve competitive performance.

1. Introduction

The placement problem has been studied extensively in the past decades, and it is a crucial problem in VLSI computer aided design. The object of this study is standard cell placement, one of the methods of placement. Standard cells are logic modules that have a predesigned internal layout. Standard cells are of a fixed height and varying widths, because of the different functionalities of the modules, and are laid out in rows on chip. Figure 1 shows an example of a standard cell layout style with whitespace in between, shown in gray [1]. Logic inputs and outputs are available at pins (or terminals) along the top and bottom edges of the cells, and wires that connect standard cells and pads pass through the whitespace and the area between rows of standard cells.

Primary objectives when placing cells on chip include wirelength minimization, where wirelength is defined as the sum of the total estimated interconnection wirelengths, and area minimization. Other objectives such as power optimization and timing optimization are common. The SCP problem is defined as follows. Given an electrical circuit consisting of standard cells with predefined input and output terminals and interconnected in a predefined way, construct

a layout indicating the positions of the cells so that objectives are optimized. The inputs to the problem are the descriptions of the standard cells and the netlist. The standard cell description consists of the shapes, sizes, and terminal locations. The netlist describes the interconnections between the terminals of the cells. The output is a list of x - and y -coordinates for all standard cells. More formally, it can be stated as follows. Given a set of n standard cells $C = \{c_1, c_2, \dots, c_n\}$, a set of k signals $S = \{s_1, s_2, \dots, s_k\}$, where each signal s_i is associated with a subset of cells C_{s_i} , where $C_{s_i} = \{c_j \mid s_i \in S\}$, and a set of locations $L = \{l_1, l_2, \dots, l_p\}$, where $p \geq n$. The SCP problem asks to assign each cell $c_i \in C$ to a unique location l_j , such that the objectives are minimized (or maximized) subject to constraints. Our chosen objective is to minimize the total wirelength TWL:

$$\text{TWL} = \sum_{i=1}^k \text{length}(s_i), \quad (1)$$

where $\text{length}(s_i)$ is the length of the net associated with signal s_i .

The SCP problem is an NP-hard problem [1, 2]. Current chip sizes may contain millions of movable objects, and,

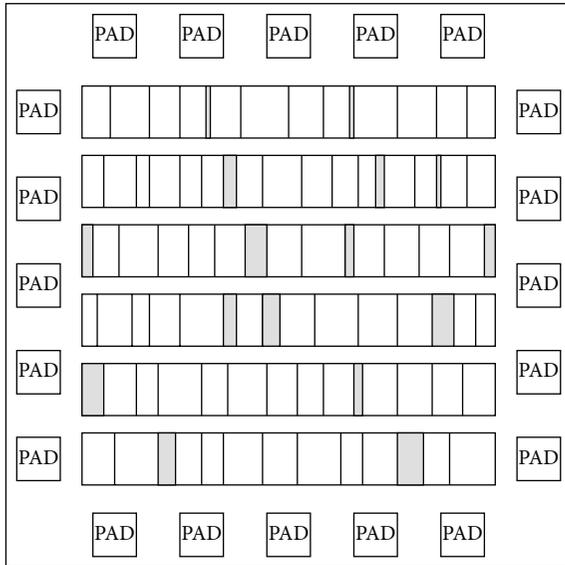


FIGURE 1: Standard cell layout style.

as such, the problem cannot be solved exactly in polynomial time. For that reason, a metaheuristic algorithm is used to search the large solution space of the problem. Previously, methods used to solve the SCP problem were broadly categorized into iterative placement methods and constructive placement methods. Iterative placement methods can rarely create a good placement from a randomly generated starting placement, especially in chips with large numbers of cells, as well as having runtime requirements significantly larger than those required by constructive placement methods. Iterative placement methods include the simulated annealing and the genetic algorithm. Constructive placement algorithms generally lack accuracy on placement objectives such as wirelength [3], and they include the min-cut algorithm, the force-directed algorithm, and the quadratic placement algorithm. A combined approach is more likely to yield a successful placement [3].

In this paper, we propose a novel hybrid method to solve the SCP problem based on the algorithm swarm random walk (SwarmRW) [4] together with hMetis, a min-cut based partitioner [5, 6], resulting in a placer called sPL (swarm placer). The objective function chosen in this work is wirelength minimization. In some cases, secondary performance measures are adopted, such as minimizing the length of some critical nets. The cost of total wirelength may increase, as these objectives are generally conflicting. Other objectives include power consumption and routability. Weighted total wirelength is a convenient representative because it can be optimized efficiently, and with some net reweighting, it can be used to represent other objectives [7].

Wirelength is estimated by the semiperimeter method, in which the half-perimeter of the bounding rectangle of all pins in the net is taken to be the wirelength of the net. Experiments were conducted on the PEKU benchmark [8], which contains several sets of placement examples with known upper bound of wirelength, containing local nets as well as nonlocal nets. sPL was compared against other placers including Dragon

[9], a proven and well-known placer based on partitioning and simulated annealing, and abcPL [10], a placer based on an artificial bee colony (ABC) algorithm [11]. The results obtained demonstrate the competitiveness of the proposed approach and show particularly that sPL outperforms abcPL.

The rest of this paper is organized as follows. Section 2 reviews some of the most prominent works in cell placement. Section 3 describes the general SwarmRW algorithm. Section 4 describes the sPL placer. Section 5 shows the experimental results obtained by sPL and discusses its performance compared with others. Finally, Section 6 concludes this work and outlines some future research directions.

2. Related Work

As mentioned previously, SCP algorithms can be broadly categorized into constructive placement methods and iterative placement methods. These methods, by themselves, are rarely used today. Instead, hybridization of these methods, with some advanced heuristics, is used to solve the SCP problem.

QPlace [12] is an industrial placer by Cadence Design Systems, Inc., but a detailed description is unavailable to the authors as it has been discontinued. mPG [13] is a multilevel and simulated annealing placement algorithm with integrated global routing to update and optimize the cost of congestion. Global routing is done via a fast routing and incremental A-tree algorithm. Congestion is highlighted because overly congested areas force connections to make detours or to change layers, thereby increasing the actual wirelength and reducing performance.

mPL [14, 15] is an efficient multilevel placer that uses nonlinear programming to minimize quadratic wirelength. It uses recursive first choice clustering in the coarsening phase. A custom nonlinear programming solver using the interior point method is used at mPL's coarsest level to obtain an initial solution. Relaxation is restricted to sweeps of local refinements and spreading out of cells.

Dragon [9, 16] is a top-down hierarchical placement tool that deploys min-cut based partitioning and simulated annealing to place designs with thousands of macroblocks and millions of standard cells. Dragon emphasizes congestion during placement, in addition to traditional wirelength minimization. Congestion is a main objective during routing; highly congested regions necessitate routing detours around the site, leading to increased wirelength, and in the worst case, the placement is unroutable. In order to reduce congestion, it can be included in the cost function of simulated annealing, by combining a regional router into the placement, or in a postprocessing step. Dragon uses white space to improve routability dynamically during placement. In addition, two methods to control target utilization are incorporated into Dragon. Target utilization is a metric that estimates the capability of the placement tool to generate routable designs.

CAPO [17] is a placer that implements a top-down cut-size driven recursive bipartitioning and uses the multilevel hypergraph partitioner MLpart [18] and some time-limited branch-and-bound heuristics. It has some enhancements to avoid the corking effects caused by balance constraints, as well as terminal propagation.

abcPL [10] is a top-down hierarchical placer that uses hMetis [5, 6], a min-cut based partitioner, and the ABC algorithm [11] to place a chip, minimizing wirelength. These methods have all been tested on the PEKU benchmark suite [8], and the wirelengths produced can be 2.98 times the optimal in the worst case.

There remains significant room for improvement in existing placement algorithms, suggesting that new—more scalable and stable—hybrid techniques may be needed for future generations. There are other methods that have been used to solve the SCP problem, using genetic algorithms (parallel genetic algorithm (PGA) [19] and genetic algorithm for placement (GAP) [20]), simulated evolution (force-directed simulated evolution [21] and distributed parallelized SE algorithm (SimE) [22]) in addition to some hybridized methods (parallel simulated annealing/genetic algorithm (PSAGA) [23] and SimE-GA [24]); however, these methods are not as competitive, being tested on older benchmarks. Although simulated evolution was used to solve instances of the PEKO suite in [25], it is not included in the experimental results as it was uncompetitive, with quality ratios ranging from 6.33 to 8.52.

3. SwarmRW Algorithm

SwarmRW [4] is a new SI algorithm. It employs a swarm of potential solutions that cooperate or “learn” from each other one dimension at a time, to optimize each swarm member. First, a swarm of initial solutions is randomly generated, and each individual then generates a new solution. Individuals learn from each other the best values, or positions, and change themselves accordingly: each individual will “imitate” a randomly selected neighbor in a single randomly selected dimension at a time, and if this improves the fitness of the individual, the change will be accepted; otherwise, it will be rejected. Intuitively, changing the solution in a single dimension at a time allows the search process to better judge if the proposed change is beneficial.

For an optimization problem with d dimensions, a swarm of n initial solutions $\mathbf{x}_i = (x_{i1}, \dots, x_{id})$, $1 \leq i \leq n$, is randomly generated. A change to \mathbf{x}_i is made in a single dimension j , and a greedy selection is performed. If this move results in a better fitness value, the change is accepted. However, if the move decreases fitness, the move is rejected. The change made to x_{ij} is done through swarm cooperation. A random solution \mathbf{x}_k , $1 \leq k \leq n$ and $k \neq i$, is selected, and solution \mathbf{x}_i imitates solution \mathbf{x}_k in dimension j .

SwarmRW changes a solution in only one dimension j at a time t , using a simple formula:

$$x_l^{t+1} = \begin{cases} x_l^t + \rho * (x_l^t - y_l^t), & l = j, \\ x_l^t, & l \neq j, \end{cases} \quad (2)$$

where j is a randomly selected dimension, and \mathbf{y} is a randomly selected solution, distinct from \mathbf{x} . The parameter ρ is a randomly generated number that describes how much \mathbf{x} is attracted or repelled by \mathbf{y} . This process is a random walk performed by the swarm. Algorithm 1 outlines the main steps of SwarmRW [4].

The SwarmRW method is similar to how employed bees in the ABC algorithm produce neighboring solutions. However, the main difference between SwarmRW and ABC is that no onlooker bees are employed to probabilistically improve the best solutions produced by employed bees. In addition, solutions are never abandoned and scout bees are not used to replace abandoned solutions with new solutions in the search space.

4. sPL Placer

Simulated annealing (SA) has been heavily researched for SCP and has proven to be successful (e.g., Dragon [16]). Dragon replaces the current solution by a random neighboring state solution constructed from the current solution by moving a randomly selected cell or groups of cells to a new position. The neighboring solution may be accepted or rejected with a probability that depends on the change in the objective function value and the temperature, T . The temperature is gradually decreased throughout the SA process. When T is large, the probability of accepting a solution that decreases the objective function value is high, allowing the solution to move out of local optima. As T decreases, the acceptance probability goes down, meaning that the probability of accepting inferior solutions decreases, allowing the algorithm to converge to a close-to-optimal solution. SwarmRW works similarly by moving cells or groups of cells to new positions; however, it does not employ a temperature to accept inferior solutions with a certain probability and only greedily accepts solutions that improve current solution quality. It is our belief that SwarmRW would be equally applicable to problems other than SCP, with no added benefit from its properties that are specific to the SCP problem.

sPL solves large-sized SCP problems with the objective of minimizing wirelength. sPL is implemented in a top-down hierarchical approach. While hMetis partitions cells into bins using recursive bisection with the objective of minimizing the net-cut, SwarmRW uses the wirelength objective (half-perimeter wirelength) to place the bins. Each swarm member (potential solution) contains the following information: the number of bins and their arrangement on chip, the total wirelength of the bins, in addition to the net length of each net, and the number of rows and columns on chip.

Initialization is performed as follows: an initial partition of 64 bins is created using hMetis, where half the bins are initialized randomly, while the rest are initialized sequentially (see Figures 2 and 3). Bins are of equal size and contain approximately the same number of cells, with each bin having a unique numerical identifier. Cells within the bins are placed on top of each other in the lower left corner of the bin (i.e., cells overlap in bins). Optimization of bin positions commences through SwarmRW. After a preset maximum number of iterations, or if no reduction in wirelength is noted for some iterations, SwarmRW relinquishes the bins to hMetis, which partitions them either horizontally or vertically, in turn.

The SwarmRW algorithm is converted to solve the discrete SCP problem, as it was designed for continuous domain

```

input:  $S, \max, \rho$ 
output: Best Solution
Initialize:  $S$ : swarm size,  $\max$ : maximum runs,  $\rho$ : parameter;
begin
  Initialize swarm randomly;
  for  $\text{iter} = 0 \rightarrow \max - 1$  do
    for  $i = 0 \rightarrow S - 1$  do
      Randomly select  $k, 0 \leq k < S, k \neq i$ ;
      Set  $\mathbf{y} = \mathbf{x}_k$ ;
      Randomly select dimension  $j$ ;
      Calculate  $x_{ij}^{t+1}$  by (2);
      Set  $\bar{\mathbf{x}}_i = (x_{i1}^t, \dots, x_{ij}^{t+1}, \dots, x_{id}^t)$ ;
      if  $\text{fitness}(\mathbf{x}_i) < \text{fitness}(\bar{\mathbf{x}}_i)$  then
         $\mathbf{x}_i = \bar{\mathbf{x}}_i$ ;
      end
    end
  end
end
return Best Solution

```

ALGORITHM 1: SwarmRW pseudocode.

14	10	6	2
11	7	3	15
8	4	0	12
5	1	13	9

FIGURE 2: Random initialization of bins.

3	7	11	15
2	6	10	14
1	5	9	13
0	4	8	12

FIGURE 3: Sequential initialization of bins.

problems. For example, the PSO algorithm can be converted from its continuous version into a discrete version to solve the TSP problem by using a series of swaps (swapping cities) instead of a velocity. A similar method is adopted for SwarmRW, where only a single swap can be made at a time, in keeping with updating only a single dimension at a time in the continuous version. An individual in SwarmRW will produce a nearby solution by swapping two bins, and greedy

selection is used to select the best solution from among the new and old solutions.

Bins are partitioned at the end of each SwarmRW phase, either vertically or horizontally. If the last cut was vertical, the bins are partitioned horizontally, and vice versa. The cells and nets within the bin are fed into hMetis using the appropriate format, disregarding connections to outside bins. The resulting partitions are then used to continue the optimization process.

This two-step process is repeated until each bin has 3-4 cells [9, 26]. Finally, at the end of a placement run, the bin structure is discarded, and bins are converted into rows, where the cells in each bin are placed within the bin area on chip. While placing cells, the chip width and the number of rows are taken into consideration. Columns of bins might contain a number of cells larger than the available space on chip for those columns, in which case cells are placed as close to their intended position as possible. Finally, a greedy heuristic further improves wirelength. Pseudocode for our SPL placer is shown in Algorithm 2.

4.1. Fitness Function. The fitness function uses the semiperimeter method for calculating wirelength and is calculated as follows:

$$\text{fitness} = \frac{10,000,000}{\text{wirelength} + 1}, \quad \text{wirelength} \geq 0, \quad (3)$$

where the wirelength is calculated as shown in (4) and net_i^u is calculated as in (5). Consider the following:

$$\text{wirelength} = \sum_{i=0}^{\max_{\text{nets}}-1} \text{net}_i^u, \quad (4)$$

$$\text{net}_i^u = ((X_{\max} - X_{\min}) + (Y_{\max} - Y_{\min})) + (\text{cells}_i * w), \quad (5)$$

```

input:  $S$ , max, StoppingCondition, benchmark files
output: Best Placement
Initialize:  $S$ : swarm size, max: maximum runs;
begin
  Read benchmark files;
  repeat
    Partition Bins;
    Initialize swarm;
    for count = 0  $\rightarrow$  max - 1 do
      for  $i = 0 \rightarrow S - 1$  do
        Generate a solution  $\bar{x}_i$  using a method from Section 4.2;
        if  $fitness(x_i) < fitness(\bar{x}_i)$  then
           $x_i = \bar{x}_i$ ;
        end
      end
    end
  until StoppingCondition = true;
  Remove bin structure to convert best solution  $x$  bins to rows;
  Use greedy heuristic;
  Output wirelength;
  return Best Placement
end

```

ALGORITHM 2: sPL pseudocode.

where net_i^u is the total wirelength (taking into consideration the number of cells in the net), X_{\max} is the maximum x -coordinate of all cells in the net, X_{\min} is the minimum x -coordinate, Y_{\max} is the maximum y -coordinate, Y_{\min} is the minimum y -coordinate of all cells in the net, $cells_i$ is the number of cells in net_i^u , and w is the average cell width of the benchmark cells.

4.2. Producing Neighboring Solutions. An individual produces a nearby solution by swapping two bins, and greedy selection is used to select the best solution from among the new and old solutions. To produce a nearby solution, the following methods are implemented.

- (1) Switch one bin using a neighbor's content: a bin is moved to a new position in the individual, according to that bin's position in a neighboring individual.
- (2) Randomly select two bins and switch their places.
- (3) Switch a randomly selected bin with a randomly selected nearby bin. A nearby bin is defined as a bin that is above or below the selected bin by a preset number of spaces, usually less than 8.
- (4) Switch a randomly selected bin with a bin around its perimeter. Note that bins around the edge of the chip have fewer neighbors and thus fewer bins to be switched with.

The first method is the only one in which the swarm "cooperates" to produce better solutions. The other methods were added as they are acceptable variations of changing the solution in one dimension at a time. Note that other methods could have been considered, such as subtracting one solution from another and then selecting one of the different

swaps (needed to change one solution into another) produced by such a subtraction, to perform the swap. The SwarmRW parameter ρ can be used to affect the probability of choosing one of these swaps. However, method 1 above produces one of these differences randomly without needing to perform a lengthy subtraction. Calculating the entire difference between two individuals is needed for algorithms such as PSO, where the velocity is changed over all dimensions simultaneously; however, in SwarmRW's case, it is unnecessary.

4.3. Converting Bins to Rows. Usually, the number of bin rows does not correspond to the real number of rows on chip. In the final sPL phase, bins are converted into rows. A column of bins is merged and cells are placed within the column area. Cells are placed beginning with cells contained in the bottom leftmost bin. When the first row area is placed, and if there are some remaining cells in the bin, the remaining cells are placed in the next row (the row above). When all cells in a bin are placed, the bin above it in the same column is placed in the same manner, before placing the next column of bins. If the space available to a column of bins is used and there are some unplaced cells, they are placed in the nearest empty position. However, if the number of unplaced cells is above a certain threshold, the bin is allowed extra width on chip, to allow all cells to be placed within bin boundaries.

5. Experimental Results

sPL was implemented in Java. Some experiments were performed on an Intel Xeon 3.20GHzx4 CPU, with 32 GB RAM under 64-bit Ubuntu 12.04.1, and some were performed on Intel Core 2 Duo 2.80GHzx2 CPU, with 6.8 GB RAM under 64-bit Ubuntu 12.04.1. Two experiments were performed

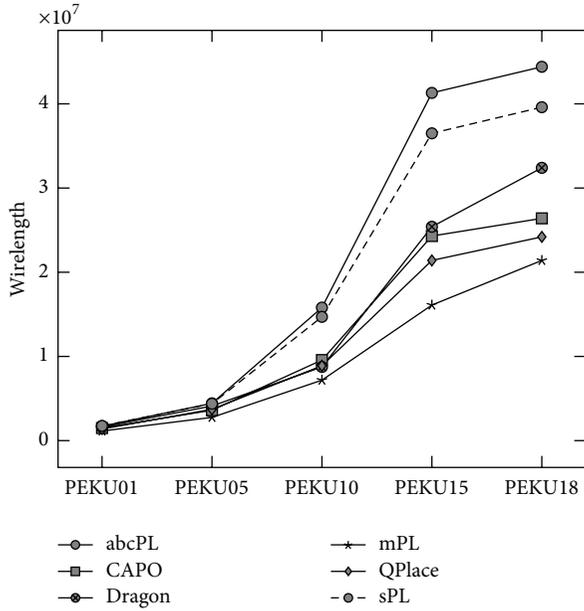


FIGURE 4: Wirelength variation on PEKU with 0.0% nonlocal nets.

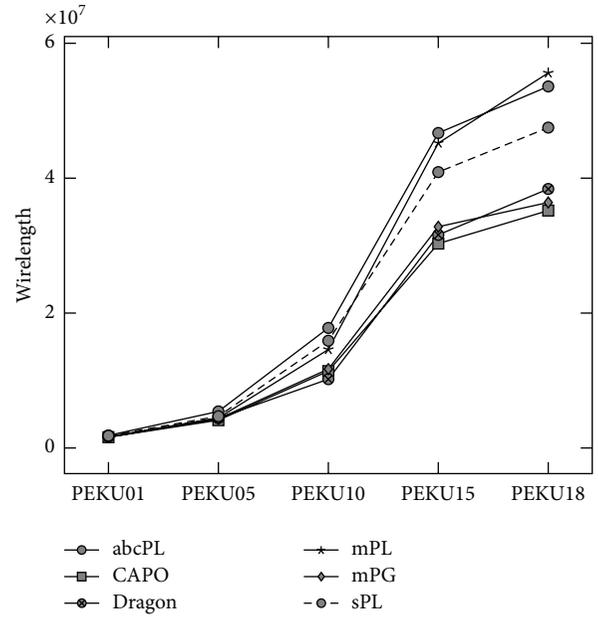


FIGURE 6: Wirelength variation on PEKU with 0.25% nonlocal nets.

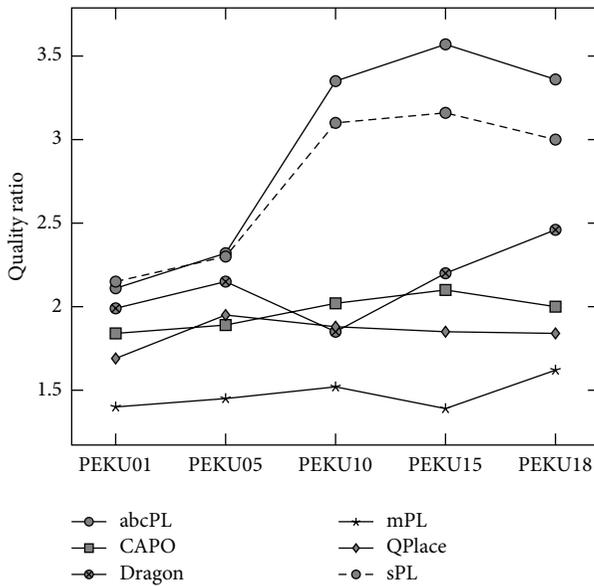


FIGURE 5: Quality ratios on PEKU with 0.0% nonlocal nets.

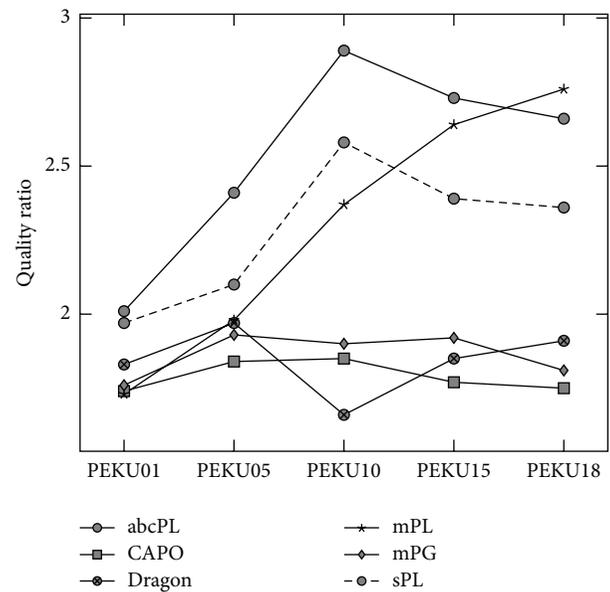


FIGURE 7: Quality ratios on PEKU with 0.25% nonlocal nets.

on the Intel Xeon in tandem. The latest version of hMetis (2.0pre1), with support for 64-bit architectures, was used. The benchmark suite selected for this study is the PEKU suite [8]. PEKU contains several sets of placement examples with a known upper bound of wirelength, containing local nets as well as nonlocal nets. There are 5 basic chip layouts in PEKU, each having 8 different percentages of nonlocal nets, for a total of 40 instances. The number of movable objects ranges from tens of thousands to hundreds of thousands of cells. As an example, PEKU01 contains 12506 cells, 14111 nets, and 113 rows. The upper bound on the wirelength is 8:14e5 units.

sPL is compared with abcPL, as well as mPL, mPG, CAPO, and Dragon. Results achieved by these placers on the selected benchmark are reported in the literature. These placers are state-of-the-art academic placers and have achieved the best reported results on our chosen benchmark. The other methods reported in the related work are not as competitive, being tested on older benchmarks.

An initial tuning of the algorithm's parameters on PEKU01 showed that varying the swarm size from 3 to 10 resulted in an average increase of 3% in the performance of sPL. However, the required runtime also increased by nearly

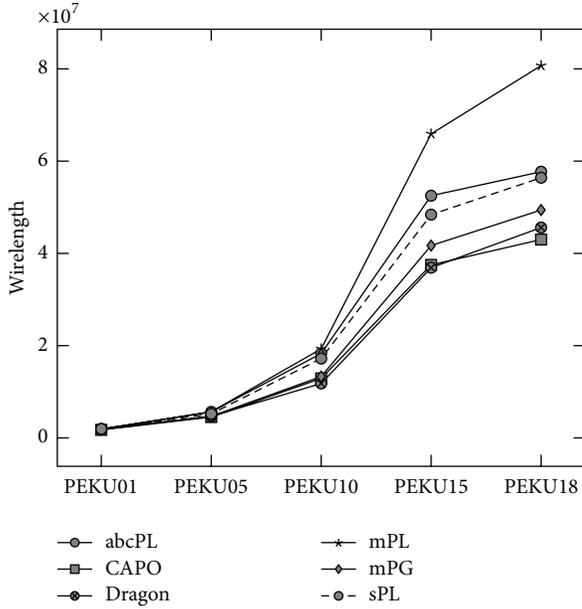


FIGURE 8: Wirelength variation on PEKU with 0.50% nonlocal nets.

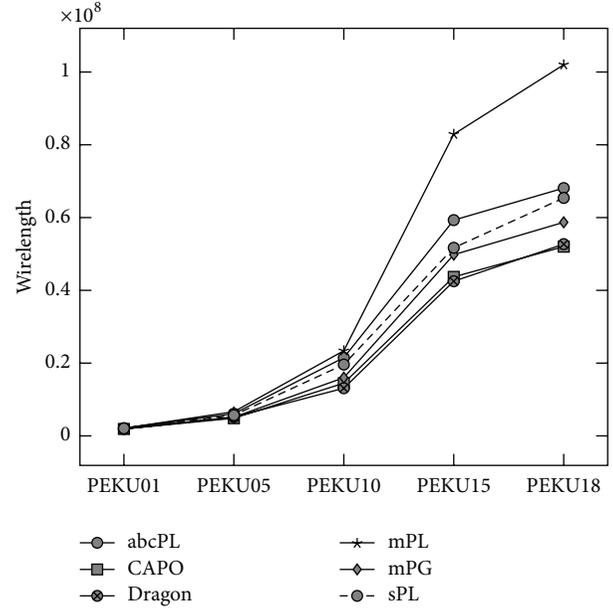


FIGURE 10: Wirelength variation on PEKU with 0.75% nonlocal nets.

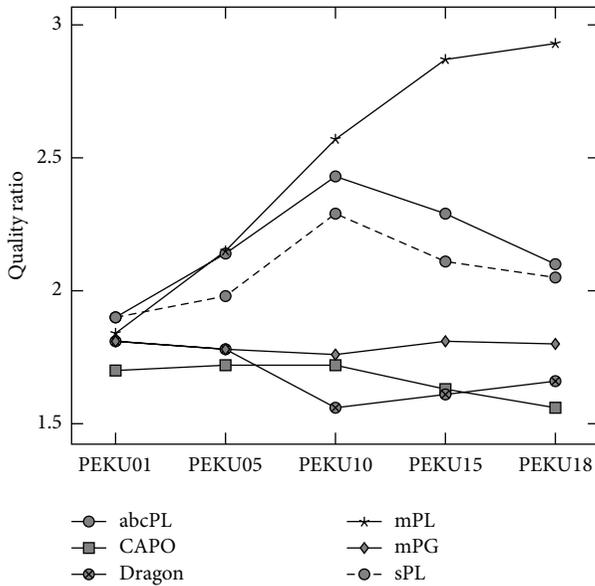


FIGURE 9: Quality ratios on PEKU with 0.50% nonlocal nets.

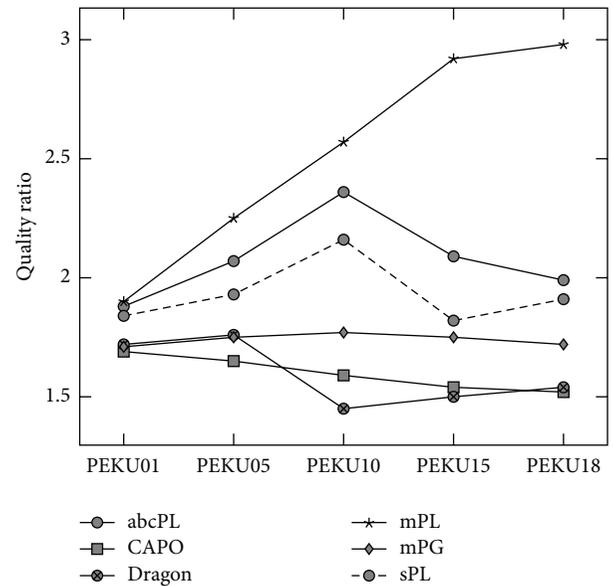


FIGURE 11: Quality ratios on PEKU with 0.75% nonlocal nets.

334%. This prohibitive amount of runtime constrained us to use a colony size of 3. The maximum number of cycles allowed to improve individuals in the swarm is set to 70000. Individuals are optimized in each partitioning step for this maximum, and if no improvement is detected for a period of 10000 cycles, then an early exit strategy is enforced, moving the placement into the next partitioning phase. Partitioning is stopped when the number of cells in each bin is 3-4, as recommended in [9, 26].

Figure 4 shows the wirelength variation of the placers on PEKU with 0.0% nonlocal nets. Figure 5 shows the quality ratio of sPL on PEKU with 0.0% nonlocal nets. The quality

ratio is the wirelength achieved by the placer divided by the minimum wirelength. The results of sPL were averaged over 10 runs. QPlace [12] is the placement engine used in the Silicon Ensemble of Cadence. The version used is QPlace 5.1.55, in Silicon Ensemble v.5.3.

On PEKU with 0.25% to 10% nonlocal nets, sPL was run for a total of 5 times. Figures 6, 8, 10, 12, 14, 16, and 18 show the wirelengths. The quality ratios are reported in Figures 7, 9, 11, 13, 15, 17, and 19. The average runtimes for sPL and abcPL are shown in Figure 20.

The performance of sPL improves dramatically as the number of nonlocal nets increases, as shown in Figures 4,

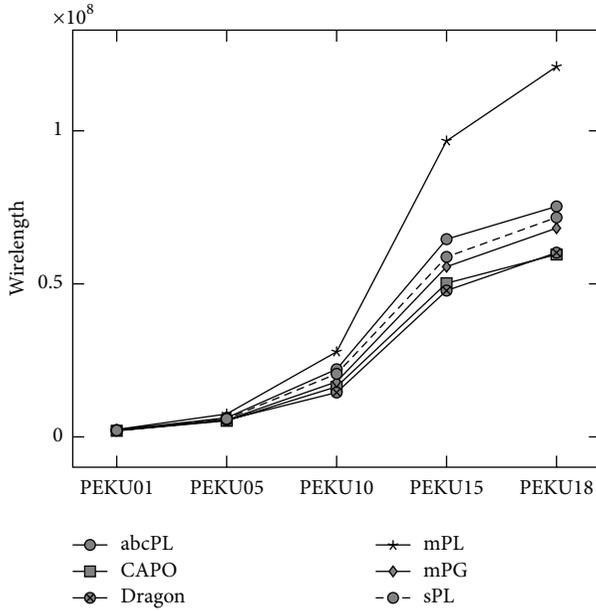


FIGURE 12: Wirelength variation on PEKU with 1% nonlocal nets.

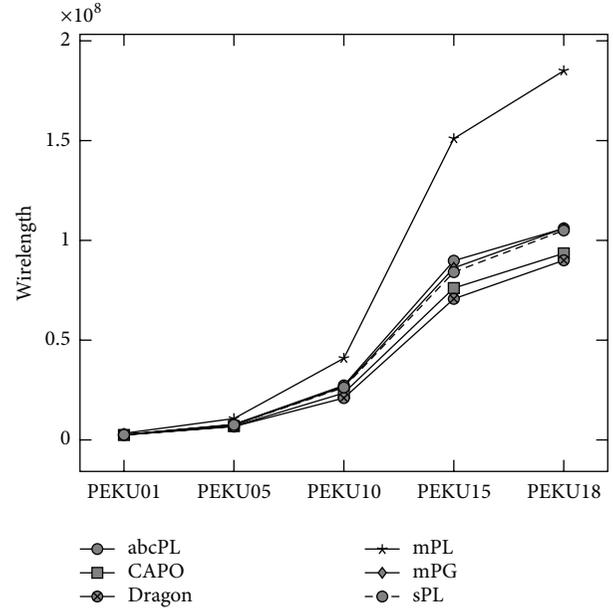


FIGURE 14: Wirelength variation on PEKU with 2% nonlocal nets.

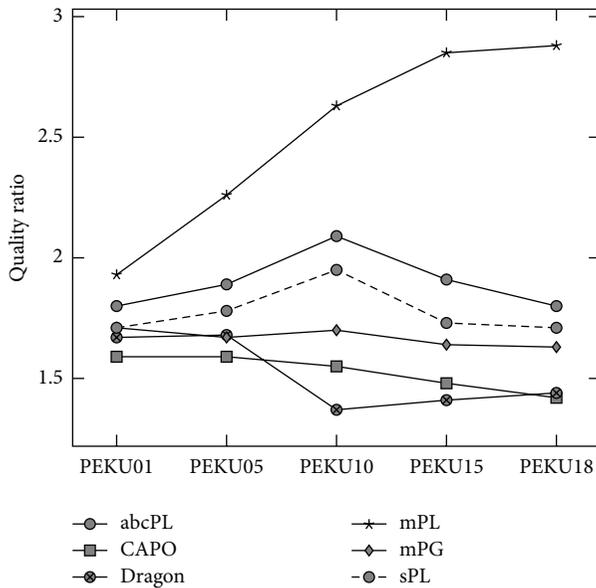


FIGURE 13: Quality ratios on PEKU with 1% nonlocal nets.

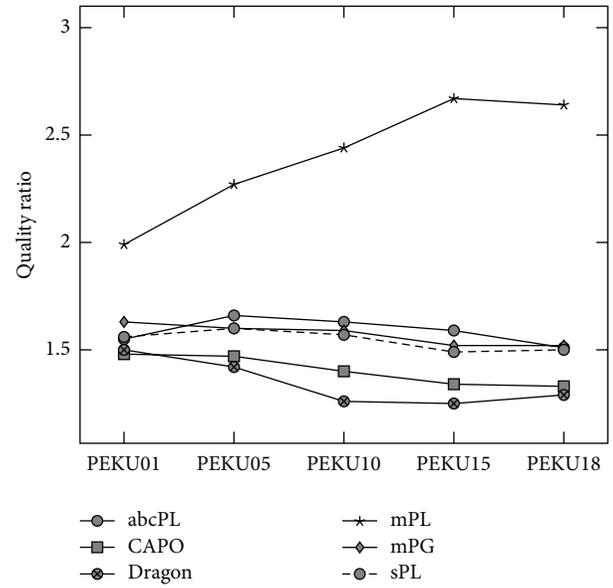


FIGURE 15: Quality ratios on PEKU with 2% nonlocal nets.

6, 8, 10, 12, 14, 16, and 18 for the variation of the wirelength and in Figures 5, 7, 9, 11, 13, 15, 17, and 19 for the variation of the quality ratio. This trend is similar to the performance improvement of Dragon and abcPL. This might be a result of increased nonlocal nets providing more global information on the optimal positions of cells to sPL (and other placers), allowing them to better place these cells. Nonlocal nets offer more leeway for placement of the cells compared with just local nets, allowing deeper wells in the fitness function and lesser local minima. This might explain the placer’s ability to achieve better solutions.

sPL shows that the performance of a random walk of a swarm is a promising approach for solving the SCP problem. The placer is not mature, and, as such, the performance of the first version is good compared to other, more mature, placers. Using SwarmRW is better than using ABC on all instances except for PEKU1 0% and 2%, where it is slightly worse. sPL is faster than abcPL on all tested instances. However, runtimes are long compared to mathematical tools, such as mPL. This can be explained by the frequent reads and writes to disk while partitioning using hMetis. sPL is better than mPL on all instances with 0.50% and over nonlocal nets. On instances with 0.75% and above, sPL performs comparably with mPG,

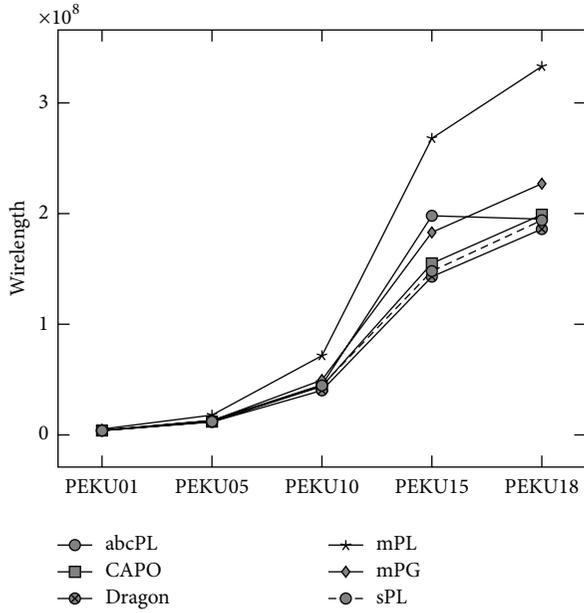


FIGURE 16: Wirelength variation on PEKU with 5% nonlocal nets.

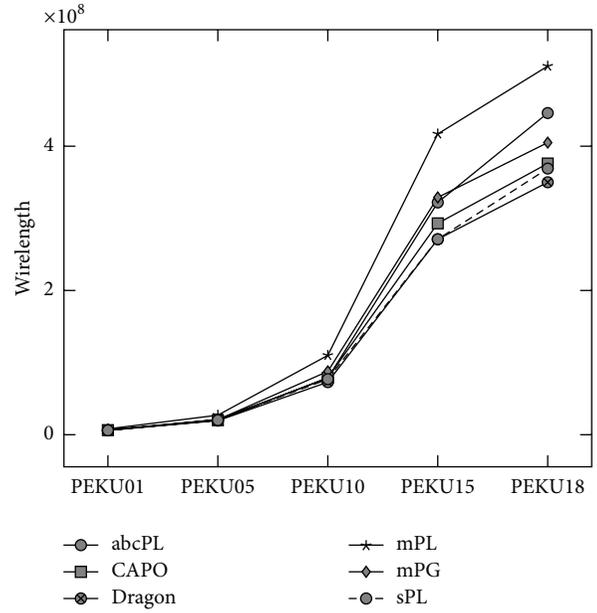


FIGURE 18: Wirelength variation on PEKU with 10% nonlocal nets.

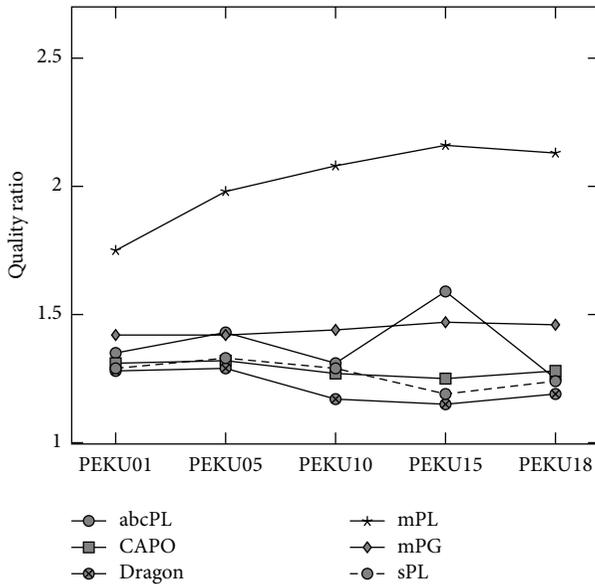


FIGURE 17: Quality ratios on PEKU with 5% nonlocal nets.

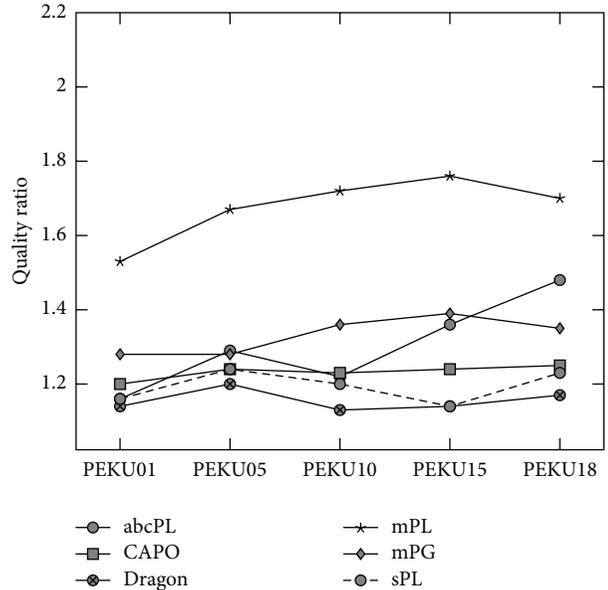


FIGURE 19: Quality ratios on PEKU with 10% nonlocal nets.

and outperforms it on some instances. It is comparable to Dragon and CAPO on instances with 5% and 10% nonlocal nets.

Results of sPL and the rival placers were analyzed using SPSS (<http://www-01.ibm.com/software/analytics/spss/>). The Friedman test was performed on the results obtained by the following placers: CAPO, Dragon, mPG, mPL, abcPL, and sPL. Results for PEKU 0.0% were not included in the test as mPG results are unavailable on those instances, so the total number of instances in the test is 35. Table 1 presents the descriptive statistics. The test showed there was a statistically

significant difference among all the placers with a P value of 0.000.

Next, the Wilcoxon ($1 * N$) signed-ranks test was run on sPL and the rest of the placers for pairwise analysis. Table 2 summarizes the results and the statistical significance among the placers. There is a statistically significant difference among all placer combinations except for mPG-sPL. Overall, sPL is better than mPL and abcPL is not statistically different from mPG and is worse than CAPO and Dragon.

We think that the performance of sPL can be improved by investigating other partitioning algorithms and heuristics to

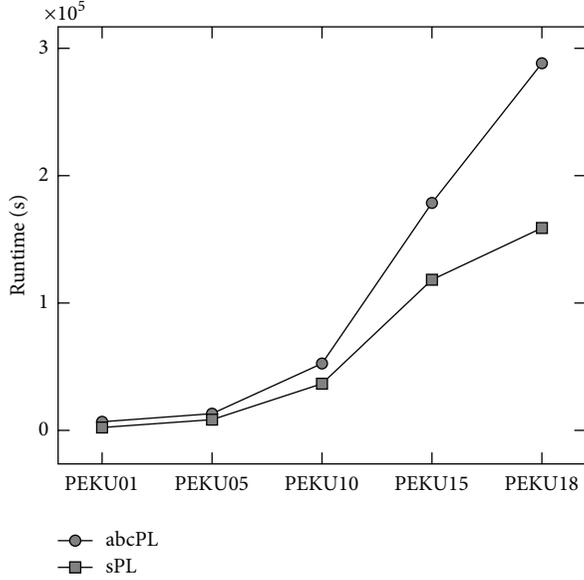


FIGURE 20: Average runtimes of abcPL and sPL.

TABLE 1: Friedman test descriptive statistics.

Placer	Mean rank	Mean	Standard deviation
Dragon	1.63	$4.94e + 7$	$7.74E + 07$
CAPO	1.90	$5.21e + 7$	$8.34E + 07$
mPG	3.47	$5.81e + 7$	$9.24E + 07$
sPL	3.50	$5.47e + 7$	$8.01E + 07$
mPL	5.67	$8.36e + 7$	$1.22E + 08$
abcPL	4.83	$6.15e + 7$	$9.51E + 07$

TABLE 2: Wilcoxon test.

Placers	z -test	P value (2-tailed)	Rank
Dragon-sPL	-5.433	0.000	Positive
CAPO-sPL	-3.582	0.000	Positive
mPG-sPL	-0.402	0.688	Positive
mPL-sPL	-3.992	0.000	Negative
sPL-abcPL	-5.394	0.000	Positive

generate neighbors. sPL uses a heuristic to reduce wirelength after converting bins to rows and final cell placement on the chip. Other heuristics, such as the one in Dragon, may also be beneficial to the wirelength reduction objective. In addition, partitioning and then merging within the same level may allow better recovery from partitioning errors. This is because if a cell was incorrectly partitioned, merging before repartitioning may allow the cell to be placed in the correct partition.

6. Conclusion and Future Work

This paper has introduced a new placer called sPL that solves the SCP problem based on the SwarmRW metaheuristic and min-cut based partitioner hMetis. The experimental results obtained on the PEKU benchmark demonstrate that sPL

outperforms particularly abcPL, a metaheuristic based placer, and mPL, a nonlinear programming based placer, on most of the instances. Moreover, sPL's performance remains close to the performance of two other well-established placers: Dragon and CAPO. Therefore, this shows that SwarmRW is a promising approach for the SCP problem, particularly for large problem instances as the results of wirelength and quality ratios improve dramatically for sPL as the number of nonlocal nets increases.

Future work includes converting the algorithm to deal with macroblocks in addition to standard cells, in order to solve the mixed-size cell placement problem. Other flat partitioners may produce partitions more suited to our approach, as the metaheuristics alter the placement after partitioning. In addition, other methods for producing neighboring solutions can also be studied. Investigation of various heuristics to improve the solution might also be beneficial. In addition, investigating other versions of SwarmRW for the SCP by testing various escape strategies may result in improving the performance of the proposed placer.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This research project was supported by a grant from the Research Center of the Center for Female Scientific and Medical Colleges, Deanship of Scientific Research, King Saud University.

References

- [1] K. Shahookar and P. Mazumder, "VLSI cell placement techniques," *ACM Computing Surveys*, vol. 23, no. 2, pp. 143–220, 1991.
- [2] W. E. Donath, "Complexity theory and design automation," in *Proceedings of the 17th Design Automation Conference*, pp. 412–419, ACM, 1980.
- [3] M. Wang, X. Yang, and M. Sarrafzadeh, "Dragon2000: standard-cell placement tool for large industry circuits," in *Proceedings of the IEEE/ACM International Conference on Computer Aided Design (ICCAD '00)*, pp. 260–263, IEEE, 2000.
- [4] N. Altwaijry and M. B. Menai, "A swarm random walk algorithm for global continuous optimization," in *Proceedings of the 7th International Conference on Genetic and Evolutionary Computing (ICGEC '13)*, J.-S. Pan, Ed., vol. 238 of *Advances in Intelligent and Soft Computing*, pp. 33–43, Springer, 2014.
- [5] G. Karypis, R. Aggarwal, V. Kumar, and S. Shekhar, "Multilevel hypergraph partitioning: application in VLSI domain," in *Proceedings of the 34th Design Automation Conference*, pp. 526–529, June 1997.
- [6] N. Selvakkumaran, "Multiobjective hypergraph-partitioning algorithms for cut and maximum subdomain-degree minimization," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 3, pp. 504–517, 2006.
- [7] J. Cong, J. R. Shinnerl, M. Xie, T. Kong, and X. Yuan, "Large-scale circuit placement," *ACM Transactions on Design Automation of Electronic Systems*, vol. 10, no. 2, pp. 389–430, 2005.

- [8] J. Cong, M. Romesis, and M. Xie, "Optimality, scalability and stability study of partitioning and placement algorithms," in *Proceedings of the International Symposium on Physical Design (ISPD '03)*, pp. 88–94, ACM, New York, NY, USA, April 2003.
- [9] X. Yang, *Congestion and timing optimization for standard-cell placement [Ph.D. thesis]*, University of California, Los Angeles, Calif, USA, 2002.
- [10] N. Altwajry and M. B. Menai, "Artificial bee colony for the standard cell placement problem," *International Journal of Metaheuristics*, vol. 2, no. 3, pp. 234–255, 2013.
- [11] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Tech. Rep. tr06, Erciyes University Press, Erciyes, Turkey, 2005.
- [12] Cadence Design Systems Inc, "Envisia ultra placer reference," QPlace Version 5.1.55, 1999.
- [13] C. C. Chang, J. Cong, and Z. D. Pan, "Physical hierarchy generation with routing congestion control," in *Proceedings of the International Symposium on Physical Design (ISPD '02)*, pp. 36–41, Del Mar, Calif, USA, April 2002.
- [14] T. F. Chan, J. Cong, T. Kong, and J. R. Shinnerl, "Multilevel optimization for large-scale circuit placement," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pp. 171–176, IEEE Press, 2000.
- [15] T. F. Chan, J. Cong, T. Kong, J. R. Shinnerl, and K. Sze, "An enhanced multilevel algorithm for circuit placement," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, p. 299, IEEE Computer Society, 2003.
- [16] T. Taghavi, X. Yang, B.-K. Choi, M. Wang, and M. Sarrafzadeh, "Congestion minimization in modern placement circuits," in *Modern Circuit Placement*, G. J. Nam and J. Cong, Eds., Integrated Circuits and Systems, pp. 135–163, Springer, 2007.
- [17] A. E. Caldwell, A. B. Kahng, and I. L. Markov, "Can recursive bisection alone produce routable placements?" in *Proceedings of the 37th Design Automation Conference (DAC '00)*, pp. 477–482, ACM, June 2000.
- [18] A. E. Caldwell, A. B. Kahng, and I. L. Markov, "Improved algorithms for hypergraph bipartitioning," in *Proceedings of the Asia and South Pacific Design Automation Conference*, pp. 661–666, ACM, 2000.
- [19] P. Subbaraj, S. Saravana Sankar, and S. Anand, "Parallel genetic algorithm for VLSI standard cell placement," in *Proceedings of the International Conference on Advances in Computing, Control and Telecommunication Technologies (ACT '09)*, pp. 80–84, Trivandrum, India, December 2009.
- [20] T. W. Manikas and M. H. Mickle, "A genetic algorithm for mixed macro and standard cell placement," in *Proceedings of the 45th Midwest Symposium on Circuits and Systems (MWSCAS '02)*, vol. 2, pp. II-115–II-118, Tulsa, Okla, USA, August 2002.
- [21] J. A. Khan and S. M. Sait, "Fast fuzzy force-directed/simulated evolution metaheuristic for multiobjective vlsi cell placement," *Arabian Journal for Science and Engineering*, vol. 32, no. 2, pp. 263–280, 2007.
- [22] S. M. Sait, M. I. Ali, and A. M. Zaidi, "Multiobjective vlsi cell placement using distributed simulated evolution algorithm," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 6, pp. 6226–6229, Kobe, Japan, May 2005.
- [23] Z. Zha, *A highly scalable, parallel genetic algorithm for standard cell placement in VLSI design [Ph.D. thesis]*, Lamar University, Beaumont, Tex, USA, 2004.
- [24] S. M. Sait, M. Faheemuddin, M. I. Ali, and S. Sanaullah, "Simulated evolution based hybrids for genetic algorithm and tabu search," in *Proceedings of the International Conference on Computers and Their Applications (CATA '07)*, pp. 30–35, Honolulu, Hawaii, USA, March 2007.
- [25] Y. Shiraishi, T. Ono, and M. A. E. Dabb, "Solution space reduction of simulated evolution algorithm for solving standard cell placement problem," in *Proceedings of the 5th International Conference on Natural Computation (ICNC '09)*, pp. 420–424, Tianjin, China, August 2009.
- [26] X. Yang, M. Wang, K. Eguro, and M. Sarrafzadeh, "Snap-on placement tool," in *Proceedings of the International Symposium on Physical Design*, pp. 153–158, ACM, April 2000.

Research Article

Core Business Selection Based on Ant Colony Clustering Algorithm

Yu Lan,¹ Yan Bo,² and Yao Baozhen³

¹ School of International Trade and Commerce, Yanching Institute of Technology, Beijing 065201, China

² Transportation Management College, Dalian Maritime University, Dalian 116026, China

³ School of Automotive Engineering, Dalian University of Technology, Dalian 116024, China

Correspondence should be addressed to Yao Baozhen; yaobaozhen@yahoo.cn

Received 19 May 2014; Revised 10 July 2014; Accepted 14 July 2014; Published 6 August 2014

Academic Editor: Rui Mu

Copyright © 2014 Yu Lan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Core business is the most important business to the enterprise in diversified business. In this paper, we first introduce the definition and characteristics of the core business and then describe the ant colony clustering algorithm. In order to test the effectiveness of the proposed method, Tianjin Port Logistics Development Co., Ltd. is selected as the research object. Based on the current situation of the development of the company, the core business of the company can be acquired by ant colony clustering algorithm. Thus, the results indicate that the proposed method is an effective way to determine the core business for company.

1. Introduction

Core business is the most important business to the enterprise in diversified business. It is the business with the best ability in enterprise management. It has a high added value and the best prospects for development among the enterprise business and can create a relatively high value to the enterprise [1]. Core business decides economic efficiency of enterprises which is closely related to enterprise's strategic positioning. Specifically, the core business is a competitive choice in the intense market competition with related peers [2]. This choice is consistent and associated with the ability and resources of the enterprise. Since the company's resources can be of better and greater use, so the core business has a competitive advantage that noncore businesses do not have and is the key point of the enterprise to stand out among competitors and major source of profit. Development of core business is the big question about the survival of enterprises and the decisive factor in the enterprise based on the market. Meanwhile, the core business is the emphasis and consideration in enterprises continuing strategic positioning [3].

Campbell et al. [4] proposed an alternative frame work guiding core business diversification strategy based on some matching factors. Their results can provide a useful method

to guide enterprises to develop their own core business but do not give a specific actionable recognition method based on the above research. For enterprises, it's difficult to find their core business. However, without core business, the enterprises cannot reasonably arrange their corporate resources. Therefore, how to establish the selection method of core business is a very important role to optimize corporate resources and thus improve the level of corporate strategic decisions.

Ant colony algorithm is a newly proposed bionic optimization algorithm, which has unique advantages in solving complex optimization problems [5]. Currently, in the study of algorithms, the study of clustering algorithm based on ant behavior is not much. But, only some studies have shown that behavior-based clustering algorithm can solve complex problems that many traditional clustering algorithms are difficult to resolve and is a promising research direction [6–16]. This paper introduces the ant colony clustering algorithm into the core business areas and mentions a new selection method of the core business that seeks to get closer to reality and provides a theoretical basis for enterprises to select the core business. The method is calculated by a case, and the results will be compared with those obtained by traditional clustering methods.

2. Core Business Segment Indicators

The company's core business is not the only one and there may be multiple core businesses. American experts have done a survey for the enterprise that had been investigated and studied by Bain Capital [17]. They found that 87% of the reasons for the enterprise healthy and sustainable development are enterprises having their own core business. But, only three percent of enterprises can have more than three core businesses [18]. Through the analysis of the core business, we can find the direction that enterprises focus on.

2.1. The Extent of the Resource Allocation. Business mainly relies on investment and production to exchange material and information in the community [19]. If the core business can be well developed, then the company's core competencies will be improved. So, leadership is more focus on core business in the enterprise. Companies tend to give priority to the core business when making the allocation of resources, such that the cost of the study will be more willing to invest. But, if there is excessive diversity issue, it will result in disproportionate or unreasonable resource allocation, making noncore businesses wasted resources that should belong to the core business. Enterprise resource has a certain limit and too much business sector will lead to resource misallocation and affect the core competitiveness of the enterprise [20].

2.2. The Contribution of the Enterprise Output. Core business brings together the most outstanding resources of the enterprise and occupies the largest part of human and material resources of corporate [21]. So, the core business will inevitably determine the overall output of the enterprise. Due to the different nature of each industry, the core business will be different, but the core business, without exception, is the business with the largest investment and utilization of resources.

2.3. The Ability to Obtain the Degree. Competitiveness of enterprises mainly comes from the company's core competencies. Core competence is also comprehensive enterprise knowledge. It includes the recessive and dominant ability of enterprises and is the source of enterprises to maintain a competitive advantage in the fierce market competition [22]. Through ability support to the business sector, core competence can make the operating department transform the core ability to their capabilities on the basis of their original. However, for the absorption of core competencies of different business sectors, there are some differences in a certain extent. Different departments have their own absorption capacity, so these differences of ability have led to differentiation in the effect of the business sector [23].

We summarize the characteristics of the core business and obtain segment indicators of core business and we can find that a major feature of the core business is to have a higher share of resources. In most enterprises, the businesses with higher input and a higher proportion of resources also create higher revenue and this is consistent with the second feature of the core business. The best indicator to

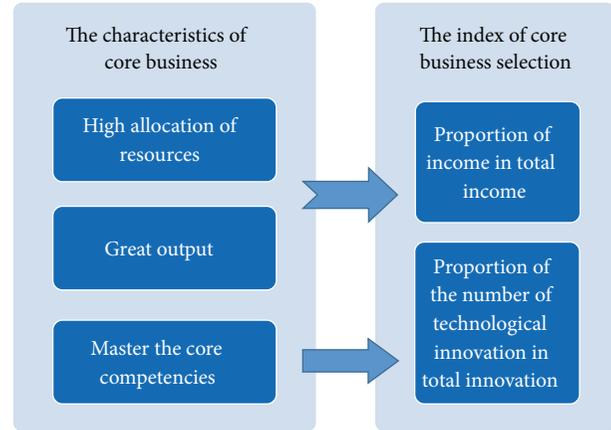


FIGURE 1: The derivation process of segment indicators.

measure the extent of a business in revenue contribution is the proportion of income in total income of this business. The third characteristic of the core business is that it can access competitiveness better. In today's competitive environment, the key to competitive advantage lies in innovation. Therefore, the key to measure whether a business is core business is to evaluate whether the business is innovative, specifically, is the proportion of the number of technological innovation in total innovation. We select proportion of the number of technological innovations in total innovation and proportion of income in total income as the main factors influencing the choice of the core business. This process is shown in Figure 1.

3. Ant Colony Clustering Algorithm

The inspiration of ant colony clustering algorithm derived from the ants piled their corpses and classification of their larvae. Deneubourg et al. [24] were enlightened by the behavior of ant colony classified ant eggs and firstly proposed the ant colony clustering algorithm. In the algorithm, the robot imitates the behavior of ants to identify two different objects. Thus, according to the similarity of data objects and their surrounding objects, ants move randomly, picking up or dropping data objects, in order to achieve the purpose of data clustering, and this model is successfully applied to the field of robotics. Thereafter, Lumer and Faieta [25], Ramos and Merelo, and so forth [26] carried out some research in this area and got some successful experience. Although the study of ant colony clustering algorithm is not a lot, but preliminary studies [27–29] have shown that the algorithm is a good clustering algorithm and can be applied to complex clustering problem in many areas of engineering.

In order to carry out the algorithm description, we introduce the basic principle of the algorithm. First, the data is randomly projected onto a target plane, and then select a data object to each ant randomly. According to the probability obtained from the similarity of object in the local area, determine whether the ants pick up, putdown, or move object. After a limited number of iterations, the data objects on the plane are aggregating according to their similarity to give the final clustering result.

```

(1) The number of individual ants is  $N$ ; maximum number of iterations is  $M$ ; side length of localized area is  $s$ ; velocity of ants,  $v$ , takes a random number.
(2) The data object is projected onto a plane and each object is assigned a point  $(x, y)$ .
(3) Each ant is initialized to unload and randomly selects an object.
(4) for  $i = 1, 2, \dots, M$ 
      for  $j = 1, 2, \dots, N$ 
        Calculate the average similarity of object.
        If the antis unloaded, then calculate the probability of picking up,  $P_p$ . If  $P_p$  is greater than a random probability, while on the same time the object is not being picked up by other ants, the ant picks up the object and moves elsewhere randomly, and marks that it is load; otherwise, the ant refuses to pick up the object and selects other objects randomly.
        If the antis loaded, then calculate the probability of putting down,  $P_d$ . If  $P_d$  is greater than a random probability, the ant puts down the object and marks that it is unload, and then re-select a new object. Otherwise, the ant continues to move the object to a new location.
      End
    End
  End
(5) for  $i = 1, 2, \dots, N$ 
  If an object is isolated or the number of its neighborhood is less than a certain constant, then mark the object as an isolated point; otherwise, the object is assigned to a cluster serial number, and recursively their neighborhood object tag for the same serial number.
End

```

PSEUDOCODE 1

Pseudocode of basic process of algorithm is as shown in Pseudocode 1.

3.1. Calculation of the Average Similarity. Assuming an ant found the data object o_i at time t in placer, the average similarity of object o_i and its neighborhood object o_j is calculated by the following formula:

$$f(o_i) = \max \left\{ 0, \frac{1}{s^2} \sum_{o_j \in \text{Neigh}_{s \times s}(r)} \left[1 - \frac{d(o_i, o_j)}{\alpha (1 + (v - 1) / v_{\max})} \right] \right\}, \quad (1)$$

where α is a similarity parameter; v is the speed of ants; v_{\max} is the maximum speed; $\text{Neigh}_{s \times s}(r)$ is a square of side length s of the local area around place r . $d(o_i, o_j)$ represents the distance between o_i and o_j in the property space. α is the parameter to adjust the similarity between data, and it also determines the number of clusters and the speed of convergence. The larger the α is, the greater the degree of similarity between objects is, the fewer the number of clustering is, and the faster the convergence is. This also may make different objects classified as a class. On the contrary, the smaller the α is, the smaller the degree of similarity between objects is. In extreme cases, a big category can be divided into many small categories while increasing the number of clusters and slowing down the convergence. Thus, in the actual calculation, α should be determined by experience or trial. In addition, the movement speed of the ants also influences the effect of clustering. The faster ants can roughly divide the objects into large categories, the slower ants can more accurately divide the objects. In this article, in order to facilitate the study, ants speed is set to a random number from 1 to v_{\max} .

3.2. Calculation of the Probability Conversion Function. Probability conversion function is a function about $f(o_i)$. It converts the average similarity of data objects into the probability of picking up or putting down. Conversion principle is the smaller the average similarity of data objects and their neighborhood is, the smaller the possibility of data object that belongs to this neighborhood is, the higher the probability of picking up is, and the lower the probability of putting down is, and vice versa. According to this principle, the symmetric Sigmoid function is selected as probability conversion function:

$$\text{Sigmoid}(f(o_i)) = \frac{1 - e^{-cx}}{1 + e^{-cx}}. \quad (2)$$

The probability of an ant that is unloaded and move randomly picking up an object is defined as

$$P_p = 1 - \text{Sigmoid}(f(o_i)). \quad (3)$$

Similarly, the probability of an ant that is loaded and moves randomly putting down an object is defined as

$$P_d = \text{Sigmoid}(f(o_i)). \quad (4)$$

It should be noted that in the process of clustering, some objects called isolated points are not similar to other objects. Once the ants pick them up, it is hard to put them down as soon as possible, which affects the convergence speed. Here, we increase the value of c in the latter algorithm to lay down the isolated points as soon as possible.

4. Computational Experiments

4.1. Brief Introduction of Tianjin Port Logistics Development Co., Ltd. Tianjin Port Logistics Development Co., Ltd. was

established in April 1, 2009 with a registered capital of 667 million yuan and 2,800 employees. The company has the ability yard area of nearly 1.2 million square meters, the distance between the marina and the main was less than three kilometers, and stockpiling even reached 40,000 TEUs. The company has a dedicated railway line which is up to 4080 meters and seven dedicated rail line of sea. Thus, railway transportation can be expanded. Because of the structure of dry port, the functions of port are even extended to the hinterland of the city. By this barrier-free access between the port and the hinterland, the company is able to provide customers with good service in domestic shipping, freight forwarding, handling, and other logistics services. On the basis of the excellent resources on the port, combining with the function of the dock, Port Group company continued to attract a variety of integrated logistics resources; its type of business has covered shipping agency, freight forwarding, warehousing and transportation, bonded logistics, large logistics and warehouse receipts pledge, marine insurance, and other related logistics financial services; and with a number of shipping companies and land transport company it established a strategic alliance, into a port logistics featuring large-scale comprehensive logistics enterprises. Tianjin Port Logistics Development Co., Ltd. adheres to the market demand, through the use of information networks, building up the whole process of the logistics network platform. On the basis of full understanding of industry characteristics, company further optimizes resource allocation, enhances service competitiveness, and strengthens marketing force, and with good logistics infrastructure, advantages of port resources, and efficient operation of the team and good service system, dedicated to providing our customers with better quality, modern logistics services.

Table 1 selects ten businesses that Tianjin Port Logistics Development Co., Ltd. is mainly engaged in. From the data in the table, we can see that the profit of domestic cargo agents, pier horizontal transport, container yard operations, and automotive logistics are the biggest. To Tianjin Port Logistics Development Co., Ltd., such a logistic service enterprise without actual output products, revenue is the basis of corporate survival. Therefore, it can be concluded from the data in the table that domestic cargo agents, pier horizontal transport, container yard operations, and automotive logistics are core businesses in Tianjin Port Logistics Development Co., Ltd.

4.2. Cluster Calculation of Sample Enterprises. According to the results of the second chapter, we select proportion of the number of technological innovations in total innovation and proportion of income in total income as the main factors influencing the choice of the core business. Relevant data are shown in Table 1. In order to characterize the importance of the business, the business is divided into four levels, namely, core business, potential core business, integrated business, and recession business. They are represented as 1, 2, 3, and 4 in the calculation.

Using ant colony clustering algorithm to calculate sample examples in Table 1, algorithm parameters of calculation take the value as follows: the number of individuals of ants, N , is

TABLE 1: Tianjin Port Logistics Development Co., Ltd. patent and business income proportion table.

Project	Proportion of the number of technological innovation in total innovation	Proportion of income in total income
Domestic shipping and cargo agents	0	27.3%
International shipping and cargo agents	0	0.56%
Pier horizontal transport	30%	20.5%
Container yard operations	35%	37.1%
Customs clearance	0	0.56%
Multimodal transportation	5%	1.4%
Automotive logistics	8%	10%
Bonded logistics	1%	0.42%
Project logistics	3%	1.7%
Hazardous material logistics	8%	0.42%

TABLE 2: Computed results.

Project	Result	
	Conventional clustering	Ant clustering
Domestic shipping and cargo agents	3	3
International shipping and cargo agents	4	4
Pier horizontal transport	2	1
Container yard operations	1	1
Customs clearance	4	4
Multimodal transportation	3	3
Automotive logistics	3	2
Bonded logistics	2	2
Project logistics	3	2
Hazardous material logistics	2	3

10; the maximum number of iterations, M , is taken as 2000; similarity parameter, α , is taken as 1.5; the maximum speed of ants, v_{\max} , is taken as 0.8; length of local area, s , is taken as 2; the parameter in the probability transfer function, c , is taken as 3.

Under these conditions, the results are shown in Table 2. For comparison, the calculated results of conventional clustering method are also shown in Table 2.

According to the results of Table 2, the company's core businesses are container yard business and pier horizontal transport business; potential core businesses are automotive logistics, bonded logistics, and project logistics; integrated

businesses are domestic shipping and cargo agents, multimodal transportation and hazardous material logistics; recession businesses are international shipping and cargo agents and customs clearance.

4.3. Comparative Analyses. Through the analysis of Table 2, we can find that the result obtained by the ant colony clustering algorithm and the result obtained by the conventional clustering algorithm are basically the same. But, the result obtained by the ant colony clustering algorithm is more consistent with the basic situation of enterprises.

Conventional clustering algorithm needs to determine cluster centers in advance and there are strict requirements for input order of sample. As a local search algorithm, it cannot handle local minima problems. Therefore, when the initial cluster centers in the whole sample space are imbalanced, it is difficult to rectify this imbalance. As a new bionic algorithm, ant colony clustering algorithm does not have these questions. Visibly, ant colony clustering algorithm is a good analysis algorithm with broad applicability and higher accuracy.

5. Conclusions

This paper described the concept and characteristics of the core business, selects Tianjin Port Logistics Development Co., Ltd. as the research objects, and uses SVM method to analyze ten main businesses of Tianjin Port Logistics Development Co., Ltd. Finally, we come to the core business of Tianjin Port Logistics Development Co., Ltd.: container yard business, domestic ship and cargo agency business, pier horizontal transport business. There, our contribution is attempted to use ant colony clustering algorithm to solve the core business selection and provide a new method for researchers and producers.

Although we achieved the expected results in the research of the core business choice of Tianjin Port Logistics Development Co, Ltd., there are still some limitations on this study because of the complexity, variability, and unpredictable nature of the internal and external environment and the restriction of paper length and the author's own level. And the case used in this paper is relatively simple, too. Therefore, more sophisticated scenario needs to be further studied in our future work.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by National Natural Science Foundation of China 51208079, the Fundamental Research Funds for the Central Universities 3013-852019, and the help from Dr. B. YU from School of Traffic and Transportation, Beijing Jiaotong University, and Dr. L.P. Zhou from Wuxi Mingda Traffic & Technology Consulted Co., Ltd.

References

- [1] C. K. Prahalad and G. Hamel, *The Core Competence of the Corporation*, Harvard Business Review, 1990.
- [2] U. Arnold, "New dimensions of outsourcing: a combination of transaction cost economics and the core competencies concept," *European Journal of Purchasing and Supply Management*, vol. 6, no. 1, pp. 23–29, 2000.
- [3] G. Michael and L. Kathleen, "Why diversify? Four decades of management thinking," *Executive*, vol. 7, no. 3, pp. 7–25, 1993.
- [4] A. Campbell, M. Goold, and M. Alexander, *Corporate Strategy: The Quest for Parenting Advantage*, Harvard Business Review, 1995.
- [5] M. Dorigo, V. Maniezzo, and A. Coloni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 26, no. 1, pp. 29–41, 1996.
- [6] B. Yao, C. Yang, J. Yao, and J. Sun, "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.
- [7] B. Z. Yao, J. B. Yao, M. H. Zhang, and L. Yu, "Improved support vector machine regression in multi-step-ahead prediction for rock displacement surrounding a tunnel," *Scientia Iranica*, accepted, 2013.
- [8] B. Yu, Z. Yang, and K. Chen, "Hybrid model for prediction of bus arrival times at next station," *Journal of Advanced Transportation*, vol. 44, no. 3, pp. 193–204, 2010.
- [9] B. Yu, Z. Yang, and S. Li, "Real-time partway deadheading strategy based on transit service reliability assessment," *Transportation Research A: Policy and Practice*, vol. 46, no. 8, pp. 1265–1279, 2012.
- [10] B. Yu, T. Ye, X.-M. Tian, G.-B. Ning, and S.-Q. Zhong, "Bus travel-time prediction with forgetting factor," *Journal of Computing in Civil Engineering*, vol. 28, no. 3, 2012.
- [11] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research C*, vol. 43, pp. 233–248, 2014.
- [12] B. Yao, P. Hu, M. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem," *Simulation*, vol. 89, no. 6, pp. 762–770, 2013.
- [13] B. Z. Yao, P. Hu, M. H. Zhang, and X. M. Tian, "Improved ant colony optimization for seafood product delivery routing problem," *PROMET-TRAFFIC & Transportation*, vol. 26, no. 1, pp. 1–10, 2014.
- [14] B. Yu, Z. Yang, and J. Yao, "Genetic algorithm for bus frequency optimization," *Journal of Transportation Engineering*, vol. 136, no. 6, pp. 576–583, 2010.
- [15] B. Yao, Z. Z. Yang, and B. Z. Yao, "A hybrid algorithm for vehicle routing problem with time windows," *Expert Systems with Applications*, vol. 38, no. 1, pp. 435–441, 2011.
- [16] B. Yu, H. B. Zhu, W. J. Cai, N. Ma, Q. Kuang, and B. Z. Yao, "Two-phase optimization approach to transit hub location—the case of Dalian," *Journal of Transport Geography*, vol. 33, pp. 62–71, 2013.
- [17] C. K. Prahalad, "The role of core competence in the corporation," *Research Technology Management*, pp. 40–47, 1993.
- [18] C. Zook and J. Allen, *Main Profit*, CITIC Publishing House, Beijing, China, 2002.
- [19] G. Hamel, "Competition for competence and inter-partner learning within international strategic alliances," *Strategic Management Journal*, vol. 12, pp. 83–100, 1991.

- [20] E. P. Przybylowicz and T. W. Faulkner, "Kodak applies strategic intent to the management of technology," *Research Technology Management*, no. 1, pp. 31–38, 1993.
- [21] G. Hnaje and C. K. Prahalad, *Competing for the Future*, Harvard Business School Press, Cambridge, Mass, USA, 1994.
- [22] D. Klostad, M. Kesler, and W. E. Clarke, "Third generation R&D: the key to leveraging core competencies," *The Columbia Journal of World Business*, vol. 28, no. 3, p. 34, 1993.
- [23] R. E. Stake, "Case studies," in *Handbook of Qualitative Research*, N. K. Denzin and Y. S. Lincoln, Eds., pp. 435–454, Sage, Thousand Oaks, Calif, USA, 2000.
- [24] J. L. Deneubourg, S. Goss, N. Franks et al., "The dynamics of collective sorting: robot-like ant and ant-likerobot," in *Proceedings 1st Conference on Simulation of Adaptive Behavior: From Animals to Animals*, pp. 356–365, MIT Press, Cambridge, Mass, USA, 1991.
- [25] E. Lumer and B. Faieta, "Diversity and adaptation in populations of clustering ants," in *Proceedings of the 3rd International Conference on Simulation of Adaptive Behavior: From Animals to Animals*, pp. 499–508, MIT Press, Cambridge, Mass, USA, 1994.
- [26] V. Ramos and J. J. Merelo, "Self-organized stigmergic document maps: environment as a mechanism for context learning," in *Proceedings of the 1st Spanish Conference on Evolutionary and Bio-Inspired Algorithms (AEB '02)*, pp. 284–293, Mérida, Spain, 2002.
- [27] M. Reed, A. Yiannakou, and R. Evering, "An ant colony algorithm for the multi-compartment vehicle routing problem," *Applied Soft Computing*, vol. 15, pp. 169–176, 2014.
- [28] J. Wang, A. Tu, and H. Huang, "An ant colony clustering algorithm improved from ATTA," *Physics Procedia B*, vol. 24, pp. 1414–1421, 2012.
- [29] J. Ji, X. Song, C. Liu, and X. Zhang, "Ant colony clustering with fitness perception and pheromone diffusion for community detection in complex networks," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 15, pp. 3260–3272, 2013.

Research Article

A Framing Link Based Tabu Search Algorithm for Large-Scale Multidepot Vehicle Routing Problems

Xuhao Zhang,¹ Shiquan Zhong,¹ Yiliu Liu,² and Xuelian Wang³

¹ College of Management and Economics, Tianjin University, Tianjin 300072, China

² Department of Production and Quality Engineering, Norwegian University of Science and Technology, 7491 Trondheim, Norway

³ School of Management, Hebei University of Technology, Tianjin 300401, China

Correspondence should be addressed to Shiquan Zhong; shiquan_tju@163.com

Received 21 April 2014; Accepted 29 June 2014; Published 23 July 2014

Academic Editor: Baozhen Yao

Copyright © 2014 Xuhao Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A framing link (FL) based tabu search algorithm is proposed in this paper for a large-scale multidepot vehicle routing problem (LSMDVRP). Framing links are generated during continuous great optimization of current solutions and then taken as skeletons so as to improve optimal seeking ability, speed up the process of optimization, and obtain better results. Based on the comparison between pre- and postmutation routes in the current solution, different parts are extracted. In the current optimization period, links involved in the optimal solution are regarded as candidates to the FL base. Multiple optimization periods exist in the whole algorithm, and there are several potential FLs in each period. If the update condition is satisfied, the FL base is updated, new FLs are added into the current route, and the next period starts. Through adjusting the borderline of multidepot sharing area with dynamic parameters, the authors define candidate selection principles for three kinds of customer connections, respectively. Link split and the roulette approach are employed to choose FLs. 18 LSMDVRP instances in three groups are studied and new optimal solution values for nine of them are obtained, with higher computation speed and reliability.

1. Introduction

Nowadays, logistic cost is considered to be influential in fierce business competitions. The logistical delivery is a process from pickup to drop-off of goods, connecting vendors, shippers, and customers. In related studies, Vehicle Routing Problem (VRP) is always regarded as a significant issue, due to its impact on optimization of delivery vehicles scheduling and then on profit of logistic providers.

In common sense, Vehicle Routing Problem is defined as follows: how to determine appropriate delivery routes between series of collection and reception terminals and guarantee delivery vehicles in a proper order, so as to satisfy some requirements (e.g., shortest distance, minimum cost, delivery time, and needed vehicles) within kinds of constraints, such as amount of goods, sending time, vehicle capacity, mileage restriction, and time limitation. Currently, VRPs have been identified in many applications, such as products outbound distribution scheduling [1], home care

crew scheduling [2], newspaper delivery [3], school bus routing [4], cargo routing [5], airline crew scheduling [6], waste collection scheduling [7], service system design [8], and computer system integration [9].

It is difficult to solve large-scale VRPs due to their complexities. Lenstra and Rinnooy Kan [10] proved that capacity constraint VRP (CVRP) was a NP-hard problem and recently Hassin and Rubinstein [11] verified the availability of polynomial-time algorithm in the cases where $k = 3$ or 4 for the k-VRP issue. Imai and his partners [12] highlighted that the Vehicle Routing Problem with full container load was also NP-hard. Furthermore, Solomon [13] realized that VRP with time window constraints was more complicated, and Hashimoto et al. [14] confirmed the NP-hard characteristics of VRP with soft time windows. In the paper of Savelsbergh [15], the author proved that it is a NP-complete problem to decide whether a feasible solution existed or not for the TSP with time windows. Lenstra and Rinnooy Kan [10] proved that all types of VRPs are NP-hard problems.

From basic capacity constraint VRP, researchers have recognized varied VRP problems, for example, VRP with time window (VRPTW), periodical VRP (PVRP), VRP with pickups and deliveries (VRPPD), and multidepot VRP (MDVRP). Compared with VRP with a single depot, MDVRP with more than one depot is more complicated. In MDVRP, not only the delivery sequence but also the vehicle type and amount for customers at each distribution center need to be determined.

Given that $G = (V, A)$ is a complete graph, $V = (1, \dots, n)$ is the set of vertexes in the chart and A is the set of arcs. $V_c = (1, 2, \dots, n)$ represents vertexes, $V_d = (n + 1, \dots, n + m)$ represents depots, and q_i is the delivery amount to vertex i . The capacity of vehicles from depot j is W_j , $j = n + 1, \dots, n + m$, and c_{ij} is the delivery cost from vertex i to j . In this paper, MDVRP is described as how to establish multiple paths in the chart G , so as to satisfy the following:

- (a) each path starts and ends at the same depot;
- (b) all vertexes are allocated to paths, and each vertex is allocated once;
- (c) the total delivery amount of depot j is no more than W_j , $j = n + 1, \dots, n + m$.

A heuristic approach is adapted by most researchers in dealing with MDVRP. It can be divided into the classical heuristic and the metaheuristic. Initially, the classical heuristic was more acceptable; for example, Gillett and Johnson [16] applied sweep heuristic in MDVRP and Golden et al. [17] used borderline search strategy and improved saving algorithm. A three-level heuristic algorithm was proposed by Salhi and Sari [18], where feasible solutions were generated on level 1 and delivery routes were optimized on levels 2 and 3. Sumichras and Markham [19] developed a C-W saving method. Wasner and Zäpfel [20] proposed a local search strategy with a series of feedback loops, and Nagy and Sahli [21] put forward multiple enhanced optimization strategies for MDVRP with pickups and deliveries problem (MDVRPPD). Lim and Wang [22] offered two solution methodologies—one-stage and two-stage approaches—to solve MDVRP with fixed distribution of vehicles. Recently, three hybrid heuristics were proposed by Mirabi et al. [23] for MDVRP, which were based on deterministic, stochastic techniques, and simulated annealing (SA) methods. Contardo and Martinelli [24] designed a new exact method to solve the MDVRP based upon the vehicle-flow formulation and the set-partitioning formulation.

Since the 1980s, some innovative optimization methods, such as genetic algorithm, simulated annealing, tabu search, and ant colony algorithm, have been developed greatly and acted as creative roles in tackling VRP. Cordeau et al. [25] proposed a tabu search heuristic effective for three well-known routing problems: PVRP, the periodic traveling salesman problem (PTSP), and MDVRP. Such a method generalized a tabu search in solutions to VRP. This author [26] then improved the above algorithm, to solve the periodic and the multidepot vehicle routing problems with time windows (PVRPTW and MDVRPTW). Similar methods occurred in the studies of Renaud et al. [27] and Crevier et al. [28]. Renaud et al. [27] divided the algorithm into two stages and

employed a tabu search to optimize the feasible solutions generated with heuristic. Crevier et al. [28] combined adaptive memory principle, a tabu search method, and the integer programming. Belhaiza et al. [29] presented a new hybrid variable neighborhood-tabu search heuristic for VRP with multiple time windows. Genetic algorithm is also acceptable by researchers. Bae et al. [30] developed an integrated VRP solver based on an improved genetic algorithm. Ho et al. [31] designed two hybrid genetic algorithms: one generated initial routes randomly and another created initial routes with Wright saving method and the nearest neighbor heuristic. In addition, Wang et al. [32] used genetic algorithm to study more complicated MDVRPs with constraints of time windows, limited numbers of vehicles, and multitype vehicles. Besides, the ant colony algorithm [33] and the simulated annealing approach [34, 35] were also applied in solving MDVRP.

However, few existing literatures pay attention to large-scale MDVRP (LSMDVRP, with customers more than 150), and the effectiveness of its solving methods should be discussed further. Framing link (FL) introduced in the following parts is helpful to reduce the searching space effectively and is a new direction in optimizing LSMDVRP. In fact, some scholars have accepted similar concepts in studying VRP; for example, Tarantilis and Kiranoudis [36] presented an adaptive memory based method for solving the Capacitated Vehicle Routing Problem (CVRP), called bone route. Zhong [37] proposed the concept and principium of kernel route, and a tabu search algorithm was designed to solve open Vehicle Routing Problem (OVRP) with capacity and distance limits. In this paper, the authors will combine the FL with a tabu search in solving LSMDVRP.

This paper is organized as follows. In Section 2, the authors propose the principle of FL for LSMDVRP and the structure of tabu algorithm, and procedure of the method is described in Section 3. Then, the authors measure algorithm sensitivity and the relationship between FL and optimization results and then compare the result of the proposed method with those in references, which shows that the former is performed. Conclusions occur at the end.

2. LSMDVRP Framing Link

2.1. Principle of FL and Structure of Optimization. When an intelligent algorithm is applied in the optimization of vehicle routes, iteration is conducted based on the previous generation solutions. As a result, there are numerous links between solutions of neighbor generations, and these links are updated continuously with iterations; good links (in optimized solutions) are kept and bad ones are decomposed or combined with others. In the procedure of a so-called “good” algorithm to VRP, more good links are generated. As shown in Figures 1 and 2, the routes are achieved through iterations of a tabu search for MDVRP case p10 in Gillett and Johnson [16], and their solution values are 3714.28 and 3647.22, respectively.

The two solutions shown above occur at the generation 1233 and generation 1671, respectively, in the tabu search. Although there are over 400 generations between the two

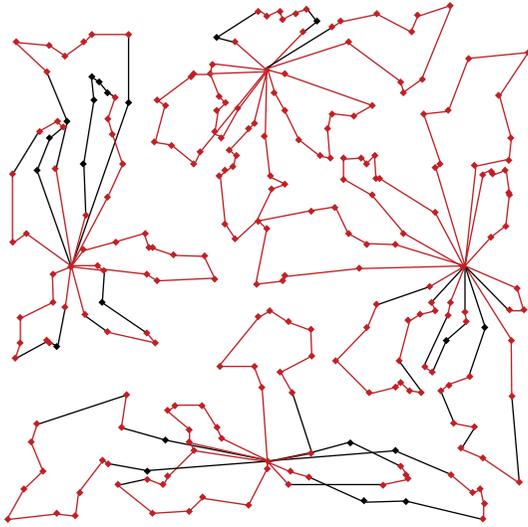


FIGURE 1: The route with a solution value 3731.38.

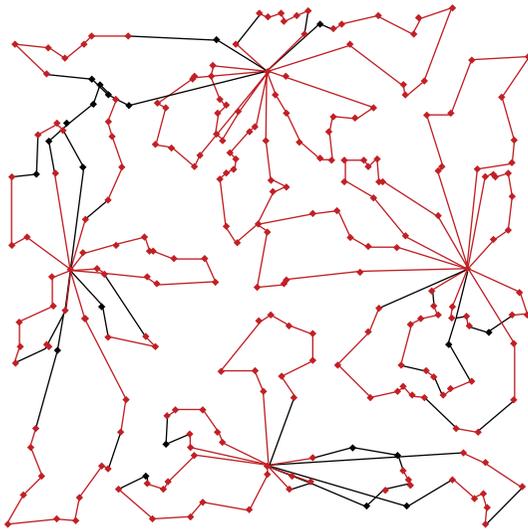


FIGURE 2: The route with a solution value 3647.22.

solutions, their structures have many similar parts as illustrated with red lines in Figures 1 and 2. In those routes with less generation gap, more similar links can be found. It is possible to optimize the VRP solutions based on these kinds of links as the skeleton, so as to obtain better results. In this paper, links that occur in both the optimal solution and suboptimal solutions are named as FLs.

Generation of framing links and update are keys of the FL based algorithm for LSMDVRP. The authors construct a FL base, which consists of links with higher frequencies in good routes in iterations. Addition and deletion of links are determined with the update condition. It is necessary to allocate vertexes to depots so as to generate LSMDVRP FLs, but, for those collection nodes close to several depots, possible allocations of them to different depots in optimization can result in generating unstable FLs and hence prevent them from entering the FL base. Consequently, the authors specify

a FL tabu area among depots, where the generating principles of FLs are more rigorous. The basic structure of FL based tabu algorithm for LSMDVRP is described in Figure 3.

2.2. *Update Condition of Framing Links.* If the requirement is satisfied, an update occurs in the FL base and the generated links are added into the base; meanwhile, those links without high usage frequency are deleted. In this study, the update condition of FLs includes:

- (a) the update of FLs has iterated N_d times;
- (b) the local optimal solution keeps unchanged for N_i generations;
- (c) all local and global optimums are less than B_q after N_q generations since the last update.

2.3. Generation of New-Entry Links

2.3.1. *Generation of Candidate Links.* FLs are generated with the continuous and large-scale optimization of current solutions in the procedure of the tabu algorithm. Compared with the premutation route, the postmutation one has different potential FLs. If these links are in the optimal solution during the current optimization period (one optimization period means the duration from updating the parameters of FLs to satisfying the next FL update condition), they are candidate links to enter the FL base. The entry procedure includes the following steps.

- (a) Identify the optimal vertex and those local optimal vertexes which have difference of less than Δs_{\max} with the current optimal vertex in solution value. Figure 4 shows the profile of parts of current solutions in an optimization period, where the optimal values of vertexes A, B, and C are s_A , s_B , and s_C , respectively. B is the optimal vertex in this period, and $(s_A - s_B)/s_B < \Delta s_{\max}$ and $(s_C - s_B)/s_B < \Delta s_{\max}$.
- (b) Recognize all local optimal vertexes and the local worst solution before the optimum occurs. As shown in Figure 4, the local worst solution vertexes before A, B, and C are A' , B' , and C' , respectively.
- (c) Compare routes of local optimums and the periodical optimum with routes of their corresponding worst vertexes, and then choose different links among them. Define $\Omega_{k,i}$, $i = 1, 2, \dots, n_L^k$, as the set of different links between the i th local optimal route and its corresponding worst route in the optimization period k and n_L^k as the number of local optimal solutions (including the periodical optimum) in the period k . For example, in Figure 4, comparing routes of A and A' , different links between them can be found. Similarly, different links between B and B' and between C and C' also can be obtained. Consider $\Omega_k = \{\Omega_{k,1}, \Omega_{k,2}, \dots, \Omega_{k,n_L^k}\}$.
- (d) Set the corresponding link set of the optimal solution in the optimization period k as Ω_B^k and the set of

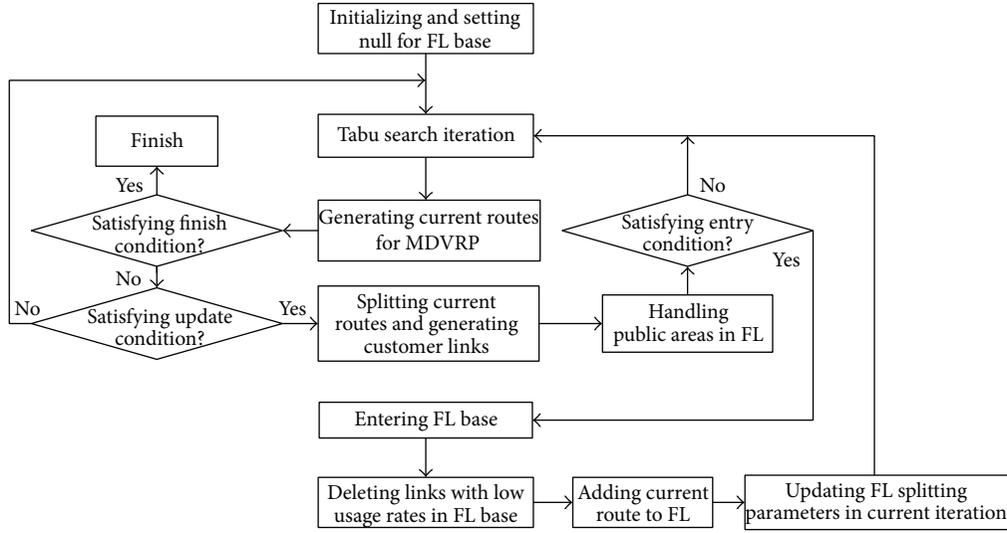


FIGURE 3: The basic structure of FL based tabu algorithm for LSMDVRP.

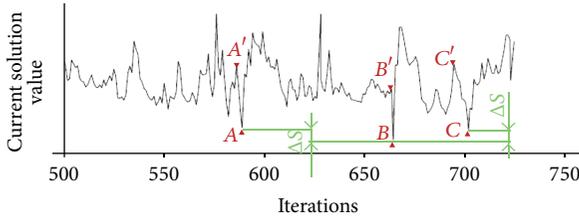


FIGURE 4: Generation of framing links.

candidate links to the FL base as $\Omega'_k, \Omega'_k = \{l \mid l \in \Omega_k, l \in \Omega'_B\}$. Different parts between A and A' , B and B' , and C and C' are extracted and compared, and links in the route of B are the candidate links to be added in the FL base.

2.3.2. Selection of Links in the Sharing Area for the Base. In Figure 5, the area A with depot i as the center and $R_{i,j}$ ($i, j \in V_d, i \neq j$) as the radius is defined, and V_d is the total collection of all depots. The sharing area of two depots is $A_i \cap A_j, i, j \in V, i \neq j$, as shown in blue, red, and green shadow area. For vertexes outside the sharing area of two depots, the rule of generating candidate links in Section 2.3.1 is inappropriate.

Principles to generate candidate links for the FL base in the sharing area include the following.

- If all vertexes in a link are located in the sharing area of $A_i \cap A_j, i, j \in V_d, i \neq j$ (Link Type 1), this link cannot be taken as a candidate link, as shown in Figure 6.
- If vertexes of a link are spread in all following areas including $A_i \cap A_j, A_i, A_j, i, j \in V_d, i \neq j$ (Link Type 2), this link cannot be taken as a candidate link, as shown in Figure 7.
- If vertexes of a link are located in A_i and $A_i \cap A_j$, respectively, $i, j \in V_d, i \neq j$ (Link Type 3), this link can be a candidate link, as shown in Figure 8. If a link

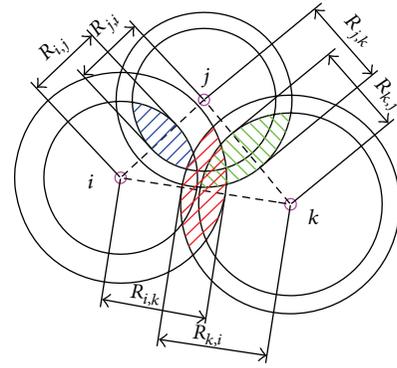


FIGURE 5: Illustration of the sharing area.

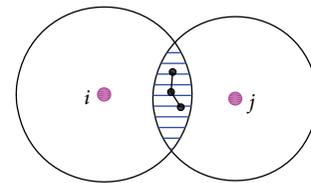


FIGURE 6: Link Type 1.

of Type 3 is added into the FL base, it can survive until the next N_c th generation and then will be inspected whether to be kept in the FL base or not. Meanwhile, the borderline of the sharing area will be adjusted with the rules in Section 2.3.3.

2.3.3. Adjusting Rules of Sharing Area. After the sharing area is initialized, its size and location will change with iterations, following the adhering rules.

- Sharing area initialization: $R_{i,j} = \delta_{i,j} d_{i,j}, i, j \in V_d, i \neq j, \delta_{i,j}$ is the borderline parameter of the sharing

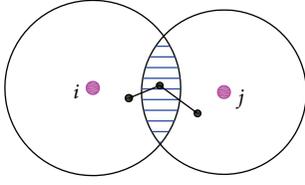


FIGURE 7: Link Type 2.

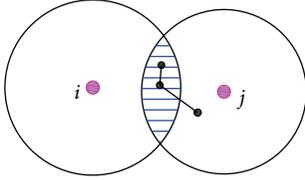


FIGURE 8: Link Type 3.

area, $d_{i,j}$ is the distance from depot i to depot j , and V_d is the collection of all depots.

- (b) Adjusting rules of sharing area: since $d_{i,j}$ and $d_{j,i}$ remain constant, the area of $A_i \cap A_j$ is actually determined by $\delta_{i,j}$ and $\delta_{j,i}$ together. If there is a link of Type 3, the borderline of the sharing area will be adjusted. That means that if vertexes are located in the two areas A_i and $A_i \cap A_j$, $i, j \in V_d$, $i \neq j$, and this link is added to the current route after N_c generations and the current solution value is better than that before the N_c th generation, $\delta_{j,i} := \delta_{j,i} + \Delta\delta$ and $\Delta\delta$ is the updating step of the sharing area borderline coefficient; if the link is added to the current route after N_c generations but the current solution value is no better than that before the N_c th generation, $\delta_{j,i} := \delta_{j,i} - \Delta\delta$, as shown in Figure 9.

2.3.4. Selection of Qualified Links into the FL Base. Selection of qualified links into the FL base follows several principles.

- Amount restriction: a number of vertexes to enter the FL base are restricted into the range of $[L_{\min}, L_{\max}]$; ones out of this range are excluded.
- Minimized split: for those routes to be added in the base, except that the complete routes are kept, they also should be decomposed into links connecting L_{\min} vertexes. Considering the case 3-7-4-1, $L_{\min} = 2$, this route needs to be decomposed into six FLs: 3-7-4-1, 3-7-4, 7-4-1, 3-7, 7-4, and 4-1.
- Backward generation: links in the FL base occur in pairs with opposite directions, so, once a link is selected into the FL base, its opposite link is generated synchronously. For example, if the link 3-7-4-1 is added, then a link 1-4-7-3 is generated.
- Entry: based on the minimized split results of an optimized route, If this route does not exist in the FL base, it will be added in.
- Update: once an achieved FL has more than L_{\min} vertexes, it should be updated, as well as its subroutes.

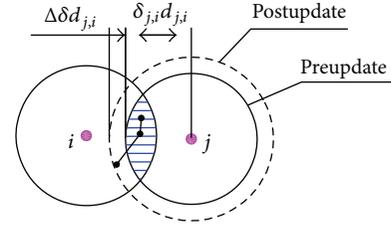


FIGURE 9: Update of the public area.

In above case, the occurrence frequencies and corresponding optimal solution values of 3-7-4-1, 3-7-4, 7-4-1, 3-7, 7-4, and 4-1 should be updated.

2.4. Deletion Principle in the Framing Link Base

- Comparison between links split from the current route is conducted every η iterations. After k generations, links before generation d have been compared for $c = [k/\eta]$ times. Set the usage frequency of a link as $u(c)$. If $u(c) < u_s(c)$, this link is deleted from the FL base. Here, $u_s(c)$ is the lower limit of FL usage, as a function of comparison times c .
- If the link is included in the current optimal solution, even if its usage is less than $u_s(c)$, it cannot be deleted.

2.5. Adding Framing Links into the Current Route. In the FL base, some link parameters are set: the optimal solution value of the route generated based on current links, the value of the worst solution, the average value of solutions, the usage times, the usage frequency, and the depot. Steps of adding FLs into the current route include the following.

- Computation of the link adaptability $f_i = 1/s_i^*$, $i = 1, 2, \dots, N_j$; N_j is the amount of links in the base, and s_i^* is the optimal solution value of the route where the i th link lies in. Set H as the collection of links in the FL base, and $H' = H$, $H^* = \Phi$.
- Judgment on H' : if it is equal to Φ , the procedure ends.
- Selection of link l_j with a roulette according to the value of f_i , $l_j \in H'$: set $H^* = H^* + \{l_j\}$.
- Calculation with $L = \{l_h \mid l_h \in H', \text{ there is } v \in l_j \text{ and } v \in l_h\}$, where v is a vertex included in the link: if $H' = H' - L$, go back to step 2.

3. Framing Link Based LSMDVRP Tabu Search Algorithm

In this study, the algorithm for MDVRP consists of two parts: initial optimization and follow-up optimization. Initial optimization is based on links extracted from the FL base, and extracted link is regarded as a vertex, not to be decomposed, so as to maximize FLs' advantages in generating optimized routes. On the other hand, follow-up optimization splits

links individually, excluding non-FLs so as to avoid inferior solutions.

3.1. Initial Optimization

3.1.1. p -Neighborhood. The p -neighborhood of a vertex: The p vertexes or links in the nearest collection A of vertex v are the p -neighborhood of v , denoted by $N_p(v, A)$. If the element in $N_p(v, A)$ is a vertex, it is illustrated with v' , and if it is a link, it is described with the link $v_1^s(z_{1,v}^s) - v_1^s(z_{1,v}^s) - v_1^s(z_{1,v}^s) - \dots - v_1^s(z_{1,v}^s)$ in the neighborhood, where v_1^s represents the start vertex of link l and v_1^s represents the ending vertex. Furthermore, that $z_{1,v}^s = 0$ means that v_1^s is not in the p neighborhood of vertex v ; otherwise, $z_{1,v}^s = 1$. As showed in Figure 10, if $p = 2$, $N_2(4) = \{8, 3(1) - 2(1) - 1(0)\}$ and $N_2(7) = \{9(1) - 10(1) - 11(0), 12(1) - 13(0)\}$.

The p -neighborhood of a link: p vertexes or links in the nearest collection A of the start vertex v_1^s of link l are the p -neighborhood of link l , represented with $N_p^s(l, A)$. At the same time, p vertexes or links in the nearest collection A of the link l are the p -neighborhood of the ending vertex v_1^e of link l , represented with $N_p^e(l, A)$.

3.1.2. Insertion Method. For MDVRP with the coexistence of vertexes and links, there are three kinds of insertions: (a) insertion between two vertexes, (b) insertion between two links, and (c) insertion between a vertex and a link. The insertion method is similar to traditional insertion method and the only difference is that this method treats the link as a node. The specific content can refer to Solomon [38].

3.1.3. Generation of the Initial Solution. The generation of the initial solution includes the following steps:

Step 1. Allocate vertexes and links to initial depots. For a point, the nearest depot is regarded as the initial depot; for a link, the initial depot is the one where its initial route is included. $N_r(c_h)$ represents the collection of all unallocated vertexes of the depot h , $G(c_h, r)$ represents the total delivery amount of the r th route of the h th depot c_h , $G_{\max}(c_h, r)$ represents the delivery limit of vehicles in the r th route of the depot c_h , $g(v_k \text{ (or } l_k))$ represents the delivery amount of vertex v_k or link l_k , and N_C is the amount of depots.

Step 2. Set $h = 0$.

Step 3. Consider $h := h + 1$.

Step 4. If $h > N_C$, go to Step 8.

Step 5. Set A_{c_h} as the collection of all vertexes or links belonging to the depot c_h , $\forall v_i \text{ (or } l_i) \in N_p(c_h, A_{c_h})$, and $v_j \text{ (or } l_j) \in N_p(c_h, A_{c_h})$. For the r th route $R(c_h, r)$ of the depot c_h , $c_h - v_i \text{ (or } l_i) - v_j \text{ (or } l_j) - c_h$. $G(c_h, r) = g(v_i \text{ (or } l_i)) + g(v_j \text{ (or } l_j))$, $N_r(c_h) := N_r(c_h) - v_i \text{ (or } l_i) - v_j \text{ (or } l_j)$.

Step 6. For all two neighboring vertexes and links $v_i \text{ (or } l_i) - v_j \text{ (or } l_j)$ in the route r , $v_k \text{ (or } l_k) = \min\{v \text{ (or } l) \mid \Delta d(v_i \text{ (or } l_i), v \text{ (or } l), v_j \text{ (or } l_j))\}$. If $G(c_h, r) + g(v_k \text{ (or } l_k)) >$

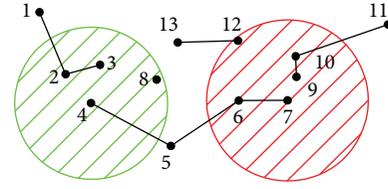


FIGURE 10: p -neighborhoods of vertexes and links.

$G_{\max}(h, r)$, update $r := r + 1$ and turn to Step 5. Otherwise, insert $v_k \text{ (or } l_k)$ to form a route of $v_i \text{ (or } l_i) - v_k \text{ (or } l_k) - v_j \text{ (or } l_j)$.

Step 7. Update $G(c_h, r) := G(c_h, r) + g(v_k \text{ (or } l_k))$, $N_r(c_h) := N_r(c_h) - v_k \text{ (or } l_k)$. If $N(c_h) = \Phi$, turn to Step 3; otherwise, turn to Step 6.

Step 8. End.

3.1.4. Construction of Neighborhood. According to the characteristics of FL MDVRP, the authors introduce three neighborhood operators: insertion, interchange, and crossover.

Insertion. α is a random number in the range of $[\alpha_{\min}, \alpha_{\max}]$. In the route $R(c_h, r)$, if capacity constraint of $R(c_h, r)$ is satisfied, α vertexes or links are randomly selected in r , as $v_i \text{ (or } l_i)$ to $v_j \text{ (or } l_j)$. $A_{\bar{R}(c_h, r)}$ is set as the collection of vertexes or links outside the route $R(c_h, r)$, and, according Section 3.1.2, $v_k \text{ (or } l_k) \in A_{\bar{R}(c_h, r)}$ is selected.

Interchange. θ is a random number in the range of $[\theta_{\min}, \theta_{\max}]$, as well as φ in $[\varphi_{\min}, \varphi_{\max}]$. Choose two routes: $R(c_h, r_1)$ and $R(c_h, r_2)$, and set $A_{R(c_h, r_1)}$ and $A_{R(c_h, r_2)}$ as the collection of vertexes and links in $R(c_h, r_1)$ and $R(c_h, r_2)$. θ vertexes and links in $R(c_h, r_1)$ are selected randomly, and then $v_m \text{ (or } l_m) \in A_{R(c_h, r_2)}$ is obtained based on the method in Section 3.1.2, $m = 1, 2, \dots, \theta$. Similarly, φ vertexes and links in $R(c_h, r_2)$ are selected. If neither capacity constraint of $R(c_h, r_1)$ nor that of $R(c_h, r_2)$ is unsatisfied after insertions, $v_m \text{ (or } l_m)$ can be inserted into the corresponding place in $A_{R(c_h, r_1)}$, or $v_n \text{ (or } l_n)$ in $A_{R(c_h, r_2)}$. Otherwise, $v_m \text{ (or } l_m)$ and $v_n \text{ (or } l_n)$ will be selected again.

Crossover. γ_1 and γ_2 are random numbers in the range of $[\gamma_{\min}, \gamma_{\max}]$, where $\gamma_1 < \gamma_2$, and the random number λ is in the range of $[\lambda_{\min}, \lambda_{\max}]$. In the route of $R(c_h, r_1)$, a segment s_1 from γ_1 to γ_2 is extracted. Meanwhile, segment $s^* = \{s \mid \min d(c_{s_1}, c_s)\}$ with a distance of λ is selected in a neighboring route of $R(c_h, r_1)$, where $d(c_{s_1}, c_s)$ represents the distance from the center of segment s_1 to the center of the whole route s . If the exchange of s_1 and s^* to the counterpart routes cannot lead to the capacity unconstraint of these two routes, this transform is allowed; otherwise, new γ_1 and γ_2 need to be selected.

3.2. Follow-Up Optimization. Although the FL method prompts to generate more desired solutions, FLs in the initial

TABLE 1: Column headings for Tables 2–10.

I	Instance ID
n	Number of customers
m	Number of depots
Q	Vehicle capacity
D	Maximum duration of a route
$c(s^*)$	Previous best known solution cost
$c(s^{**})$	Best known solution found by proposed tabu search algorithm on FL
L_{\min}, L_{\max}	Min and max amount limitation for vertexes to be added into the FL base
$\delta_{i,j}$	Borderline parameter of the sharing area
$\Delta\delta$	Updating step of $\delta_{i,j}$
Δs_{\max}	Max difference between optimal vertex and local optimal vertexes in an optimization period
n_Y	Number of optimization periods
n_L	The total number of local optimal vertexes
Ω'	The total number of candidate links
ρ_1^i	The proportion of customers in FLs to total customers in optimization period i
ρ_2^j	The proportion of customers in FLs to total customers in local optimal vertex j
ω_1^i	The optimal solution value of optimization period i
ω_2^j	The value local optimal solution j

TABLE 2: MDVRP instances and previous best known solution cost.

I	source	n	Q	D	m	GJ	CGW	CGL	$c(s^*)$
p08	GJ	249	500	310	2	4832.0	4511.6	4437.68	4437.68
p09	GJ	249	500	310	3	4219.7	3950.9	3900.22	3900.22
p10	GJ	249	500	310	4	3822.0	3727.1	3663.02	3663.02
p11	GJ	249	500	310	5	3754.1	3670.2	3554.18	3554.18
p15	CGW	160	60	0	4	—	2610.3	2505.42	2505.42
p16	CGW	160	60	200	4	—	2605.3	2572.23	2572.23
p17	CGW	160	60	180	4	—	2816.6	2709.09	2709.09
p18	CGW	240	60	0	6	—	3877.4	3702.85	3702.85
p19	CGW	240	60	200	6	—	3863.9	3827.06	3827.06
p20	CGW	240	60	180	6	—	4272.0	4058.07	4058.07
p21	CGW	360	60	0	9	—	5791.5	5474.84	5474.84
p22	CGW	360	60	200	9	—	5857.4	5702.16	5702.16
p23	CGW	360	60	180	9	—	6494.6	6095.46	6095.46
pr04	CGL	192	185	440	4	—	—	2072.52	2072.52
pr05	CGL	240	180	420	4	—	—	2385.77	2385.77
pr06	CGL	288	175	400	4	—	—	2723.27	2723.27
pr09	CGL	216	180	450	6	—	—	2153.1	2153.1
pr10	CGL	288	170	425	6	—	—	2921.85	2921.85

optimization are not necessarily parts of routes in the optimized solution; consequently, follow-up operation is needed to iterate continuously after splitting links into individual vertexes. Once the optimization solution is not yet updated ξ generations after the initial optimization, follow-up optimization should be introduced. The initial solution of the follow-up optimization is the optimal one of the initial optimization, and if this initial solution has occurred in previous periods, the second optimal solution of the initial optimization can be selected.

4. Computational Experiments

The algorithm proposed above is programmed in MS Visual C++ 6.0 and tested with a PC with a AMD Athlon (tm) X2 2.0 GHz CPU and 2 GB RAM. In this paper, the authors solve problems in existing literature with this new algorithm and then solved different LSMDVRPs with different parameters, in order to identify the coefficient valuing discipline.

The authors test the algorithm based on 18 instances from literature. Instances of p8–p11 were provided in Gillett and

TABLE 3: Sensitivity data of parameters L_{\min} and L_{\max} .

I	n	m	$[L_{\min}^*, L_{\max}^*]$	Solution value % gap	CPU time % gap
p08	249	2	[3, 7]	3.37	403.23
p09	249	3	[3, 6]	4.6	476.87
p10	249	4	[3, 6]	7.62	322.33
p11	249	5	[3, 5]	6.26	398.74
p15	160	4	[2, 4]	5.06	366.86
p16	160	4	[2, 4]	3.01	485.78
p17	160	4	[2, 4]	3.1	573.27
p18	240	6	[2, 4]	6.53	397.2
p19	240	6	[2, 4]	4.61	676.61
p20	240	6	[2, 4]	7.85	238.97
p21	360	9	[2, 4]	8.18	291.3
p22	360	9	[2, 4]	6.76	334.59
p23	360	9	[2, 4]	6.14	618.93
pr04	192	4	[2, 5]	7.57	474.76
pr05	240	4	[3, 6]	3.17	356.43
pr06	288	4	[3, 6]	4.31	649.79
pr09	216	6	[2, 4]	7.55	414.62
pr10	288	6	[2, 4]	3.62	689.48

TABLE 4: Sensitivity data of parameter $\delta_{i,j}$ and its update step $\Delta\delta$.

I	n	m	$\delta_{i,j}^*$	$\Delta\delta^*$	Solution value		CPU time	
					% gap 1	% gap 2	% gap 3	% gap 4
p08	249	2	0.85	0.10	11.54	1.51	11.81	29.2
p09	249	3	0.75	0.08	9.82	1.79	19.13	34.48
p10	249	4	0.65	0.04	4.35	1.57	10.83	30.79
p11	249	5	0.65	0.04	17.68	1.2	16.61	51.14
p15	160	4	0.65	0.04	3.24	1.69	7.99	26.85
p16	160	4	0.65	0.06	16.32	1.95	8.68	41.14
p17	160	4	0.65	0.04	9.49	1.43	14.09	45.51
p18	240	6	0.7	0.06	9.59	2.59	12.67	48.92
p19	240	6	0.65	0.04	18.23	1.01	19.57	37.92
p20	240	6	0.65	0.06	13.66	1.99	17.61	44.74
p21	360	9	0.65	0.04	6.95	2.84	4.9	22.24
p22	360	9	0.65	0.04	8.49	1.9	19.68	57.11
p23	360	9	0.65	0.04	10.8	2.55	15.73	21.33
pr04	192	4	0.65	0.04	18.3	2.52	8.75	50.53
pr05	240	4	0.7	0.06	11.59	1.59	14.52	25.42
pr06	288	4	0.75	0.08	12.44	1.13	13.44	35.39
pr09	216	6	0.6	0.04	6.96	1.73	19.96	46.54
pr10	288	6	0.65	0.04	17.1	2.07	12.92	57.18

% Gap 1: % Gap between the worst and the best solution value with different $\delta_{i,j}$ and $\Delta\delta = 0.04$.

% Gap 2: % Gap between the worst and the best solution value with different $\Delta\delta$ and $\delta_{i,j}^*$.

% Gap 3: % Gap between the longest and the shortest CPU time with different $\delta_{i,j}$ and $\Delta\delta = 0.04$.

% Gap 4: % Gap between the longest and the shortest CPU time with different $\Delta\delta$ and $\delta_{i,j}^*$.

Johnson [16], p15–p23 by Chao et al. [39], and pr04–pr6, pr9, and pr10 by Cordeau et al. [25]. Table 1 is the column headings for Tables 2, 3, 4, 5, 6, 7, 8, 9, and 10.

The characteristics of the instances are summarized in Table 2 and the complete data sets and best known results are given in the website <http://neo.lcc.uma.es/radi-aeb/WebVRP/index.html>. The costs of the best solutions found by each method are listed in Table 2.

4.1. Sensitivity Analyses

4.1.1. $[L_{\min}, L_{\max}]$. Firstly, all other parameters are set as constants. L_{\max} is selected from (3, 4, 5, 6, 7) and L_{\min} is from (2, 3, 4), given $L_{\max} \geq L_{\min}$. Each of the 18 instances is calculated for 14 times, and the results are listed in Table 3.

It is not hard to find that the value of $[L_{\min}, L_{\max}]$ has not much influence on solutions, and average gap between

TABLE 5: The value range of $\delta_{i,j}^*$ and $\Delta\delta^*$.

n/m	$\delta_{i,j}^*$	$\Delta\delta^*$
≤ 50	[0.6, 0.7]	[0.04, 0.06]
≤ 100	(0.7, 0.8]	(0.06, 0.08]
> 100	(0.8, 0.9]	(0.08, 0.10]

TABLE 6: The calculation results of parameter Δs_{\max} .

I	n	m	Δs_{\max}^*	n_Y	n_L	Ω'
p08	249	2	0.03	3	17	201
p09	249	3	0.04	5	18	193
p10	249	4	0.03	7	21	187
p11	249	5	0.03	7	42	422
p15	160	4	0.06	5	18	143
p16	160	4	0.07	5	18	159
p17	160	4	0.06	5	17	129
p18	240	6	0.05	8	31	255
p19	240	6	0.06	9	48	646
p20	240	6	0.05	10	47	531
p21	360	9	0.02	10	45	559
p22	360	9	0.02	11	46	315
p23	360	9	0.02	9	43	302
pr04	192	4	0.06	5	23	309
pr05	240	4	0.03	6	36	400
pr06	288	4	0.03	6	21	215
pr09	216	6	0.05	8	27	232
pr10	288	6	0.04	7	41	343

TABLE 7: The value of n_Y , n_L , and Ω' and the accuracies of Ω' with different Δs_{\max} in the instance p09.

Δs_{\max}^*	n_Y	n_L	Ω'	Ω' accuracy
0.01	3	5	58	37.77%
0.02	3	8	101	38.37%
0.03	4	11	131	40.71%
0.04	5	18	193	42.40%
0.05	9	23	287	32.05%
0.06	10	26	402	23.77%
0.07	12	27	606	21.96%
0.08	12	40	726	15.59%
0.09	13	55	958	10.02%
0.1	15	101	974	11.23%

solutions is 5.52%. Inferior solutions always occur in the condition where L_{\min} is relatively large and $L_{\min} = L_{\max}$; for example, $[L_{\min}, L_{\max}] = [5, 5]$. However, the value of $[L_{\min}, L_{\max}]$ greatly affects the computation speed of the algorithm, with an average gap of 453.88%, since L_{\max} can control the upper limits of lengths of candidate links, and L_{\min} determines their lower limits as well as split numbers. With the decrease in L_{\min} , the number of splits increases with an exponential distribution. According to this case, the relation of $[L_{\min}^*, L_{\max}^*]$ and n/m is revealed: a larger n/m leads to larger L_{\min}^* and L_{\max}^* . When the value of n/m is less

than 40, the commendation value of $[L_{\min}^*, L_{\max}^*]$ is [2, 4], when $40 < n/m \leq 100$, the value is [3, 5] or [3, 6], and when n/m is more than 100, the value is [3, 7].

4.1.2. *Borderline Parameter $\delta_{i,j}$ and Update Step $\Delta\delta$.* Keeping other parameters unchanged, the authors measure the effect of $\delta_{i,j}$ here. Firstly, $\Delta\delta = 0.04$, $\delta_{i,j}$ is in [0.5, 0.9], the interval between successive $\delta_{i,j}$ is 0.05, and then $\delta_{i,j}^*$ for 18 instances are obtained through 9 times of calculations for each instance. Next, $\Delta\delta$ is taken varied in [0.02, 0.12], $\delta_{i,j}^*$ are allocated to these instances, and the interval for $\Delta\delta$ is defined as 0.02. $\Delta\delta^*$ for 18 instances are achieved through 6 times of calculations for each one.

According to Table 4, given $\Delta\delta$ is a constant, difference between the best and the worst solution values of different $\delta_{i,j}$ is 11.48%. If $\delta_{i,j}$ is regarded as a constant, difference between the best and the worst solution values of different $\Delta\delta$ only is 1.84%. Such results illustrate that valuing of $\delta_{i,j}$ has a more obvious impact on the computation result of the algorithm and that of $\Delta\delta$. Similarly, valuing of $\delta_{i,j}$ is much more influential on the computation efficiency of the algorithm (average difference is 39.25%) than that of $\Delta\delta$ (average difference is 13.83%). The values of $\delta_{i,j}^*$ and $\Delta\delta$ are proportional to the customer numbers allocated in each depot (n/m). The approximate value range of $\delta_{i,j}^*$ and $\Delta\delta^*$ is shown in Table 5.

4.1.3. Δs_{\max} . Set $[L_{\min}, L_{\max}] = [L_{\min}^*, L_{\max}^*]$, Δs_{\max} in [0.01, 0.10], and the interval between different Δs_{\max} as 0.01. The authors calculate 10 times for each instance, the number of optimization periods is n_Y , the number of local optimal vertexes in optimization period k is n_L^k , $n_L = \sum_{k=1}^{n_Y} n_L^k$, Ω'_k is the number of candidate links produced in optimization period k , and $\Omega' = \sum_{k=1}^{n_Y} \Omega'_k$. The calculation results are shown in Table 6.

According to the analysis results, Δs_{\max} can control the number of local optimal vertexes n_L , and the value of Δs_{\max} is inversely proportional to n . This is because global optimization is difficult with the increase in n and more local optimal vertexes attained. As a result, a smaller Δs_{\max} is needed to restrict the number of n_L , so as to control Ω' . Although a larger Ω' raises the possibilities to obtain FLs, it lowers the average quality of candidate links as well, and so some “bad” links corresponding to local optimal vertexes far from the periodical vertex are involved in the candidate link base. Once these unexpected links enter the FL base, the global optimization will be staggered and its results will be weakened. In this study, a proper Δs_{\max} for computation scale and characteristics is needed. Table 7 lists the value of n_Y , n_L , and Ω' and the accuracies of Ω' with different Δs_{\max} in the instance p09.

As shown in Table 7, when Δs_{\max} is at its best value Δs_{\max}^* , Ω' is the most accurate, so as to guarantee the most efficient and effective optimization. Based on this study, the recommendation value of Δs_{\max}^* is [0.06, 0.07] when $150 < n \leq 200$, [0.03, 0.06] when $200 < n \leq 300$, and [0.02, 0.03] when $300 < n \leq 400$.

TABLE 8: Numbers of FLs in every optimization period.

I	Optimization period										
	1	2	3	4	5	6	7	8	9	10	11
p08	20	34	38	—	—	—	—	—	—	—	—
p09	13	16	24	28	30	—	—	—	—	—	—
p10	11	15	19	26	30	33	34	—	—	—	—
p11	15	17	19	23	28	30	33	—	—	—	—
p15	12	18	22	26	27	—	—	—	—	—	—
p16	15	17	25	31	31	—	—	—	—	—	—
p17	16	21	25	27	27	—	—	—	—	—	—
p18	15	18	19	23	28	32	32	33	—	—	—
p19	13	18	22	27	32	37	37	38	40	—	—
p20	10	13	12	17	23	27	32	34	36	36	—
p21	11	15	19	27	31	35	38	45	48	50	—
p22	12	17	21	27	33	39	46	51	52	52	53
p23	12	15	19	25	33	43	47	54	56	—	—
pr04	14	21	25	24	25	—	—	—	—	—	—
pr05	16	25	34	39	39	40	—	—	—	—	—
pr06	15	19	25	32	36	38	—	—	—	—	—
pr09	13	18	23	27	31	32	33	35	—	—	—
pr10	19	22	29	34	34	35	37	—	—	—	—

TABLE 9: The correlation of ρ_1^i , ω_1^i , ρ_2^j , and ω_2^j .

I	n_Y	n_L	Correlation coefficient of ρ_1^i and ω_1^i	Correlation coefficient of ρ_2^j and ω_2^j
p08	3	17	-0.92	-0.91
p09	5	18	-0.91	-0.99
p10	7	21	-0.97	-0.96
p11	7	42	-0.97	-0.97
p15	5	18	-0.89	-0.89
p16	5	18	-0.97	-0.90
p17	5	17	-0.88	-0.93
p18	8	31	-0.95	-0.90
p19	9	48	-0.92	-0.95
p20	10	47	-0.93	-0.98
p21	10	45	-0.88	-0.95
p22	11	46	-0.90	-0.96
p23	9	43	-0.91	-0.93
pr04	5	23	-0.96	-0.91
pr05	6	36	-0.90	-0.89
pr06	6	21	-0.96	-0.95
pr09	8	27	-0.90	-0.92
pr10	7	41	-0.92	-0.90

4.2. *FLs and Optimization Results.* With the best parameters in Section 4.1, the generation process of FLs in the optimization is tracked. Set ρ_1^i as the proportion of customers in FLs to total customers in each optimization period, ω_1^i as the optimal solution value of optimization period i , $i = 1, 2, \dots, n_Y$, ρ_2^j as the proportion of customers in FLs to total customers in each local optimal vertex, and ω_2^j as the value local optimal solution j , $j = 1, 2, \dots, n_L$. Table 8 shows numbers of FLs in every optimization period.

In Tables 8 and 9, the following can be found. (a) Average numbers of FLs increase with the progress of optimization, to the most in the last optimization period, and the optimal solution is achieved at the same time. (b) Increasing speed of FL numbers falls down gradually in the optimization process. Due to the existence of sharing area, when the proportion of customers on FLs reaches a certain number, the increase in its absolute amount will be slowed. (c) The optimal solution keeps updating with optimization. Although FLs do not increase rapidly, the optimal search ability is not challenged.

TABLE 10: The results found by proposed tabu search algorithm on FL.

I	n	$c(s^*)$	$c(s^{**})$	Average CPU time (min)	% CPU time	Average solution value	% solution value	% above best known solution
p08	249	4437.68	4417.46	10.40	27.04	4465.69	1.08	0.46
p09	249	3900.22	3886.82	10.32	28.60	3937.76	1.29	0.34
p10	249	3663.02	3647.22	9.61	28.20	3684.78	1.02	0.43
p11	249	3554.18	3547.70	9.20	12.03	3611.16	1.76	0.18
p15	160	2505.42	2513.43	2.88	8.31	2528.42	0.59	-0.32
p16	160	2572.23	2578.88	4.04	9.03	2591.07	0.47	-0.26
p17	160	2709.09	2709.09	3.43	11.86	2710.19	0.04	0
p18	240	3702.85	3712.61	4.22	12.45	3719.78	0.19	-0.26
p19	240	3827.06	3827.06	3.73	12.58	3853.29	0.68	0
p20	240	4058.07	4066.85	5.33	18.03	4078.59	0.29	-0.22
p21	360	5474.84	5483.31	7.72	20.46	5518.1	0.63	-0.15
p22	360	5702.16	5708.12	8.05	11.57	5743.57	0.62	-0.10
p23	360	6095.46	6095.46	7.07	17.23	6136.23	0.66	0
pr04	192	2072.52	2058.80	9.63	32.54	2081.42	1.09	0.66
pr05	240	2385.77	2369.00	8.59	19.30	2405.35	1.51	0.70
pr06	288	2723.27	2718.11	13.25	19.24	2763.24	1.63	0.19
pr09	216	2153.1	2147.40	8.04	13.21	2182.06	1.59	0.26
pr10	288	2921.85	2865.31	10.88	22.79	2934.16	2.35	1.94

% CPU time: the gap in percentage between the average CPU time and the fastest CPU time among the ten runs.

% solution value: the gap in percentage between the average value of the solutions and the best solution value among the ten runs.

In the latter periods of optimization, low quality FLs are likely to be replaced by high quality ones. (d) the average correlation coefficient between ρ_1^i and ω_1^i reaches -0.924 , and the number between ρ_2^j and ω_2^j is -0.933 for all 18 instances. Such a result apparently bridges the proportion of customers in FLs and their corresponding local optimal solutions, so as to prove that the improvement of ρ_1^i and ρ_2^j prompts to achieve better optimal solutions.

4.3. Optimization Results. The 18 instances are calculated for 10 times based on parameters recommended above, respectively, and the optimization results are shown in Table 10. Figure 11 describes the corresponding route of the optimal solution for p09.

In 9 of all 18 instances, new optimal solutions are obtained. The average fluctuation value of optimal solution in 18 instances is 0.97%. This fluctuation value consolidates the independence of optimization results to initial solutions in the proposed algorithm, as well as its effectiveness and robustness.

Scales of all 18 instances are large enough ($150 < n \leq 360$), but the average computation duration is only 7.58 min with a little fluctuation rate of 18.03% in 10 calculations. All these results show the improvement of this algorithm in optimization efficiency.

For p15–p23, no better route is attained, and values of the average computation time, fluctuation rate of computation time, and optimal solutions are lower than the average one in the total 18 instances. The reason is that regular distributions

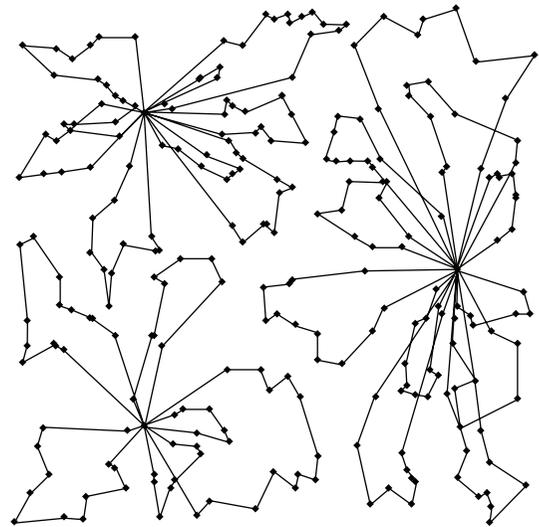


FIGURE 11: The corresponding route of the optimal solution for p09.

and similar requirements of customers in these instances reduce the optimization space and difficulty.

5. Conclusions

The authors propose a new tabu algorithm to optimize large-scale multidepot routes, using FLs as skeleton to improve search ability, speed up optimization, and then obtain better

solutions. The validity of the new algorithm has been verified by example of verification.

FL is a new concept proposed in view of the VRP. However, in terms of the present study level, it is hard to get an algorithm and recommended parameters applying to all VRP constraints because of the complexity of VRP. Studies of this paper only aimed at large-scale VRP and capacity constraints. For the VRP in other conditions, this paper has certain reference value on the way of thinking; however, the specific operation of operator, particularly the recommended value parameters, needs analysis case by case. In principle, the effectiveness of FL is the result of joint action of many factors and the designs and parameter values of different operators are not unique. As the research of FL under different constraint conditions of VRP and influence factors continues, more complete processing methods and recommended parameters will gradually form.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

Data of 18 instances is from the website <http://neo.lcc.uma.es/radi-aeb/WebVRP/index.html>. This research is supported by National Natural Science Foundation of China (no. 71371134). Meanwhile, the authors appreciate the help and comments from editors and reviewers.

References

- [1] C. A. Ullrich, "Integrated machine scheduling and vehicle routing with time windows," *European Journal of Operational Research*, vol. 227, no. 1, pp. 152–165, 2013.
- [2] M. S. Rasmussen, T. Justesen, A. Dohn, and J. Larsen, "The home care crew scheduling problem: preference-based visit clustering and temporal dependencies," *European Journal of Operational Research*, vol. 219, no. 3, pp. 598–610, 2012.
- [3] J. Wilck and T. Cavalier, "A genetic algorithm for the split delivery vehicle routing problem," *Physics & Mathematics*, vol. 2, pp. 207–216, 2012.
- [4] J. Riera-Ledesma and J. Salazar-González, "Solving school bus routing using the multiple vehicle traveling purchaser problem: a branch-and-cut approach," *Computers and Operations Research*, vol. 39, no. 2, pp. 391–404, 2012.
- [5] A. Lim, B. Rodrigues, and Z. Xu, "Transportation procurement with seasonally varying shipper demand and volume guarantees," *Operations Research*, vol. 56, no. 3, pp. 758–771, 2008.
- [6] R. Agarwal and Ö. Ergun, "Ship scheduling and network design for cargo routing in liner shipping," *Transportation Science*, vol. 42, no. 2, pp. 175–196, 2008.
- [7] B. Kim, S. Kim, and S. Sahoo, "Waste collection vehicle routing problem with time windows," *Computers and Operations Research*, vol. 33, no. 12, pp. 3624–3642, 2006.
- [8] S. S. Syam, "A multiple server location-allocation model for service system design," *Computers & Operations Research*, vol. 35, no. 7, pp. 2248–2265, 2008.
- [9] J. F. Campbell, A. T. Ernst, and M. Krishnamoorthy, "Hub arc location problems: part I: introduction and results," *Management Science*, vol. 51, no. 10, pp. 1540–1555, 2005.
- [10] J. K. Lenstra and A. H. G. Rinnooy Kan, "Complexity of vehicle routing and scheduling problem," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.
- [11] R. Hassin and S. Rubinstein, "On the complexity of the k -customer vehicle routing problem," *Operations Research Letters*, vol. 33, no. 1, pp. 71–76, 2005.
- [12] A. Imai, E. Nishimura, and J. Current, "A Lagrangian relaxation-based heuristic for the vehicle routing with full container load," *European Journal of Operational Research*, vol. 176, no. 1, pp. 87–105, 2007.
- [13] M. M. Solomon, "On the worst-case performance of some heuristics for the vehicle routing and scheduling problem with time window constraints," *Networks*, vol. 16, no. 2, pp. 161–174, 1986.
- [14] H. Hashimoto, T. Ibaraki, S. Imahori, and M. Yagiura, "The vehicle routing problem with flexible time windows and traveling times," *Discrete Applied Mathematics. The Journal of Combinatorial Algorithms, Informatics and Computational Sciences*, vol. 154, no. 16, pp. 2271–2290, 2006.
- [15] M. W. P. Savelsbergh, "Local search in routing problems with time windows," *Annals of Operations Research*, vol. 4, no. 1–4, pp. 285–305, 1985.
- [16] B. E. Gillett and J. G. Johnson, "Multi-terminal vehicle-dispatch algorithm," *Omega*, vol. 4, no. 6, pp. 711–718, 1976.
- [17] B. Golden, T. Magnanti, and H. Nguyen, "Implementing vehicle routing algorithms," *Networks*, vol. 7, pp. 113–148, 1977.
- [18] S. Salhi and M. Sari, "A multi-level composite heuristic for the multi-depot vehicle fleet mix problem," *European Journal of Operational Research*, vol. 103, no. 1, pp. 95–112, 1997.
- [19] R. T. Sumichras and I. S. Markham, "A heuristic and lower bound for a multi-depot routing problem," *Computers & Operations Research*, vol. 22, no. 10, pp. 1047–1056, 1995.
- [20] M. Wasner and G. Zäpfel, "An integrated multi-depot hub-location vehicle routing model for network planning of parcel service," *International Journal of Production Economics*, vol. 90, no. 3, pp. 403–419, 2004.
- [21] G. Nagy and S. Salhi, "Heuristic algorithms for single and multiple depot vehicle routing problems with pickups and deliveries," *European Journal of Operational Research*, vol. 162, no. 1, pp. 126–141, 2005.
- [22] A. Lim and F. Wang, "Multi-depot vehicle routing problem: a one-stage approach," *IEEE Transactions on Automation Science and Engineering*, vol. 2, no. 4, pp. 397–402, 2005.
- [23] M. Mirabi, S. M. T. F. Ghomi, and F. Jolai, "Efficient stochastic hybrid heuristics for the multi-depot vehicle routing problem," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 6, pp. 564–569, 2010.
- [24] C. Contardo and R. Martinelli, "A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints," *Discrete Optimization*, vol. 12, pp. 129–146, 2014.
- [25] J. Cordeau, M. Gendreau, and G. Laporte, "A tabu search heuristic for periodic and multi-depot vehicle routing problems," *Networks*, vol. 30, no. 2, pp. 105–119, 1997.
- [26] J. Cordeau, G. Laporte, and A. Mercier, "A unified tabu search heuristic for vehicle routing problems with time windows," *Journal of the Operational Research Society*, vol. 52, no. 8, pp. 928–936, 2001.

- [27] J. Renaud, G. Laporte, and F. F. Boctor, "A tabu search heuristic for the multi-depot vehicle routing problem," *Computers and Operations Research*, vol. 23, no. 3, pp. 229–235, 1996.
- [28] B. Crevier, J. Cordeau, and G. Laporte, "The multi-depot vehicle routing problem with inter-depot routes," *European Journal of Operational Research*, vol. 176, no. 2, pp. 756–773, 2007.
- [29] S. Belhaiza, P. Hansen, and G. Laporte, "A hybrid variable neighborhood tabu search heuristic for the vehicle routing problem with multiple time windows," in *Proceedings of the International Conference on Industrial Engineering and Systems Management (IESM '11)*, Metz, France, 2011.
- [30] S. Bae, H. S. Hwang, G. Cho, and M. Goan, "Integrated GA-VRP solver for multi-depot system," *Computers and Industrial Engineering*, vol. 53, no. 2, pp. 233–240, 2007.
- [31] W. Ho, G. T. S. Ho, P. Ji, and H. C. W. Lau, "A hybrid genetic algorithm for the multi-depot vehicle routing problem," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 4, pp. 548–557, 2008.
- [32] X. Wang, C. Xu, and H. Shang, "Multi-depot vehicle routing problem with time windows and multi-type vehicle number limits and its genetic algorithm," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '08)*, pp. 1–5, Dalian, China, October 2008.
- [33] Y. Li and X. Liu, "Modelling and its ant colony algorithm for multi-depot open vehicle routing problem with replenishment on the way," *Computer Integrated Manufacturing Systems*, vol. 14, no. 3, pp. 557–562, 2008.
- [34] T. Wu, C. Low, and J. Bai, "Heuristic solutions to multi-depot location-routing problems," *Computers and Operations Research*, vol. 29, no. 10, pp. 1393–1415, 2002.
- [35] S. Chen, A. Imai, and B. Zhao, "A SA-based heuristic for the multi-depot vehicle routing problem," *Journal of Japan Institute of Navigation*, vol. 113, pp. 209–216, 2005.
- [36] C. D. Tarantilis and C. T. Kiranoudis, "BoneRoute: an adaptive memory-based method for effective fleet management," *Annals of Operations Research*, vol. 115, pp. 227–241, 2002.
- [37] S. Zhong, "Open vehicle routing problem based on kernel route tabu search algorithm," *Computer Integrated Manufacturing Systems*, vol. 13, no. 4, pp. 827–832, 2007.
- [38] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research*, vol. 35, no. 2, pp. 254–265, 1987.
- [39] I. Chao, B. Golden, and E. Wasil, "A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions," *American Journal of Mathematical and Management Sciences*, vol. 13, pp. 371–406, 1993.

Research Article

Displacement Prediction of Tunnel Surrounding Rock: A Comparison of Support Vector Machine and Artificial Neural Network

Qingdong Wu,¹ Bo Yan,² Chao Zhang,² Lu Wang,³ Guobao Ning,⁴ and B. Yu⁵

¹ Shandong Luqiao Group CO., Ltd, Jinan 250021, China

² Transportation Management College, Dalian Maritime University, Dalian 116026, China

³ China Academy of Civil Aviation Science and Technology, Beijing 100028, China

⁴ School of Automotive Studies, Tongji University, Shanghai 201804, China

⁵ School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Guobao Ning; guobao.tj@163.com and B. Yu; yubjtt@126.com

Received 5 May 2014; Revised 13 July 2014; Accepted 14 July 2014; Published 23 July 2014

Academic Editor: Rui Mu

Copyright © 2014 Qingdong Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Displacement prediction of tunnel surrounding rock plays an important role in safety monitoring and quality control tunnel construction. In this paper, two methodologies, support vector machines (SVM) and artificial neural network (ANN), are introduced to predict tunnel surrounding rock displacement. Then the two modes are tested with the data of *Fangtianchong* tunnel, respectively. The comparative results show that solutions gained by SVM seem to be more robust with a smaller standard error compared to ANN. Generally, the comparison between artificial neural network (ANN) and SVM shows that SVM has a higher accuracy prediction than ANN. Results also show that SVM seems to be a powerful tool for tunnel surrounding rock displacement prediction.

1. Introduction

Limited by geographic condition, digging method, and many other factors, the loading features of tunnel construction are extremely complicated. As a result, the displacement of tunnel surrounding rock has random variations. Although the conditions of tunnel surrounding field monitoring could be gained by field monitoring, the field monitor required many resources that could not be afforded by constructors [1, 2]. Meanwhile, tunnel surrounding rock distortion and tunnel lining crack could also lead to the instability of the tunnel during tunnel construction [3]. So it is necessary to estimate the potential tunneling problems as early as possible.

Along with the development of information technologies, a number of intelligent algorithms, such as regression analysis, gray theory, data smoothing processing, time series analysis, and genetic algorithm [4–13], have been applied to predict the displacement of tunnel surrounding rocks [14, 15]. Till now there are many literatures that have studied

the displacement prediction of tunnel surrounding rock [16]. Parlos et al. [17] presented a multistep-ahead prediction for the systems of high complexity and a new method that is proved that it could be used to increase the accuracy of prediction relying on a dynamic recurrent neural network. Besides the dynamic model, Cheng et al. [18] also presented a multistep-ahead prediction model, which is not relying on a dynamic recurrent neural network but uses a method based on support vector machine (SVM) to predict and diagnose the type of long-term changes of hydropower structures. In the model, to diagnose the long-term changes of hydropower, a dynamic spline interpolation is established based on a multilayer adaptive time delayed network to ensure the prediction accuracy. In order to solve the nonlinear time series problems, Lee et al. [19] also proposed multistep-ahead model based on artificial neural network to improve the robust and accuracy of predicting in solving this type of problems. And, recently, Chevillon et al. [20] proposed that not only the multistep-ahead model, but also the nonparametric direct

multistep-ahead model could be used to predict the reliability of surrounding rock and other geographic structures. There are also some other studies that rely on multistep-ahead model to do the predicting [21, 22]. In general, in recent years, there are more and more researchers and institutions, which tend to use the multistep-ahead model to predict the complex internal structures.

Nevertheless, as many researchers stated, the accuracy of the predicting models based on the multistep-ahead model extremely relied on the similarity between the present and previous data. And considering the complexity of internal structure it is almost impossible to gain a large amount of accurate historical data. Therefore, with the unreliable historical data, the predicting results could also become seriously incorrect. Along with the development of intelligent rock mechanics, artificial neural network (ANN) technique is gradually matured and widely used. In recent years, ANN has been successfully applied in the field of tunnel surrounding rock displacement prediction. Meanwhile, thanks to the development of statistical learning theory, support vector machine (SVM), as a new machine learning technology, has drawn much attention in the research area of time series forecasting. Therefore, in the paper, two models, displacement prediction of tunnel surrounding rock based on ANN and displacement prediction of tunnel surrounding rock based on SVM, are presented. And then in order to compare the two models, in the last section the two models are evaluated with data of *Fangtianchong* tunnel, respectively. If the simple models presented in the paper could be applied to practice in the future, the building security and effectiveness could be definitely improved.

The objectives of this study are to examine and compare the feasibility of applying ANN and SVM to predict tunnel surrounding rock displacement. Thus, this paper is structured as follows: Section 2 provides a brief introduction on ANN and then practices ANN to predict the displacement of tunnel surrounding rock; Section 3 provides a simple introduction of SVM and then applies SVM to predict the displacement of tunnel surrounding rock; at the end of the paper, the conclusions are presented.

2. Artificial Neural Network (ANN) for Tunnel Surrounding Rock Displacement

Artificial neural network (ANN) is considered as one of the most effective tools that could be used to predict non-linear problems, especially the problem with a real world background [23]. The framework of artificial neural network (ANN) requires it to gain new data constantly; the characteristic of ANN could be used to track even a little variation repeatedly. Besides the advantage, ANN also has the ability to learn from random and noisy data, which the traditional statistical techniques cannot handle very well; the ability could make them solve problems that are often resolved by extremely complex mythologies previously.

While the BP neural network [24] is an ongoing and popular academic area of ANN, which has attracted much attention of the researchers all over the world [1, 25], ANN

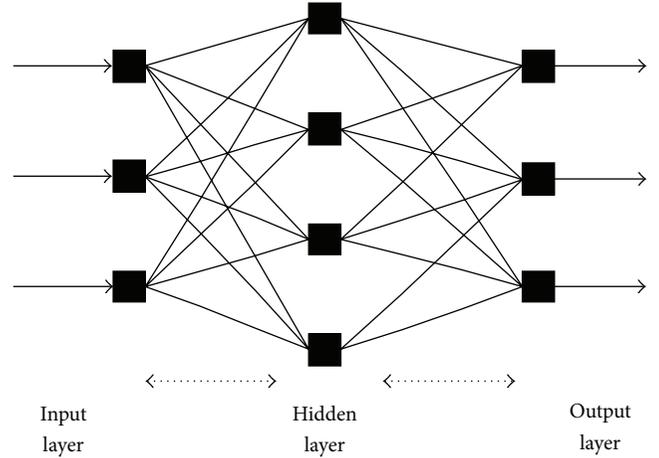


FIGURE 1: The structure of 3-layer BP neural network.

is a hierarchical feed-forward network, which also consisted of several different layers. During the training BPN would be offered by a series of input or output pattern sets over and over again. Then the network would gradually “learn” the input and output relationship by correcting the weights to minimize the error between the actual and predicted output patterns of the training set. And the trained network would be inspected through a separate set of data, which is the test set that is used to monitor performance and validity of the model. If there is a minimum in the mean squared error (MSE), network training is completed. As shown in Figure 1, most of the BP neural networks consisted of three layers: input layer, hidden layer, and output layer.

The learning process of BP neural network for predicting tunnel surrounding rock displacement could be described as follows.

(a) Initialize the BP neural network by generating a random number within $[-1, 1]$.

(b) Input a set of learning samples into the input layer to gain the output set of each unit of hidden layer by the following functions:

$$A_j = \sum_{i=1}^m w_{ij} X_i - \theta_j, \quad (1)$$

$$B_j = f(A_j), \quad j = 1, 2, \dots, n,$$

where m is the number of the units belonging to the input layer, n is the number of the units belonging to the hidden layer, i could be any unit of the input layer, j could be any unit of the hidden layer, w_{ij} is the network link weight from any unit of the input layer to that of the hidden one, θ_j is the threshold of the j unit of the hidden layer, and B_j is the output value of the j unit of the hidden layer.

(c) Calculate the output values of the units of the output layer.

The output values could be calculated by the following functions:

$$C_k = \sum_{j=1}^n V_{jk} B_j - a_k, \quad (2)$$

$$D_k = f(C_k),$$

where l is the number of the units belonging to the output layer; k could be any unit of the output layer; V_{jk} is the network link weight from any unit of the hidden layer to that of the output one; a_k is the threshold of the k unit of the hidden layer; and D_k is the output value of the k unit of the output layer.

(d) Calculate the mean squared error (MSE) of each unit of the output layer.

MSE (E_k) could be calculated by the following functions:

$$E_k = D_k - Y_k, \quad (3)$$

where Y_k is the output of the training samples.

(e) Calculate the MSE of each unit of the hidden layer (F_j) as the following functions:

$$F_j = \frac{df(A_j)}{dx} \sum_{k=1}^l V_{jk} E_k. \quad (4)$$

(f) Based on the MSE of each unit of the output layer E_k to adjust V_{jk} , that is, the network link weight from any unit of the hidden layer to that of the output one, and a_k , that is, the threshold of the k unit of the hidden layer, the adjusting functions are shown as follows:

$$V'_{jk} = V_{jk} + \beta E_k D_k, \quad (5)$$

$$a'_k = a_k + \beta E_k,$$

where β is the learning speed.

(g) Adjust w_{ij} , that is, the network link weight from any unit of the input layer to that of the hidden one; and adjust θ_j , that is, the threshold of the j unit of the hidden layer:

$$w'_{ij} = w_{ij} + \beta F_j B_j, \quad (6)$$

$$\theta'_j = \theta_j + \beta F_j;$$

when E_k is less than the maximum permissible errors, the learning process is complete; if not, go back to Step (a).

3. Support Vector Machines (SVM) for Tunnel Surrounding Rock Displacement

SVM is a type of learning technique that is based on the structural risk minimization principle and whose objective is to minimize an upper bound of generalization error. Based on the text result from several high dimensional linear functions, SVM is shown to have very high generalization ability. Relying on the ability SVM could easily attain relatively more reliable data patterns than the traditional predicting methods.

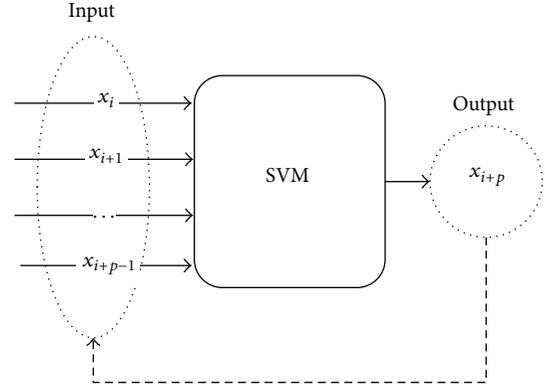


FIGURE 2: The framework of SVM.

The basic of support vector machine is shown in Figure 2 [26–28].

The time sequence of tunnel surrounding rock displacement is gained by monitoring. Given time sequence data set $\{x_i\} = \{x_1, x_2, \dots, x_N\}$ $i = 1, 2, \dots, N$, in order to predict the time sequence the relationship between $\{x_i, x_{i+1}, \dots, x_{i+p-1}\}$ and the displacement value x_{i+p} of the moment $i+p$ should be analyzed. The relationship is shown by the following function:

$$x_{i+p} = f(x_i, x_{i+1}, \dots, x_{i+p-1}). \quad (7)$$

According to the theory of SVM, the nonlinear relationship could be gained by learning the displacement sequence: $x_i, x_{i+1}, \dots, x_{i+p-1}$, $i = 1, 2, \dots, n-p$. SVM estimates the function by the following function:

$$f(x_{n+m}) = \sum_{i=1}^{n-p} (\alpha_i - \alpha_i^*) K(X_{n+m}, X_i) + b, \quad (8)$$

where $f(x_{n+m})$ is the displacement value of the time $n+m$, $X_{n+m} = (x_{n+m-p}, x_{n+m-p+1}, \dots, x_{n+m-1})$, $X_i = (x_i, x_{i+1}, \dots, x_{i+p-1})$, and the value of α , α^* , and b could be gained by the following function:

$$\text{Max } w(\alpha_i, \alpha_i^*) = -\frac{1}{2} \sum_{i,j=1}^{n-p} (\alpha_i - \alpha_i^*) (\alpha_j - \alpha_j^*) K(X_i X_j) \\ + \sum_{i=1}^k x_{i+p} (\alpha_i - \alpha_i^*) - \varepsilon \sum_{i=1}^{n-p} (\alpha_i + \alpha_i^*) \\ \text{s.t. } \begin{cases} \sum_{i=1}^{n-p} (\alpha_i - \alpha_i^*) = 0 \\ 0 \leq \alpha_i, \alpha_i^* \leq C, (i = 1, 2, \dots, n-p). \end{cases} \quad (9)$$

An overall description of the steps of the algorithm based on SVM is shown as follows.

(a) Initialize the BP neural network by generating a random number within $[-1, 1]$.

(b) Resort the data in an ascending sort. And inspect if there are any infeasible individuals in the set; if so delete the infeasible ones and then move to next step.

(c) In this set, $n-p$ sets should be gained from n sets. Select n sets from the general pool and choose n of them randomly.

(d) Use competitive complex evolution algorithm, which could be called CCE algorithm, to expand the sets.

(e) A single population, which consisted of the sets provided by the former step (c), should be generated in this step.

(f) If all the conjunction standards are satisfied, then calculating process could be stopped; if not, please move to Step (g).

(g) If the number of the sets is less than $n-p$, the minimum set should be removed and then move back to Step (b).

4. Case Study

To further analyze and compare the predicting model for tunnel surrounding rock based on SVM and BP neural network, this paper uses the *Fangtianchong* tunnel of Wuhan-Guangzhou railway as the test bed. DK1658+905 is chosen as the typical section to attain the data of tunnel surrounding rock displacement. And in the case study the paper selects 3 sections, the first section, the second section, and the third section, as the test bed. The measurement frequency should be mentioned; the frequency of this study is decided to be once per day. The data used in this case is collected during the period from October 8 to 14, 2013. The reason why we choose this period to detect the displacement of tunnel surrounding rock is that the rock of *Fantianchong* is a kind of soft rock, which would be affected by the environment temperature and humidity. And from October 8 to 14 the temperature and humidity around the tunnel are stable and appropriate for detecting based on researching and constructing experiences.

In order to test the two models proposed in the paper, the total data gained by observing is divided into 3 groups, which are the testing samples, training samples, and inspection samples. In the models the 3 groups are used as three sets. The data of training and testing sets are provided by the first and third sections and the data of the inspection set is from the second section of the test bed. In this case, we use root-mean-square error (RMSE) to evaluate the predicting models mentioned in this paper. And RMSE of each day's predicting results could be gained by the following functions:

$$\text{RMSE} = \sqrt{\left\{ \frac{\sum_{i=1}^n (x_i - \hat{x}_i)^2}{n} \right\}}, \quad (10)$$

where the predicted value of tunnel surrounding rock displacement is x_i , \hat{x}_i is the actual value of variable tunnel surrounding rock displacement, and n is the number of verifying points.

This paper compares the performance of the two models on the same test bed. The results are shown in Figure 3.

The predicting errors of SVM and ANN are shown in Figure 3; it is obvious that the errors of SVM models are much smaller than that of the model based on ANN. The comparative results show that solutions attained by SVM are more robust with a smaller standard error compared to ANN.

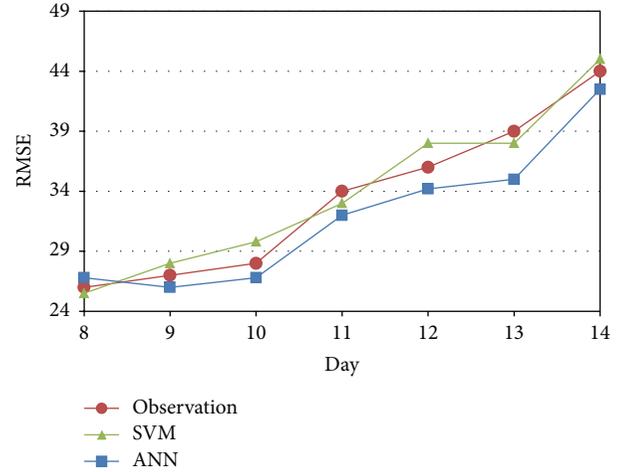


FIGURE 3: Comparison between the prediction results among observations, SVM, and ANN.

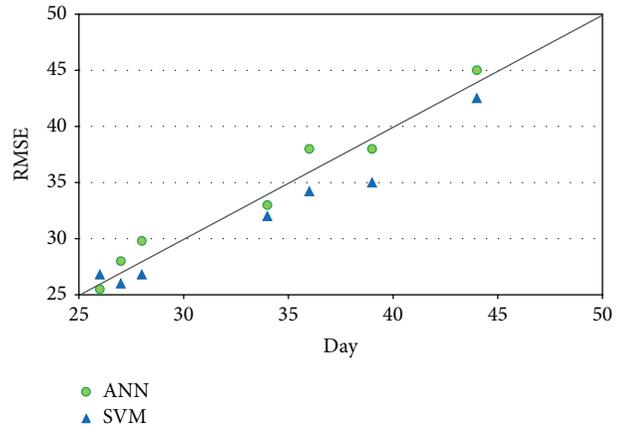


FIGURE 4: Comparison between the predicting errors of SVM and ANN.

As shown in Figure 4, with time the variations between the predicting results of ANN and the actual displacement value are small. However, the variations between the predicting results of SVM and the actual displacement value keep to be random and not in order. So it could be concluded that the structural risk minimization principle to minimize the generalization error is used by SVM, and the empirical risk minimization principle to minimize the training error is used by BP neural network. Besides, the method based on SVM tends to search for a global solution while that of BP neural network tends to fall in a certain local optimal solution. However, the predicting speed of ANN is much quicker than that of SVM. Based on our tests, the running time of the method based on SVM is about twice as much as the running time of ANN. The difference is mainly caused by the framework of SVM, which is a little more complex than the structure of ANN. Thus in the real world it is more acceptable to adapt to SVM techniques to predict the future displacements if you prefer a more accurate prediction, but

if it is emergency it could be better to rely on ANN, whose speed is quicker and errors are acceptable.

5. Conclusions

Considering the potential danger that could be caused by tunnel surrounding rock displacement, it is important and necessary to have a relatively accurate and in time prediction of the displacement. Therefore, in this paper, two methodologies, support vector machines (SVM) and artificial neural network (ANN), are introduced to predict tunnel surrounding rock displacement. To compare the two models the data of *Fangtianchong* tunnel are applied to test the two, respectively. In general, the comparison between artificial neural network (ANN) and SVM shows that SVM has higher accuracy prediction than ANN. But the predicting speed of ANN is much quicker than that of SVM. Thus, SVM can be a potential and accurate tool for tunnel surrounding rock displacement prediction, but if it is emergency it could be better to rely on ANN, whose speed is quicker and errors are acceptable.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] C. L. Zhang, X. W. Wang, and H. Chen, "Study on forecast and prediction methods for tunnel wall rock deformation," *Technology of Highway and Transport*, vol. 4, pp. 88–92, 2008.
- [2] W. Xiao and W. Leng, "Dynamic displacement prediction for the retaining structure of the deep excavation pit in subway," *China Railway Science*, vol. 25, no. 5, pp. 84–88, 2004.
- [3] Q. N. Chen, Y. X. Zhang, and J. G. Chen, "The displacement prediction of surrounding rock based on BP neural network for the tunnel," *Journal of Highway and Transportation Research and Development*, vol. 21, no. 2, pp. 65–69, 2004.
- [4] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research C: Emerging Technologies*, vol. 43, part 3, pp. 233–248, 2014.
- [5] B. Z. Yao, P. Hu, M. H. Zhang, and X. M. Tian, "Improved ant colony optimization for seafood product delivery routing problem," *PROMET-TRAFFIC & Transportation*, vol. 26, no. 1, pp. 1–10, 2014.
- [6] B. Yao, P. Hu, M. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem," *Simulation*, vol. 89, no. 6, pp. 762–770, 2013.
- [7] B. Z. Yao, J. B. Yao, M. H. Zhang, and L. Yu, "Improved support vector machine regression in multi-step-ahead prediction for rock displacement surrounding a tunnel," *Scientia Iranica*. In Press, <http://www.scientiairanica.com/en/ArticlesInPress>.
- [8] B. Z. Yao, C. Y. Yang, J. B. Yao, and J. Sun, "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.
- [9] B. Z. Yao, C. Y. Yang, J. B. Yao, J. J. Hu, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6-7, pp. 2127–2131, 2011.
- [10] B. Yu, Z. Yang, and K. Chen, "Hybrid model for prediction of bus arrival times at next station," *Journal of Advanced Transportation*, vol. 44, no. 3, pp. 193–204, 2010.
- [11] B. Yu, Z. Z. Yang, and B. Z. Yao, "A hybrid algorithm for vehicle routing problem with time windows," *Expert Systems with Applications*, vol. 38, no. 1, pp. 435–441, 2011.
- [12] B. Yu, Z. Yang, and J. Yao, "Genetic algorithm for bus frequency optimization," *Journal of Transportation Engineering*, vol. 136, no. 6, pp. 576–583, 2010.
- [13] B. Yu, H. B. Zhu, W. J. Cai, N. Ma, and B. Z. Yao, "Two-phase optimization approach to transit hub location—the case of Dalian," *Journal of Transport Geography*, vol. 33, pp. 62–71, 2013.
- [14] X. H. Li, Y. Zhao, X. G. Jin, Y. Y. Lu, and X. F. Wang, "Application of grey majorized model in tunnel surrounding rock displacement forecasting," in *Advances in Natural Computation*, vol. 3611 of *Lecture Notes in Computer Science*, pp. 584–591, 2005.
- [15] Y. S. Li, X. P. Li, and C. L. Zhang, "The displacement prediction of surrounding rock based on BP neural network for the tunnel," *Journal of Highway and Transportation Research and Development*, vol. 25, supplement 1, pp. 2970–2973, 2006.
- [16] P. J. Sellner and W. Schubert, "Prediction of displacements in tunnelling," in *Proceedings of the ISRM International Symposium*, International Society for Rock Mechanics, 2000.
- [17] A. G. Parlos, O. T. Rais, and A. F. Atiya, "Multi-step-ahead prediction using dynamic recurrent neural networks," *Neural Networks*, vol. 13, no. 7, pp. 765–786, 2000.
- [18] C. H. Cheng, J. Lehmann, and M. H. Engelhard, "Natural oxidation of black carbon in soils: Changes in molecular form and surface charge along a climosequence," *Geochimica et Cosmochimica Acta*, vol. 72, no. 6, pp. 1598–1610, 2008.
- [19] H. S. Lee, H. J. Billings, and M. N. Lehman, "The Suprachiasmatic Nucleus: A Clock of Multiple Components," *Journal of Biological Rhythms*, vol. 18, no. 6, pp. 435–449, 2003.
- [20] G. Chevillon and D. F. Hendry, "Non-parametric direct multi-step estimation for forecasting economic processes," *International Journal of Forecasting*, vol. 21, no. 2, pp. 201–218, 2005.
- [21] J. Liu, W. Wang, and F. Golnaraghi, "A multi-step predictor with a variable input pattern for system state forecasting," *Mechanical Systems and Signal Processing*, vol. 23, no. 5, pp. 1586–1599, 2009.
- [22] B. Yu, Z. Z. Yang, and B. Yu, "Hybrid model for prediction of bus arrival times," *Neural Network World*, vol. 19, no. 3, pp. 321–332, 2009.
- [23] B. Yegnanarayana, *Artificial Neural Networks*, PHI Learning, 2009.
- [24] W. Jin, Z. J. Li, L. S. Wei, and H. Zhen, "The improvements of BP neural network learning algorithm," in *Proceedings of the 5th International Conference on Signal Processing (WCCC-ICSP '00)*, vol. 3, pp. 1647–1649, IEEE, 2000.
- [25] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus arrival time prediction using support vector machines," *Journal of Intelligent Transportation Systems*, vol. 10, no. 4, pp. 151–158, 2006.
- [26] V. Cherkassky and Y. Ma, "Practical selection of SVM parameters and noise estimation for SVM regression," *Neural Networks*, vol. 17, no. 1, pp. 113–126, 2004.

- [27] C. W. Hsu, C. C. Chang, and C. J. Lin, "A practical guide to support vector classification," Tech. Rep., Department of Computer Science and Information Engineering, National Taiwan University, 2003.
- [28] C. W. Hsu, C. C. Chang, and C. J. Lin, *A Practical Guide to Support Vector Classification*, 2003.

Research Article

Short-Term Power Generation Energy Forecasting Model for Small Hydropower Stations Using GA-SVM

Gang Li,¹ Yongjun Sun,¹ Yong He,¹ Xiufeng Li,² and Qiyu Tu²

¹ Institute of Hydropower and Hydroinformatics, Dalian University of Technology, Dalian 116024, China

² Yunnan Power Dispatching Control Center, Kunming 650011, China

Correspondence should be addressed to Gang Li; glee@dlut.edu.cn

Received 12 May 2014; Revised 26 June 2014; Accepted 27 June 2014; Published 17 July 2014

Academic Editor: Bin Yu

Copyright © 2014 Gang Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Accurate and reliable power generation energy forecasting of small hydropower (SHP) is essential for hydropower management and scheduling. Due to nonperson supervision for a long time, there are not enough historical power generation records, so the forecasting model is difficult to be developed. In this paper, the support vector machine (SVM) is chosen as a method for short-term power generation energy prediction because it shows many unique advantages in solving small sample, nonlinear, and high dimensional pattern recognition. In order to identify appropriate parameters of the SVM prediction model, the genetic algorithm (GA) is performed. The GA-SVM prediction model is tested using the short-term observations of power generation energy in the Yunlong County and Maguan County in Yunnan province. Through the comparison of its performance with those of the ARMA model, it is demonstrated that GA-SVM model is a very potential candidate for the prediction of short-term power generation energy of SHP.

1. Introduction

Small hydropower (SHP) is a kind of world recognized and concerned renewable clean energies. It widely attracts attention in the whole world as its great significance for medium and small rivers management, strengthening the rural water conservancy infrastructure construction, meets rural energy demand, improves the rural energy structure, reduces the pollution of the environment, responds to climate change, promotes the development of the local economy [1–5], and so forth. In the past two decades, the installed capacity of SHP increases more than 2.5 GW per year because it has many advantages, such as small scale, mature technology, short construction time, less investment, and near-zero pollution emissions, and generally causes no immigration or land submersion.

Up to the end of 2012, the installed capacity of SHP in China had exceeded 65 GW and annual generation over 200 TWh, which take about 30% of hydropower installed capacity and power generation, respectively, and both rank first in the world [6]. Different from other countries in the world, SHP plays an important role in China's rural electricity

supply as it is widely distributed in more than 1600 mountainous counties in China; approximately half of the territories, one-third of counties, and a quarter of the total population are dependent upon SHP for rural electricity supply [7, 8]. However, with the fast development of SHP and large-scale access to power grid, its influence on the power grid is becoming more and more obvious, especially in southwest China which has rich SHP resource. SHP has become a major factor that affects the safe operation and development of power grid. Most of SHP plants are runoff river plant without regulation ability, so its power output is obviously intermittent and seasonal because of the uncertainty of rainfall. In particular, in flood season, the rainfall is very big and focused so that SHP plants may generate much more power output than other periods. At the same time, the big hydropower plant also generated even more power output. That can probably lead to water resource wasted and electricity dumped under the condition of current transmission capacity. Therefore, it is necessary to master short-term power generation energy (STPGE) of SHP in order to avoid the above situation through using regulation ability of big hydropower plants.

However, SHP plants are generally in the small remote river basin with shortage of hydrologic station and the management is weak due to nonperson supervision for a long time, so it is very difficult for forecasting STPGE of SHP because of lack of necessary runoff data. At present, a lot of research activities in short-term forecasting models of hydropower stations have been carried out, which focus on the forecasting of inflow in reservoirs [9–12], of stream flow [13–15], or of precipitation [16]. But there are few research works referring to forecasting the STPGE for SHP stations [17]. Since the parameters will greatly affect the performance of SVM, some literatures attempted to determine the proper parameter values for their problems [18, 19]. However, for large scale or real-time feature practice application, the considerable search time cannot be accepted. Heuristic algorithms have been successfully used in many complex problems [20–22].

This paper presents a novel short-term forecasting model (named GA-SVM) for power generation energy of SHP stations. In this study, support vector machine (SVM) was used to identify power generation energy based on structural risk minimization principle [18, 23–26] and its parameters are optimized by genetic algorithm (GA) to get the optimal model structure [27, 28]. Considering dynamically putting into operation of SHP plant or hydrounit, the installed capacity utilization hours of SHP are selected as input and output value of the proposed forecasting model since the power generation energy of SHP is not the same at different times. This method is applied to forecast STPGE of the small hydropower stations in Yunlong County and Maguan County, Yunnan province, China. Compared with the conventional method, the proposed GA-SVM model exhibits superior performance, demonstrating GA-SVM's effectiveness as an approach to forecast STPGE of SHP.

The paper is organized as follows. In the next section "Brief Introduction to SVM and GA," SVM and GA algorithms are briefly introduced. Then, the proposed GA-SVM forecasting method is described in the following section. In the next section, this method is applied to Yunnan province, and the results are compared with those of conventional method. The final section concludes the paper.

2. Brief Introduction to SVM and GA

2.1. Support Vector Machines (SVM). The SVM, developed by Vapnik [29], is based on statistical learning theory and implements the structural risk minimization principle rather than the empirical risk minimization principle implemented by most traditional ANN models. It seeks to minimize an upper bound to the generalization error instead of minimizing the training error and can achieve an optimum network structure. Many researchers have used SVM to implement forecasting model in every field, which mainly focuses on forecasting rainfall. Dibike et al. demonstrated the capability of the SVM in hydrological prediction, such as modeling the rainfall runoff process [30]. There are other scholars who have used the SVM for rainfall forecast ranging from 1-2 days ahead to 1h ahead [31]. In this paper, the SVM

model is used to forecast STPGE of SHP. And the radial basis function (RBF) is employed as kernel function which has shown to simplify the use of a mapping, because the RBF is more compact in comparison with other kernels and is able to shorten the computational training process and improve the generalization performance [30]. The RBF is also computationally simpler than a polynomial kernel, which has more parameters [32]. The equation for RBF is of the form

$$k(x_i, x) = \exp \left\{ \frac{-\|x - x_i\|^2}{2\sigma^2} \right\}. \quad (1)$$

2.2. Genetic Algorithm (GA). GA is a global optimal algorithm based on "survival of the fittest" in Darwin's theory of evolution and provides an efficient and robust optimized searching method in complex space. This is an excellent search algorithm adapted to the global probability. GA operates iteratively on a population of structures, each of which represents a candidate solution to the problem, encoded as a string of symbols (chromosome), and uses randomized technical guidance to effectively search a coded parameter space. GA makes use of coding technology to transform the solved space of problem into chromosome space and also convert the decisive variable into a certain structure of individual chromosomes. During the iteration of the algorithm, according to the rules set by the fitness function, these groups made up of individuals generated next generation through selection, crossover, and mutation. Fitness factor which is beneficial to the population will be inherited, while factors that reduce fitness will be eliminated with the operation of mutation and crossover in iterations. After continuous evolutions, the optimal individuals survive, which can be approximate optimal solution of the problem.

3. Short-Term Forecasting Model for Power Generation Energy Using GA-SVM

3.1. Forecasting Object. Generally, the daily power generation energy is directly selected as forecasting object for STPGE of SHP. But, considering dynamically putting into operation small hydropower plant or hydrounit in some region, there is a difference of installed capacity of SHP between one day and another day. Since the power output of SHP plant is almost close to installed capacity in flood season, the power generation energy is also very different due to the increase in installed capacity of SHP. The model prediction performance will be affected if power generation energy of SHP is only used as input and output values of the model. Therefore, the installed capacity utilization hour represents power generation energy of SHP in region. That could not only accurately reflect the characteristics of small hydropower plant without regulation ability but also alleviate short-term fluctuations in power generation curve. The installed capacity utilization hour was

$$\text{Hour}_d = \frac{\text{Energy}_d}{\text{Capacity}_d}, \quad (2)$$

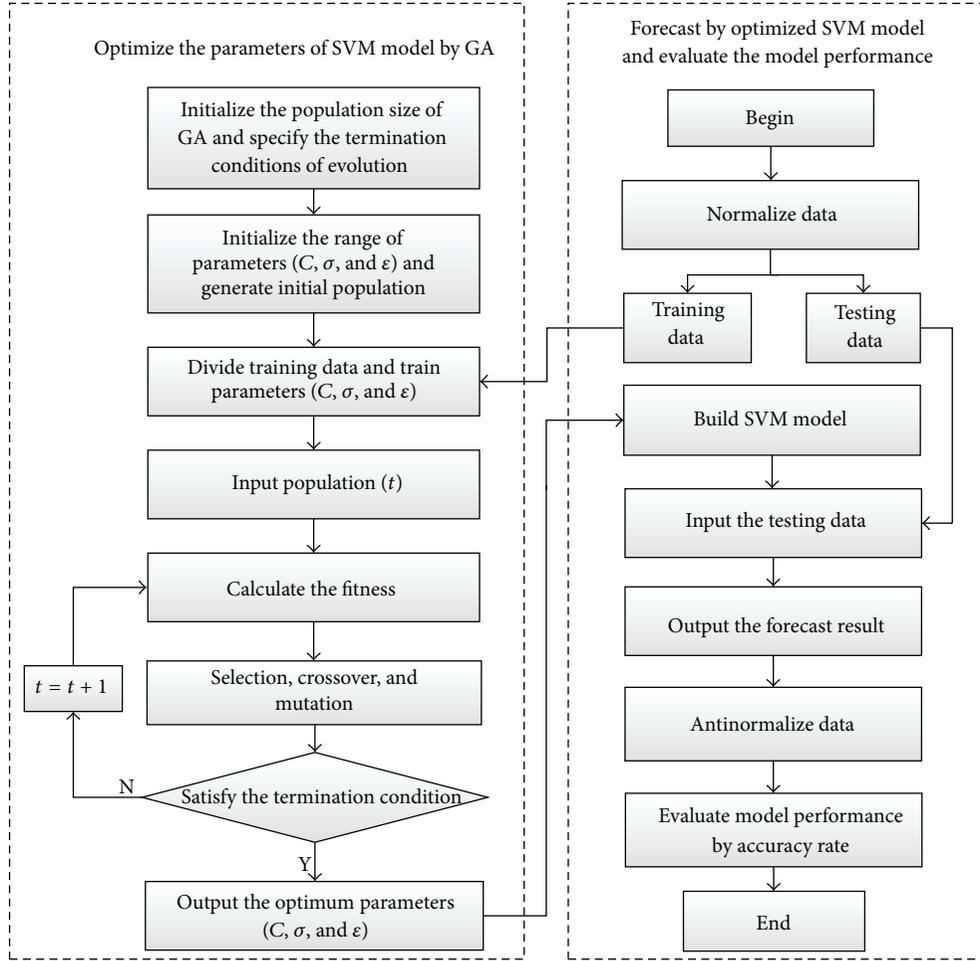


FIGURE 1: The flow chart of optimizing SVM by GA.

where Hour_d is installed capacity utilization hour in region at day d ; Energy_d is power generation energy in region at day d ; Capacity_d is the install capacity of all small hydropower plants in region at day d .

3.2. Short-Term Forecasting Model of SHP Using GA-SVM. To apply SVM model to forecast STPGE of SHP plants in region, we need to know the three vital parameters RBF kernels: C , ϵ , and σ , which respectively denote positive constant, insensitive loss function, and Gaussian noise level of standard deviation. Different values of C , ϵ , and σ can lead to large differences in the forecasting result. The parameters C , ϵ , and σ control the complicity of the model and error of the approximation, thus reflecting the difficulty of the training and the forecasting accuracy. In order to improve the forecasting accuracy, we should confirm the three parameters. In recent years, several methods such as the genetic algorithm [33, 34] and shuffled complex evolution algorithm [35–37] have been developed for model parameter calibration. In this paper, GA is used to optimize parameters of SVM kernel function. This approach requires no a priori knowledge and is of high stability and accuracy. Figure 1 illustrates the flow chart of optimizing the three parameters of SVM model by

GA. The GA is used to seek a better combination of the three parameters in the SVM so that a bigger forecasting accuracy is obtained in each iteration.

In this study, the input and output variables are normalized in the range from 0 to 1 by (3). That can minimize deformation error range and guarantee the unity of the model data in order to improve prediction accuracy. Consider

$$\tilde{\text{Hour}}_d = \frac{\text{Hour}_d - \text{Hour}_{\min}}{\text{Hour}_{\max} - \text{Hour}_{\min}}, \quad (3)$$

where $\tilde{\text{Hour}}_d$ is the normalization value at day d ; Hour_d is the original value at day d ; Hour_{\max} and Hour_{\min} are the maximum and minimum of sample data sets, respectively.

After training and testing the GA-SVM model, the forecast value of power generation energy is calculated by

$$\begin{aligned} \text{Hour}_d &= \tilde{\text{Hour}}_d \times (\text{Hour}_{\max} - \text{Hour}_{\min}) + \text{Hour}_{\min}, \\ \text{Energy}_d &= \text{Hour}_d \times \text{Capacity}_d. \end{aligned} \quad (4)$$

3.3. Model Performance Estimation. A lot of goodness-of-fit measurements have been applied to evaluate model performance. Appropriate evaluation criteria should be chosen

TABLE 1: The X_{mean} , S_x , C_s , X_{min} , and X_{max} of the data set of Yunlong County and Maguan County.

County	Data set	X_{mean}	S_x	C_s	X_{min}	X_{max}
Yunlong	Training	800.8	392.6	0.6	200.2	2366.2
	Testing	1174.1	154.8	-1.2	700.6	1369.6
	Entire	825.7	392.4	0.5	200.2	2366.2
Maguan	Training	2440.4	1121.2	0.4	440.6	4805.1
	Testing	3402.1	709.9	0.4	2336.5	4766.3
	Entire	2504.6	1124.3	0.3	440.6	4805.1

when using multicriteria to validate model performance [38]. In this paper, the following two statistical measures, which are usually used in other researches, are chosen as evaluation criteria for model performance:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{d=1}^n (\text{Energy}_d - \text{Energy}_d^*)^2}, \quad (5)$$

$$\text{MAPE} = \frac{1}{n} \sum_{d=1}^n \left| \frac{\text{Energy}_d^* - \text{Energy}_d}{\text{Energy}_d^*} \right| \times 100,$$

where n is the total amount of observed data, Energy_d^* and Energy_d are respective observed and forecasted value at day d .

The root mean squared error (RMSE) is an arbitrary positive value and will indicate a good performance when it is close to zero. The mean absolute percentage error (MAPE) is a relative index of absolute model error and can express accuracy as a percentage [39, 40]. The smaller the value of MAPE is, the better performance the model shows.

4. Numerical Results

4.1. Study Areas and Data. There is extremely rich hydropower resource in Yunnan province, whose potential capacity ranks third in China. The hydropower resources of every region are extremely uneven and mainly distributed in the west and north, followed by the east and south. By the end of October 2012, the SHP plants in Yunnan had reached 1587, with 3417 units and 8453.05 MW of the installed capacity, which accounts for more than 27% and 12% of hydropower capacity in Yunnan province and SHP capacity in China, respectively [41]. The two typical counties, Yunlong County and Maguan County, are in Dali region and Wenshan region in Yunnan province, respectively, and are selected as study areas in this paper. The location of the two counties is shown in Figure 2.

Yunlong County is located in the west of Yunnan province with a total area of 4400.95 km². And the annual average temperature and annual average rainfall are 15.9°C and 729.5 mm, respectively. By the end of 2013, there are 10 small hydropower plants with installed capacity 111.5 MW. Maguan County is located in the southeast of Yunnan province with a total area of 2676 km². And the annual average temperature and annual average rainfall are 16.9°C and 1345 mm, respectively.

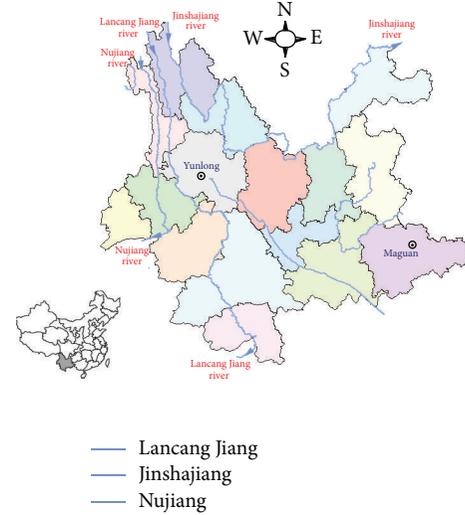


FIGURE 2: Location of the study area.

By the end of 2013, there are 22 small hydropower plants with installed capacity of 213.89 MW.

The data derived from the two counties are both 915 days long with the period between May 1, 2011, and October 31, 2013, for which 854 days of the power generation energy data from May 1, 2011, to August 31, 2013, are used for calibration and the remaining 61 days from September 1, 2013, to October 31, 2013, are used for validation. The daily statistical parameters of calibration and validation and the entire data set for the two counties are shown in Table 1. In the table, X_{mean} , S_x , C_s , X_{min} , and X_{max} stand for mean, standard deviation, skewness coefficient, minimum, and maximum, respectively. The table indicates that the training data fully includes validation data. In addition, it can be easily found that power generation energy for the two counties both vary over a wide range and are concentrated in the flood season, much bigger than other seasons. So the data from September to October in flood season is selected for model testing and other data for model training. In addition, the dispatching personnel of power grid are more concerned about power generation energy of SHP in flood season.

4.2. Results and Discussion. In this study, the GA is employed as parameter search scheme. In order to get better parameters of SVM, the maximum iterative time of GA is set as 50 and the population size is set to 30, 50, 80, 100, 120, and 150,

TABLE 2: The performance statistics of GA-SVM models for Yunlong County.

Population size	Optimal parameters (C , ϵ , and σ)	Calibration		Validation	
		RMSE	MAPE	RMSE	MAPE
(i) 30	(4.0754, 0.1989, 0.0078)	113.88	8.52	81.08	5.15
(ii) 50	(5.5762, 0.2275, 0.0073)	113.66	8.49	77.31	5.02
(iii) 80	(18.7495, 0.105, 0.0103)	113.98	8.53	80.42	5.08
(iv) 100	(10.3452, 0.1464, 0.0077)	113.87	8.51	80.86	5.11
(v) 120	(24.4174, 0.0528, 0.0064)	114.46	8.58	81.85	5.19
(vi) 150	(9.4141, 0.0348, 0.0025)	115.11	8.62	80.53	5.07

TABLE 3: The performance statistics of GA-SVM models for Maguan County.

Population size	Optimal parameters (C , ϵ , and σ)	Calibration		Validation	
		RMSE	MAPE	RMSE	MAPE
(i) 30	(2.3792, 0.6749, 0.0058)	252.25	7.54	233.67	4.37
(ii) 50	(7.3517, 0.056, 0.0197)	254.95	7.92	234.36	4.38
(iii) 80	(10.808, 0.0799, 0.0192)	254.29	7.90	233.75	4.40
(iv) 100	(8.4248, 0.0538, 0.0156)	255.18	7.92	233.54	4.36
(v) 120	(14.0828, 0.0758, 0.0191)	254.28	7.89	233.92	4.40
(vi) 150	(11.421, 0.058, 0.0063)	255.87	7.83	233.02	4.32

TABLE 4: AIC value and performance indices of alternative ARMA models for Yunlong County.

(p, q)	AIC	Calibration		Validation	
		RMSE	MAPE	RMSE	MAPE
(3, 12)	9.5266	114.30	8.70	82.07	5.58
(4, 8)	9.5270	114.84	8.87	83.40	5.24
(5, 13)	9.5153	113.17	8.98	78.49	5.06
(6, 12)	9.5281	113.98	8.91	78.54	5.04
(7, 12)	9.5276	113.80	8.90	77.88	4.99
(8, 8)	9.5265	113.66	8.85	80.18	5.22

respectively. And the optimal scope of three parameters (C , ϵ , and σ) of SVM model are $[2^{-5}, 2^5]$, $[0, 2]$, and $[2^{-13}, 2^{-1}]$, respectively. The performance statistics of SVM models are given in Tables 2 and 3 for the two counties.

The results from Table 2 clearly indicate that the population size (ii) for SVM models with the optimal parameters (C, ϵ, σ) = (5.5762, 0.2275, 0.0073) can be selected as forecast model for Yunlong County.

For Maguan County, it can be seen from Table 3 that the two statistical measures of population size (i) in calibration stage are clearly better than others since those are slightly better or worse in validation stage. So the optimal parameters (C, ϵ, σ) = (2.3792, 0.6749, 0.0058) were selected through comprehensive comparison.

In order to get a better comprehension of the GA-SVM model performance, the ARMA model was employed as a comparative purpose. The basic components to an ARMA model is autoregression (AR) and moving-average (MA). To obtain a suitable ARMA (p, q) model, the two integers p

and q have to be determined, respectively, by the number of autoregressive orders and the number of moving-average orders of the ARMA model. In this paper, the AIC (Akaike information criterion) value of ARMA models, for p and q ranging from 1 to 13, is calculated.

For Yunlong County, the models ARMA (3, 12), (4, 8), (5, 13), (7, 12), (6, 12), and (8, 8), which have relatively smaller AIC values, are selected as the candidate models. Table 4 shows the AIC value and the performance of selected ARMA models. By comparing analysis, the ARMA (7, 12) model was chosen as the final ARMA model for Yunlong County.

For Maguan County, the models ARMA (1, 2), (2, 1), (2, 2), (2, 3), (2, 4), and (3, 1), which have relatively smaller AIC values, are selected as the candidate models. Table 5 shows the AIC value and the performance of selected ARMA models. By comparing analysis, the ARMA (2, 4) model was chosen as the final ARMA model for Maguan County.

In this study, the same training and verification sets are used for the two models in order to have the same basis

TABLE 5: AIC value and performance indices of alternative ARMA models for Maguan County.

(p, q)	AIC	Calibration		Validation	
		RMSE	MAPE	RMSE	MAPE
(1, 2)	11.1782	261.22	7.94	234.76	4.46
(2, 1)	11.1739	260.49	7.95	233.48	4.53
(2, 2)	11.1755	260.37	7.96	234.82	4.55
(2, 3)	11.1778	260.36	7.96	234.13	4.54
(2, 4)	11.1779	260.14	7.95	232.24	4.47
(3, 3)	11.1767	260.07	7.96	234.95	4.60

TABLE 6: Model statistics of the calibration and validation period for Yunlong County.

Model	Calibration		Validation	
	RMSE	MAPE	RMSE	MAPE
GA-SVM	113.66	8.49	77.31	5.02
ARMA	113.80	8.90	77.88	4.99

TABLE 7: Model statistics of the calibration and validation period for Maguan County.

Model	Calibration		Validation	
	RMSE	MAPE	RMSE	MAPE
GA-SVM	252.25	7.54	233.67	4.37
ARMA	260.14	7.95	232.24	4.47

of comparison. Meanwhile, in order to evaluate the model performance for forecasting STPGE of SHP, the time series data are derived from two study sites in different region. And the two statistical measures are employed to evaluate the model performance.

For Yunlong County, the model's RMSE and MAPE statistics of the calibration and validation period are summarized in Table 6. With the results shown in Table 6, the analysis can be executed crisply. The results reveal that the GA-SVM model outperformed ARMA with respect to the two measures in the calibration period. In this stage, the GA-SVM model improved the ARMA model of about 0.24 in RMSE value and 0.41 in MAPE value. For the comparison between GA-SVM and ARMA model in the validation period, the GA-SVM obtains better RMSE value than the ARMA; while the MAPE value of the two models are nearly equal to each other. Figure 3 shows the comparison of forecasted versus observed discharge using GA-SVM and ARMA model for Yunlong County. It can be seen from the residuals that the GA-SVM model performs better than ARMA. Furthermore, it can be concluded from Table 4 and Figure 3 that GA-SVM model obtains slightly better forecast precision than ARMA.

For Maguan County, the model's RMSE and MAPE statistics of the calibration and validation period are summarized in Table 7. Table 7 demonstrates that the GA-SVM model is clearly superior to ARMA in the calibration and validation period of the two measures. In the validation period, the GA-SVM model improved the ARMA model of about 7.89 and 0.41 in RMSE and MAPE values, respectively. For the comparison between GA-SVM and SVM model in the validation period, the GA-SVM model obtains slightly better MAPE value and worse RMSE value than the ARMA. Figure 4 shows the comparison of forecasted versus observed power generation energy using GA-SVM and ARMA models for the Maguan County. As can be seen from the residuals,

the GA-SVM model performs better than ARMA except for a few peaks. Furthermore, it can be concluded from Table 5 and Figure 4 that the GA-SVM model overall performs better than the ARMA model.

5. Conclusion

In the present study, the GA-SVM prediction model comprising support vector machine with genetic algorithm has been developed for forecasting short-term power generation energy of small hydropower in region. The historical observed data derived from Yunlong County and Maguan County in Yunnan province in China were employed to investigate the modeling potentiality of GA-SVM. Data from May 1, 2011, to August 31, 2013, and from September 1, 2013, to October 31, 2013, are used for training and validation, respectively, in short-term power generation energy prediction. Due to the lack of small hydropower operation data, SVM is chosen as forecasting model because of its ability in solving small sample. The three parameters of SVM model are not known a priori and optimized by GA in order to get appropriate parameters for improving forecasting accuracy. In order to get a better comprehension of the GA-SVM model performance, the ARMA model was employed as a comparative purpose. The two models were constructed and their performances were compared crisply. The results indicated that the GA-SVM model can give slightly better prediction performance than the other model.

For the less data of small hydropower in region, the GA-SVM model proposed in this paper is an effective method for improving short-term forecasting accuracy. That is useful for fully absorbing small hydropower resources and avoiding water resource wasted and electricity dumped in flood season.

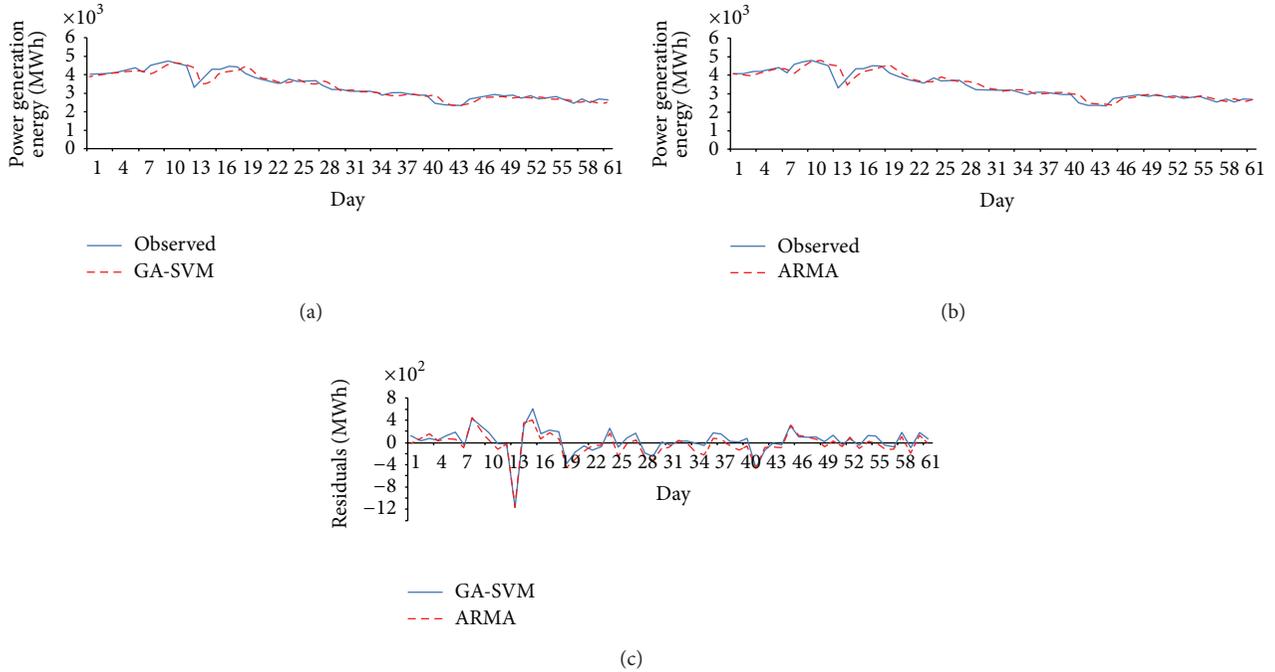


FIGURE 3: Comparison of forecasted versus observed power generation energy using GA-SVM and ARMA model for Yunlong County.

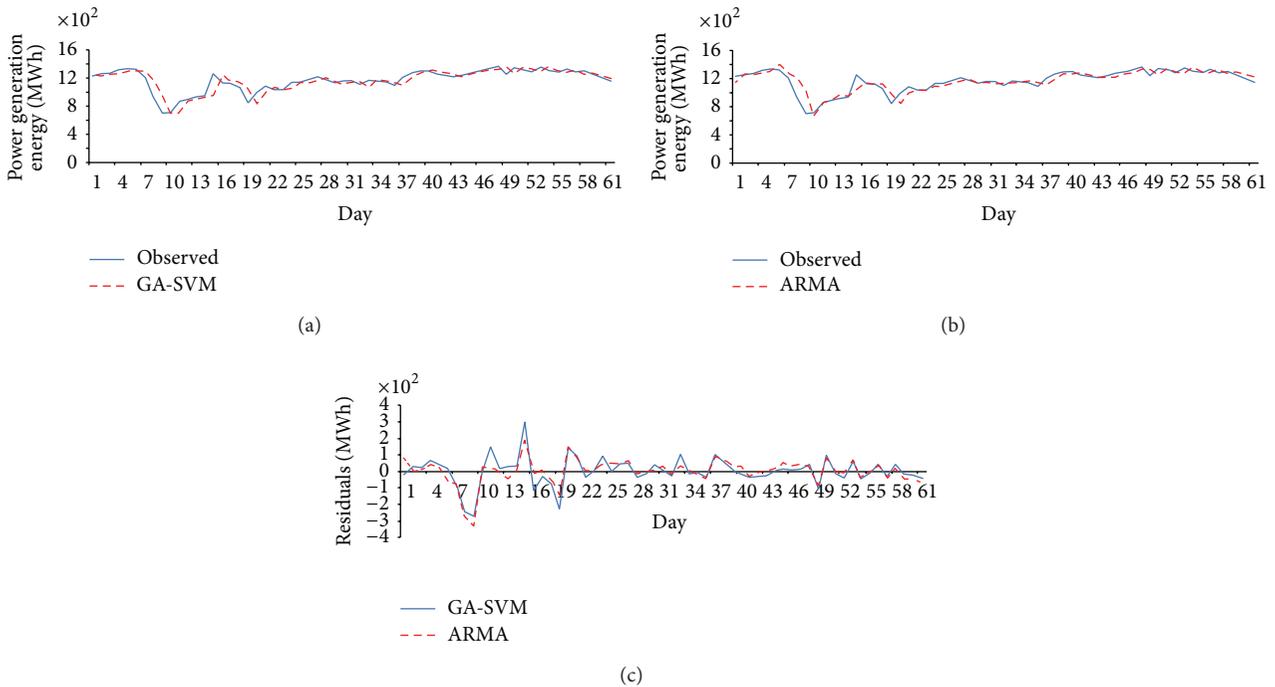


FIGURE 4: Comparison of forecasted versus observed power generation energy using GA-SVM and ARMA model for Maguan County.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the National High Technology Research and Development of China 863 Program

(2012AA050205) and the Fundamental Research Funds for the Central Universities (DUT13JN05).

References

- [1] I. Yuksel, "As a renewable energy hydropower for sustainable development in Turkey," *Renewable and Sustainable Energy Reviews*, vol. 14, no. 9, pp. 3213–3219, 2010.
- [2] S. Dudhani, A. K. Sinha, and S. S. Inamdar, "Assessment of small hydropower potential using remote sensing data for sustainable development in India," *Energy Policy*, vol. 34, no. 17, pp. 3195–3205, 2006.
- [3] T. Haakon Bakken, H. Sundt, A. Ruud et al., "Development of small versus large hydropower in Norway comparison of environmental impacts," *Energy Procedia*, vol. 20, pp. P185–P199, 2012.
- [4] T. Bockman, S. Fleten, E. Juliussen, H. J. Langhammer, and I. Revdal, "Investment timing and optimal capacity choice for small hydropower projects," *European Journal of Operational Research*, vol. 190, no. 1, pp. 255–267, 2008.
- [5] L. Kosnik, "The potential for small scale hydropower development in the US," *Energy Policy*, vol. 38, no. 10, pp. 5512–5519, 2010.
- [6] J.-K. Li, "Research on prospect and problem for hydropower development of China," *Procedia Engineering*, vol. 28, pp. P677–P682, 2012.
- [7] S. Zhou, X. Zhang, and J. Liu, "The trend of small hydropower development in China," *Renewable Energy*, vol. 34, no. 4, pp. 1078–1083, 2009.
- [8] H. Ma, L. Oxley, and J. Gibson, "China's energy situation in the new millennium," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 8, pp. 1781–1799, 2009.
- [9] V. Nikam and K. Gupta, "SVM-based model for short-term rainfall forecasts at a local scale in the Mumbai Urban area, India," *Journal of Hydrologic Engineering*, vol. 19, pp. P1048–P1052, 2014.
- [10] R. Golob, T. Štokelj, and D. Grgič, "Neural-network-based water inflow forecasting," *Control Engineering Practice*, vol. 6, no. 5, pp. P593–P600, 1998.
- [11] P. Coulibaly, F. Anctil, and B. Bobée, "Daily reservoir inflow forecasting using artificial neural networks with stopped training approach," *Journal of Hydrology*, vol. 230, no. 3–4, pp. 244–257, 2000.
- [12] D. Paravan, T. Stokelj, and R. Golob, "Improvements to the water management of a run-of-river HPP reservoir: methodology and case study," *Control Engineering Practice*, vol. 12, no. 4, pp. 377–385, 2004.
- [13] D. Dutta, W. D. Welsh, J. Vaze, S. S. H. Kim, and D. Nicholls, "A comparative evaluation of short-term streamflow forecasting using time series analysis and Rainfall-Runoff models in ewater source," *Water Resources Management*, vol. 26, no. 15, pp. 4397–4415, 2012.
- [14] M. Vafakhah, "Application of artificial neural networks and adaptive neuro-fuzzy inference system models to short-term streamflow forecasting," *Canadian Journal of Civil Engineering*, vol. 39, no. 4, pp. 402–414, 2012.
- [15] C. M. Zealand, D. H. Burn, and S. P. Simonovic, "Short term streamflow forecasting using artificial neural networks," *Journal of Hydrology*, vol. 214, no. 1–4, pp. 32–48, 1999.
- [16] R. J. Kuligowski and A. P. Barros, "Experiments in short-term precipitation forecasting using artificial neural networks," *Monthly Weather Review*, vol. 126, no. 2, pp. 470–482, 1998.
- [17] C. Monteiro, I. J. Ramirez-Rosado, and L. A. Fernandez-Jimenez, "Short-term forecasting model for electric power production of small-hydro power plants," *Renewable Energy*, vol. 50, pp. 387–394, 2013.
- [18] B. Yao, C. Yang, J. Yao, and J. Sun, "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.
- [19] B. Z. Yao, P. Hu, M. H. Zhang, and M. Q. Jin, "A support vector machine with the Tabu search algorithm for freeway incident detection," *International Journal of Applied Mathematics and Computer Science*, vol. 24, no. 2, pp. 397–404, 2014.
- [20] B. Z. Yao, P. Hu, M. H. Zhang, and X. M. Tian, "Improved ant colony optimization for seafood product delivery routing problem," *Promet—Traffic & Transportation*, vol. 26, no. 1, pp. 1–10, 2014.
- [21] B. Yao, P. Hu, M. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem," *Simulation*, vol. 89, no. 6, pp. 762–770, 2013.
- [22] B. Yao, C. Yang, J. Hu, J. Yao, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6–7, pp. 2127–2131, 2011.
- [23] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research C*, vol. 43, pp. 233–248, 2014.
- [24] B. Yu, W. H. K. Lam, and M. L. Tam, "Bus arrival time prediction at bus stop with multiple routes," *Transportation Research C*, vol. 19, no. 6, pp. 1157–1170, 2011.
- [25] B. Yu, S. Wu, B. Yao, Z. Yang, and J. Sun, "Dynamic vehicle dispatching at a transfer station in public transportation system," *Journal of Transportation Engineering*, vol. 138, no. 2, pp. 191–201, 2012.
- [26] B. Yu, Z. Z. Yang, and S. Li, "Real-time partway deadheading strategy based on transit service reliability assessment," *Transportation Research A: Policy and Practice*, vol. 46, no. 8, pp. 1265–1279, 2012.
- [27] B. Yu, Z. Yang, X. Sun, B. Yao, Q. Zeng, and E. Jeppesen, "Parallel genetic algorithm in bus route headway optimization," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5081–5091, 2011.
- [28] B. Yu, H. B. Zhu, W. J. Cai, N. Ma, and B. Z. Yao, "Two-phase optimization approach to transit hub location—the case of Dalian," *Journal of Transport Geography*, vol. 33, pp. 62–71, 2013.
- [29] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.
- [30] Y. B. Dibike, S. Velickov, D. Solomatine, and M. B. Abbott, "Model induction with support vector machines: introduction and applications," *Journal of Computing in Civil Engineering*, vol. 15, no. 3, pp. 208–216, 2001.
- [31] W. Hong, "Rainfall forecasting by technological machine learning models," *Applied Mathematics and Computation*, vol. 200, no. 1, pp. 41–57, 2008.
- [32] A. Anandhi, V. V. Srinivas, R. S. Nanjundiah, and D. Nagesh Kumar, "Downscaling precipitation to river basin in India for IPCC SRES scenarios using support vector machine," *International Journal of Climatology*, vol. 28, no. 3, pp. 401–420, 2008.
- [33] P. Pai, "System reliability forecasting by support vector machines with genetic algorithms," *Mathematical and Computer Modelling*, vol. 43, no. 3–4, pp. 262–274, 2006.
- [34] L. M. Saini, S. K. Aggarwal, and A. Kumar, "Parameter optimisation using genetic algorithm for support vector machine-based price-forecasting model in National electricity market,"

- IET Generation, Transmission and Distribution*, vol. 4, no. 1, pp. 36–49, 2010.
- [35] Q. Duan, S. Sorooshian, and V. K. Gupta, “Optimal use of the SCE-UA global optimization method for calibrating watershed models,” *Journal of Hydrology*, vol. 158, no. 3–4, pp. 265–284, 1994.
- [36] S. Sorooshian and V. Gupta, “Effective and efficient global optimization for conceptual rainfall-runoff models,” *Water Resources Research*, vol. 28, no. 4, pp. 1015–1031, 1992.
- [37] J. Lin, C. Cheng, and K. Chau, “Using support vector machines for long-term discharge prediction,” *Hydrological Sciences Journal*, vol. 51, no. 4, pp. 599–612, 2006.
- [38] R. Modarres, “Multi-criteria validation of artificial neural network rainfall-runoff modeling,” *Hydrology and Earth System Sciences*, vol. 13, no. 3, pp. 411–421, 2009.
- [39] T. S. Hu, K. C. Lam, and S. T. Ng, “River flow time series prediction with a range-dependent neural network,” *Hydrological Sciences Journal*, vol. 46, no. 5, pp. 729–745, 2001.
- [40] W. Wang, K. Chau, C. Cheng, and L. Qiu, “A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series,” *Journal of Hydrology*, vol. 374, no. 3–4, pp. 294–306, 2009.
- [41] L. Gang, L. Benxi, L. Shanzong, C. Chuntian, and L. Xiufeng, “An overview of large-scale small hydropower in yunnan power grid: situations, challenges, and measures,” in *Proceedings of the World Environmental and Water Resources Congress*, pp. 2139–2145, Cincinnati, Ohio, USA, May 2013.

Research Article

Real-Time Arterial Coordination Control Based on Dynamic Intersection Turning Fractions Estimation Using Genetic Algorithm

Pengpeng Jiao, Honglin Wang, and Tuo Sun

School of Civil and Transportation Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China

Correspondence should be addressed to Pengpeng Jiao; jiaopengpeng@bucea.edu.cn

Received 4 June 2014; Accepted 3 July 2014; Published 16 July 2014

Academic Editor: Bin Yu

Copyright © 2014 Pengpeng Jiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Real-time arterial coordination control is crucial for urban transportation systems and is partially dependent on dynamic turning flows at intersections. Few existing researches employ such information due to the restrictions of traffic surveillance systems. This paper presents a model framework for real-time arterial coordination control based on dynamic intersection turning fraction estimation, including three submodels: (1) a parameter optimization model to estimate dynamic intersection turning fractions using detected link counts at entering and exiting approaches; (2) a nonlinear model using minimum delay as an objective to optimize the time-varying public cycle for the arterial road based on the estimated turning flows; and (3) a revised optimization model to achieve real-time offset and split for the arterial road using the novel uninterrupted ratio as objective function. Two revised genetic algorithms are developed to solve the first and third submodels, respectively, and an ordinary optimization algorithm is designed for the second submodel. Time-varying public cycle, offset, and split constitute the real-time arterial coordination control scheme together. The general model framework removes most of the assumptions of conventional arterial control models and provides a time-varying timing plan. Simulation experiments using actual data indicate that the proposed model yields much better results than the existing methods.

1. Introduction

Real-time traffic signal control, especially arterial coordination control, is important for intelligent transportation systems (ITS), and the dynamic turning movement flows, that is, the dynamic origin-destination (O-D) flows at intersections, are valuable input data for signal control. However, under current traffic surveillance systems, the dynamic turning movement flows are impossible to be collected directly. Therefore, the dynamic turning fraction estimation from detected real-time link counts has been studied extensively during the past two decades.

Most researches about dynamic fractions estimation have constructed time-varying interrelations between intersection turning flows to be estimated and detected link counts at entering and exiting approaches and have proposed a series of optimization models, for example, Nihan and Davis (1987) [1], Nihan and Davis (1989) [2], Bell (1991) [3], and Jiao et al. (2005) [4]. All these models were presented as parameter

optimization formulations and were solved using traditional optimization or heuristic approaches.

To improve the estimation efficiency for on-line applications, some other works about dynamic O-D flows estimation have fallen within the scope of state-space methods and have formulated several efficient estimation models using Kalman filtering, for example, Ashok and Ben-Akiva (2002) [5], Bierlaire and Crittin (2004) [6], Lin and Chang (2007) [7] and Lou and Yin (2010) [8]. All these models focused on dynamic O-D flows estimation for freeway corridors or general road networks with rather high efficiency. Of course, they can also be transformed to estimate dynamic turning fractions at intersections; for example, Jiao et al. (2014) [9] proposed a Bayesian combined model to estimate intersection fractions, integrating Kalman filtering and back propagation neural model running simultaneously.

With respect to the arterial coordination traffic signal control, mainly two groups of models have been developed, including maximum green wave band (MGWB, or

MAXBAND) method [10, 11] and minimum delay method [12]. Most existing arterial coordination control models have been further developed based on the above two methods, including some real-world traffic signal control systems, such as Sydney Coordinated Adaptive Traffic System (SCATS) [13] and Split, Cycle and Offset Optimization Technique (SCOOT) [14]. All these models and systems have contributed a great deal to urban traffic management systems. However, very few of them employed the valuable information of dynamic O-D flows or dynamic intersection turning fractions due to the restrictions of current traffic surveillance systems. Moreover, existing arterial control models have some limitations as follows: the distance between two adjacent intersections should be approximately equal, and not longer than 800 meters; arriving vehicles must follow some given distribution; the timing plan remains unchanged during some intervals; and so forth. Therefore, the revised model which can eliminate these limitations is quite necessary.

This paper will develop a real-time arterial coordination control model with the estimated dynamic turning fractions as input data, and both the arterial control model and the turning fraction estimation model lend themselves to formulations as rather complex optimization models. Existing researches have proved that heuristic approaches or swarm intelligence algorithms are rather suitable for complicated optimization models, for example, genetic algorithm (GA) [15–17] and ant colony optimization algorithm (ACO) [18–20]. This paper will also design two revised genetic algorithms to solve the dynamic turning fraction estimation model and real-time arterial control model, respectively.

The rest of this paper is organized in the following sections. Section 2 describes the basic problem, as well as the general model architecture. Section 3 presents the model framework for real-time arterial coordination control, including dynamic intersection turning fraction estimation model, real-time public signal cycle optimization model, and real-time offset and split optimization model. Section 4 designs three algorithms to solve the above three models, respectively, including two revised genetic algorithms and a classical optimization algorithm. Section 5 illustrates the results of a case study through simulation experiments using practical data. Section 6 concludes this paper and suggests some future research directions.

2. General Model Architecture

The general model architecture of the arterial coordination control is shown in Figure 1.

Figure 1 illustrates a typical arterial corridor, the layout scheme of detectors, and the flow of the arterial coordination control. Detectors are placed at both entering and exiting approaches of each isolated intersection to collect the entrance and exit link counts, which are fundamental input data for the whole model framework. The general model consists of the following three submodels:

- (1) dynamic turning fraction estimation model: to estimate dynamic turning fractions or turning flows based on detected link counts;

- (2) nonlinear signal control model for intersection: to achieve the optimized cycle length of each intersection based on the estimated intersection turning flows, with the maximum one as the public cycle length of the arterial road;

- (3) arterial coordination control model: to optimize the offset and split of the arterial road based on the achieved public cycle length.

All three models will be illustrated in detail in the following sections.

3. Real-Time Arterial Coordination Control Model

3.1. Dynamic Intersection Turning Fraction Estimation. For each isolated intersection, the time-varying link counts at entering approach $\Omega_i(k)$ and exiting approach $\Theta_j(k)$ are detected instantaneously, where i and j denote the index of entering and exiting legs, $i = 1, \dots, r$, $j = 1, \dots, s$; k indicates the time interval. The dynamic turning flows $\eta_{ij}(k)$ shows the number of vehicles entering the intersection from leg i during interval k and leaving the intersection from leg j . Further, we define $\xi_{ij}(k)$ as the dynamic intersection turning fractions; therefore,

$$\xi_{ij}(k) = \frac{\eta_{ij}(k)}{\Omega_i(k)}. \quad (1)$$

Without considering travel time to cross the intersection, similar estimation problems have been studied in several papers. As described in our previous work [4], to accommodate the possible outliers in the detected input data, a least absolute deviation (LAD) formulation is much more robust than traditional least square (LSQ) form [3] in the objective function. Therefore, the LAD model to estimate dynamic turning fractions at intersection level is formulated as follows:

$$\begin{aligned} \min \quad & Z = \sum_k \sum_j \left| \Theta_j(k) - \sum_i \Omega_i(k) \xi_{ij}(k) \right| \\ \text{s.t.} \quad & \begin{cases} 0 \leq \xi_{ij}(k) \leq 1 \\ \sum_j \xi_{ij}(k) = 1 \\ \xi_{ij}(l) = \xi_{ij}(k), \quad l = k+1, \dots, k+K. \end{cases} \end{aligned} \quad (2)$$

In (2), the turning fractions are assumed to be constant within continuous K intervals for each step to make the model overdetermined, along with a dynamic updating mechanism to incorporate the time-varying turning fractions.

3.2. Optimization of Real-Time Public Cycle. Estimated turning movement flows provide abundant input data for the traffic signal control at single intersections. To minimize

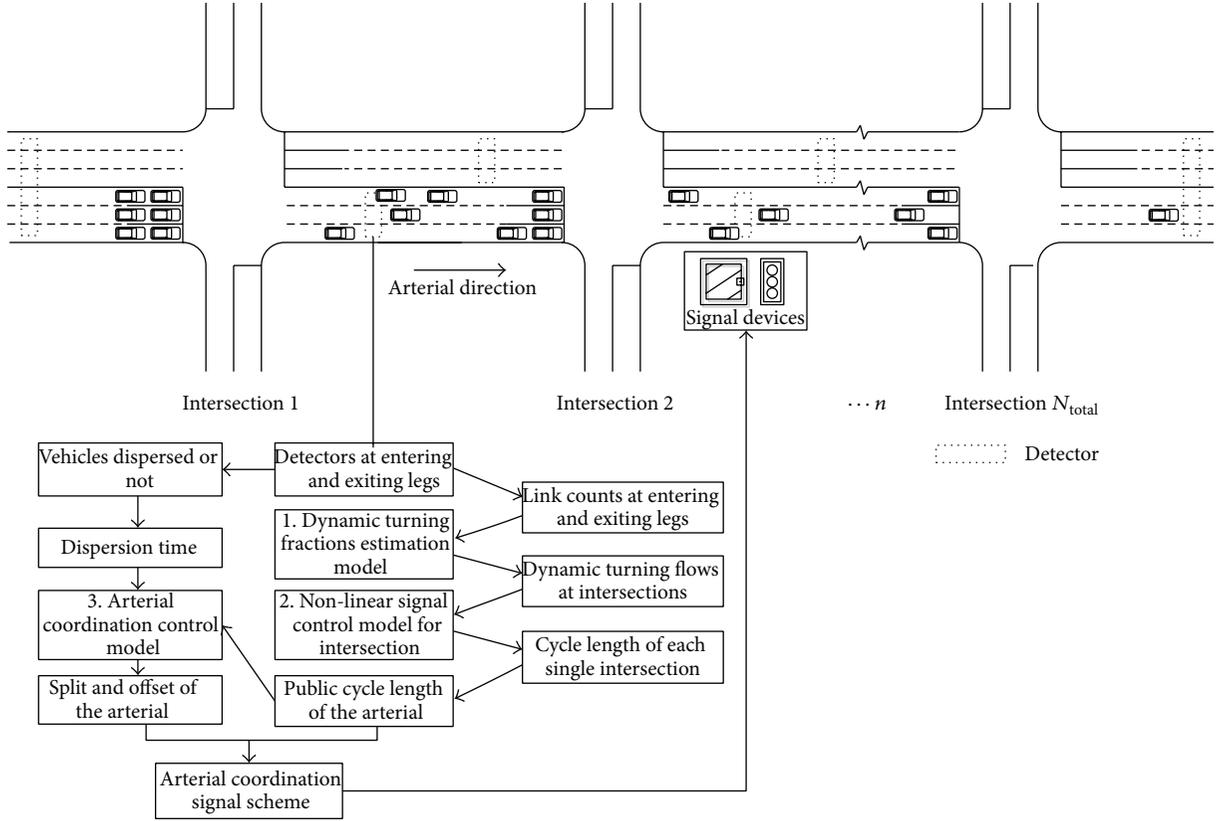


FIGURE 1: Architecture of the arterial coordination control.

the delay at intersections, a nonlinear optimization model is formulated to optimize the signal cycle length:

$$\begin{aligned}
 \min \quad & D = \sum_{x=1}^m d_x \\
 & = \sum_{x=1}^m \left[\frac{T(1 - G_x^e/T)^2}{2(1 - \text{VOC}_x)} + \frac{(1 - (\sum_{x=1}^n l_x)/T)^2}{2(\sum_{x=1}^m l_x)/T} \right] \\
 \text{s.t.} \quad & \begin{cases} \sum_{x=1}^m (G_x^e + Y_x + R_x) = T \\ \text{green}_{x,\min} \leq G_x^e \leq \text{green}_{x,\max} \\ \sum_{x=1}^m (G_x^e + l_x) \leq T_{\max} \\ G_x^e \geq 0, \quad 1 \leq x \leq m. \end{cases} \quad (3)
 \end{aligned}$$

Symbols in (3) are illustrated as follows:

D : objective function, equaling the total delay time during all phases;

d_x : average delay time during phase x ;

T : cycle length;

m : number of phases during one cycle;

G_x^e : effective green time during phase x ;

VOC_x : volume-to-saturation flow ratio during phase x , that is, the volume over capacity (VOC);

l_x : start-up lost time of phase x (3 seconds in this paper);

Y_x : yellow change interval of phase x (3 seconds in this paper);

R_x : red clearance interval of phase x (3 seconds in this paper);

$\text{green}_{x,\min}$: minimum green time during phase x ;

$\text{green}_{x,\max}$: maximum green time during phase x ;

T_{\max} : maximum cycle length (180 seconds in this paper).

Based on the estimated dynamic turning flows in Section 3.2, VOC_x can be achieved easily. Together with other detected parameters, (3) can be solved, and the optimized cycle length of each single intersection can be determined. Furthermore, since the turning flows keep changing in each interval, the optimized cycle length updates momentarily; that is, the optimized cycle length for each single intersection is real-time in nature.

The maximum cycle length is employed as the public cycle length of the arterial road for further researches.

3.3. Optimization of Real-Time Offset and Split for Arterial Coordination Control. Based on the public cycle length, we further formulate an optimization model for arterial coordination control to optimize the real-time split and offset.

In drivers' common sense, if they do not or seldom need to stop at intersections, they will feel rather comfortable, even if the speed is not very high and there exist some delays. Therefore, different from existing models, this paper defines a new index, uninterrupted ratio, to describe the effects of the model. It is equal to the percentage of vehicles which cross the intersection without interruptions. Consider

$$M_n = \frac{\text{Num}'_n}{\text{Num}_n}, \quad (4)$$

where Num'_n is the number of vehicles crossing the intersection n without interruption, Num_n is the total number of vehicles crossing the intersection n , and M_n is the uninterrupted ratio at intersection n .

According to Figure 1, to formulate the arterial coordination control model, we must know the vehicle dispersion time first. Taking n as the key intersection, there are two situations when vehicles arrive at it along the arterial direction.

Situation 1. Travel time from intersection $n - 1$ to n is shorter than dispersion time of vehicles accumulated at intersection n during last cycle, and vehicles from intersection $n - 1$ have to wait for some time to cross the intersection n . Consider

$$t'_n \leq \frac{\int_{t_n + \lambda_n C + (N-1)C}^{t_n + NC} k_n q_n(\tau) d\tau}{S_n}, \quad (5)$$

where t'_n is the travel time for vehicles to travel from intersection $n - 1$ to n ; t_n is the offset between intersection $n - 1$ and n ; C is the public cycle length obtained from Section 3.2; λ_n is the split of intersection n along the arterial direction; N is an integer number; k_n is the volume adjustment coefficient; $q_n(\tau)$ is the arrival flow rate function at intersection n , which can be assumed to follow any random distribution, and the Poisson distribution is used in this paper; S_n is the capacity of intersection n along the arterial direction.

Then, the dispersion time of vehicles at intersection n is formulated as follows:

$$T_n = \frac{\int_{t'_n}^{t_n + C} k_n q_n(\tau) d\tau + \int_{t_n + \lambda_n C + (N-1)C}^{t_n + NC} k_n q_n(\tau) d\tau}{S_n}. \quad (6)$$

In the numerator of (6), there are a total of two items. The first item is the undispersed queue vehicles during last cycle, and the second item is the vehicles arriving at intersection n from the upstream intersection $n - 1$.

Situation 2. Travel time from intersection $n - 1$ to n is longer than dispersion time of vehicles accumulated during last cycle, and vehicles from intersection $n - 1$ can cross intersection n directly without waiting. Consider

$$t'_n > \frac{\int_{t_n + \lambda_n C + (N-1)C}^{t_n + NC} k_n q_n(\tau) d\tau}{S_n}. \quad (7)$$

In this situation,

$$T_n = 0. \quad (8)$$

According to the definition of the uninterrupted ratio, we can formulate

$$M_n = \frac{\text{Num}'_n}{\text{Num}_n} = \frac{\int_{t_n + T_n}^{t_n + \lambda_n C} q_n(\tau) d\tau}{\text{Num}_n}. \quad (9)$$

Furthermore, to maximize the sum of uninterrupted ratio at all intersections, the real-time arterial coordination control model is formulated as

$$\max f(t_n, \lambda_n) = \sum_n \frac{\int_{t_n + T_n}^{t_n + \lambda_n C} q_n(\tau) d\tau}{\text{Num}_n} \quad (10)$$

$$\text{s.t.} \quad \begin{cases} t_n \geq 0 \\ 0 \leq \lambda_n \leq \lambda_{n,\max} \end{cases}$$

where $\lambda_{n,\max}$ denotes the maximum split along the arterial direction (0.8 in this paper) of intersection n .

Based on the public cycle length C and the vehicle dispersion time T_n , we can obtain the real-time offset t_n and the split λ_n of each intersection along the arterial by solving the nonlinear optimization problem in (10).

From the model formulation, we can find out that there is no limitation about geographic scheme of the arterial road or vehicle arrival distribution, and the optimized offset and split remain updated along with the time interval.

4. Solution of the Model

4.1. Genetic Algorithm for Dynamic Turning Fraction Estimation at Intersections. Since the objective function of (2) is a LAD formulation and there are equality constraints in the model, it is very difficult to be solved using traditional optimization method. Therefore, we develop a revised genetic algorithm (GA) for solution. The turning fractions $\xi_{ii}(k)$ are assumed to be 0; that is, the U-turn phenomena are neglected here.

We borrow the GA from our previous work [4] directly, and two important issues are described as following:

4.1.1. Encoding and Decoding. We use the binary encoding method in this paper. According to the equality constraint in (2), as well as the assumption that there is no U-turn at intersections, there are $s - 2$ independent turning fractions for each entering approach. Totally, there are $r(s - 2)$ independent turning fractions for an intersection during one time interval. These turning fractions to be estimated can be written as a matrix with r rows and $s - 2$ columns.

The decoding process is further represented as

$$\begin{aligned}
 \xi_{i1} &= \frac{ch_{i1}}{(2^{L_{GA}} - 1)} \\
 &\vdots \\
 \xi_{i(i-1)} &= \left(1 - \sum_{j=1}^{i-2} \xi_{ij}\right) \times \frac{ch_{i(i-1)}}{(2^{L_{GA}} - 1)} \\
 \xi_{ii} &= 0 \\
 \xi_{i(i+1)} &= \left(1 - \sum_{j=1}^i \xi_{ij}\right) \times \frac{ch_{ii}}{(2^{L_{GA}} - 1)} \\
 &\vdots \\
 \xi_{i(s-1)} &= \left(1 - \sum_{j=1}^{s-2} \xi_{ij}\right) \times \frac{ch_{i(s-1)}}{(2^{L_{GA}} - 1)} \\
 \xi_{is} &= 1 - \sum_{j=1}^{s-1} \xi_{ij},
 \end{aligned} \tag{11}$$

where ch_{ij} is a decimal real number transformed from the binary code of ξ_{ij} and L_{GA} is the length of each chromosome. As stated before, there is one turning fraction which is dependent on other $s - 1$ results, and we assume it to be as the one corresponding to the exiting approach with the index s .

Through the decoding method in (11), the estimated turning fractions will satisfy both equality and inequality constraints in (2); therefore, we do not need to perform the truncation and normalization processes [1], which are usually employed in turning proportions estimation.

4.1.2. Fitness Function. Since GA always tries to search for the maximum fitness of chromosome in the evolution process, we introduce a positive number to transform the objective function to the following fitness function:

$$\text{Fit} = \frac{U}{Z}, \tag{12}$$

where U is a positive constant and Fit is the fitness function.

The overall steps and other issues are similar to our previous work and some other existing GAs. We code this GA using M language of MATLAB software and then obtain the estimated dynamic turning fractions for each single intersection.

4.2. Solution of Real-Time Public Cycle Length Optimization at Intersections. Since (3) is an ordinary nonlinear optimization problem, it can be solved directly using existing mathematical methods. We code this model using Lingo software, and then obtain the optimized cycle length of each single intersection. The maximum one is taken as the public cycle length for the arterial road.

4.3. Genetic Algorithm for Real-Time Offset and Split Optimization at Arterial Corridors. Due to the integral formulation in the objective function of (10), its solution is very difficult. Here, we also design a GA for solving [15–17]. The overall steps are described as follows.

Step 1 (encoding). A real-coded scheme is adopted here to represent the feasible solutions during the GA evolution process.

Step 2 (fitness function). Since the objective function of (10) is a maximum formulation, we take it as the fitness function directly, as shown in the following equation:

$$\text{Fit} = f(t_n, \lambda_n) = \frac{\sum_n \int_{t_n}^{t_n + \lambda_n C} q_n(\tau) d\tau}{\text{Num}_n}. \tag{13}$$

Step 3 (selection). The classical Roulette wheel selection approach is employed for the selection operation in this paper.

Step 4 (crossover). Crossover operation is employed to pass the excellent genes of parent chromosomes to children chromosomes and to generate new chromosomes. In this paper, we use the following method to implement the crossover operation:

$$\begin{aligned}
 \text{chr}_{yj}^{\delta+1} &= \text{chr}_{yj}^{\delta} (1 - \mu) + \text{chr}_{zj}^{\delta} \mu, \\
 \text{chr}_{zj}^{\delta+1} &= \text{chr}_{zj}^{\delta} (1 - \mu) + \text{chr}_{yj}^{\delta} \mu,
 \end{aligned} \tag{14}$$

where μ is a random number between $[0, 1]$, $\text{chr}_{yj}^{\delta+1}$ and $\text{chr}_{zj}^{\delta+1}$ are genes from children chromosomes, chr_{yj}^{δ} and chr_{zj}^{δ} are genes from parent chromosomes, δ is the index of current generation, and j is the index of gene in the chromosome.

Step 5 (mutation). Mutation operation is implemented to ensure the local random search ability of GA, as well as to produce new children chromosomes. The following method is designed for the mutation operation:

$$\begin{aligned}
 \psi &= \rho \left(1 - \frac{\delta}{\Delta_{\max}}\right)^2, \\
 \text{chr}_{yj}^{\delta+1} &= \begin{cases} \text{chr}_{yj}^{\delta} + \psi (\text{chr}_{yj}^{\delta} - \text{chr}_{yj,\min}^{\delta+1}), & \omega \geq 0.5 \\ \text{chr}_{yj}^{\delta} + \psi (\text{chr}_{yj,\max}^{\delta+1} - \text{chr}_{yj}^{\delta}), & \omega < 0.5, \end{cases}
 \end{aligned} \tag{15}$$

where ρ and ω are random numbers between $[0, 1]$, $\text{chr}_{yj,\min}^{\delta+1}$ is the lower bound of chr_{yj} , $\text{chr}_{yj,\max}^{\delta+1}$ is the upper bound of chr_{yj} , and Δ_{\max} is the maximum generations.

Some important controlling parameters of the GA are set as follows: the population size is 60, the maximum number of generations is 120, the crossover rate is 0.7, and the mutation rate is 0.01.

We also code this GA using M language of MATLAB software and then obtain the optimized offset and split for the arterial road.

TABLE 1: Signal timings.

Scheme	Intersection	Cycle length (s)	Arterial split	Offset (s)
Current	I-1	85	0.52	—
	I-2	90	0.53	
	I-3	118	0.26	
MAXBAND	I-1	95	0.72	—
	I-2		0.60	22
	I-3		0.30	19
Proposed model-Cycle 1	I-1	101	0.70	—
	I-2		0.64	25
	I-3		0.30	22
Proposed model-Cycle 2	I-1	98	0.75	—
	I-2		0.62	26
	I-3		0.30	23
Proposed model-Cycle 3	I-1	92	0.76	—
	I-2		0.66	25
	I-3		0.30	22
Proposed model-Cycle 4	I-1	95	0.72	—
	I-2		0.64	26
	I-3		0.30	23
Proposed model-Cycle 5	I-1	100	0.75	—
	I-2		0.62	25
	I-3		0.30	22

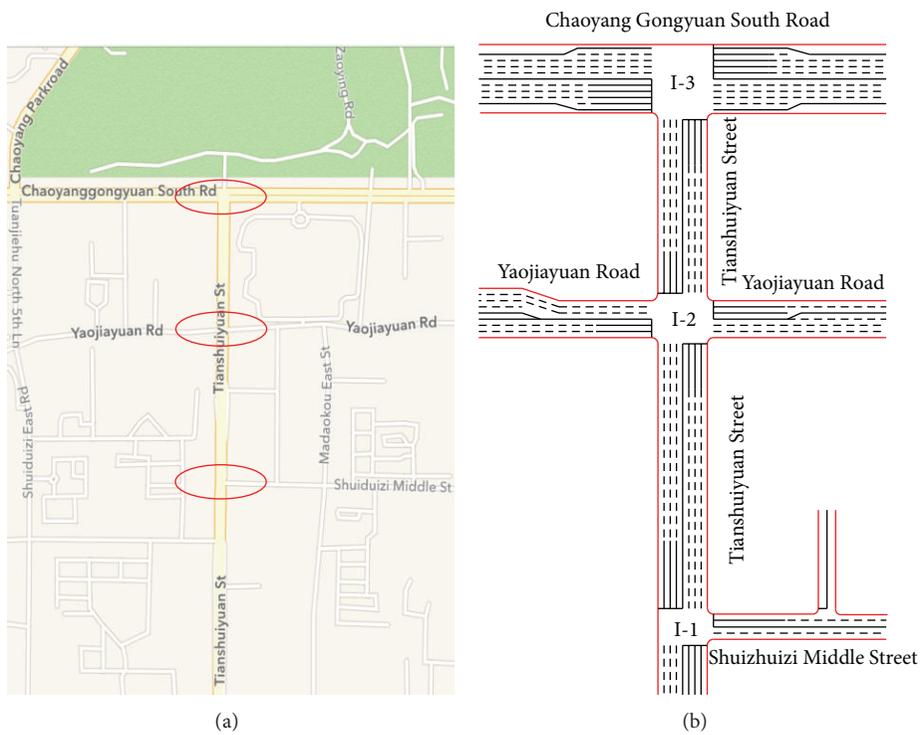


FIGURE 2: Graphical illustration of the case study arterial road.

5. Case Study

To investigate the performance of the proposed model framework, this paper implements a case study using practical data and the effects of the arterial control are evaluated through traffic simulation.

An arterial corridor around Tianshuiyuan block in Chaoyang district, in Beijing city, is taken as the case study area. Figure 2 presents a graphical illustration of the area.

Tianshuiyuan Street is the arterial road to be studied, and there are three intersections along it. Intersection (I-2) is employed as the key intersection, which is a crossroad with four approaches. The arterial direction is from south to north, that is, from I-1 to I-3.

To collect the field data, we implemented a survey around the case area and collected a great deal of data, including the time-varying entering and exiting link flows at all three intersections, the geographical information of the arterial, the existing signal timing plan, and queue length, delay time, and number of stops at all three intersections. All information needed in the case study can be extracted from the above survey data.

We further implemented the proposed model, as well as the existing MAXBAND method. Table 1 reports cycle length, arterial split, and offset of each intersection from current scheme, MAXBAND model, and the proposed model.

In the current scheme, all three intersections have fixed timing plans, and the arterial road is not coordinated at all. For the proposed model, we take 5 cycles, as examples here, and the public cycle length, arterial split, and offset are all illustrated in Table 1. We can also find out that all signal timings of the proposed model are updated in each cycle; therefore, they are real-time in nature.

To further evaluate the effects of the proposed model, we select three indices as evaluation criteria, including queue length, delay time, and number of stops. All these indices of current scheme were obtained through the field survey. Since it is impossible to adjust the signal timings in real world just for this case study, we simulated both MAXBAND method and the proposed model using Vissim software [21].

Figure 3 shows the simulation environment. The road network was created based on an original AutoCAD file, and the surveyed entering link counts at three intersections were taken as the input data, together with the current turning proportions and signal timings. Using the above three indices as evaluation criteria, the simulation model was calibrated by adjusting some parameters, such as speed, acceleration, and priority rules. The timings from the proposed methodology were then input to the simulation model, and all three evaluation indices under proposed timings were extracted through some detectors laid in the road network in Vissim.

We further compared the three simulated evaluation indices with those from current traffic control scheme. Tables 2, 3, and 4 report queue length, delay time, and number of stops, respectively.

The detailed graphical comparisons of the three evaluation indices are further illustrated in Figures 4, 5, and 6.

From Tables 2, 3, and 4 and Figures 4, 5, and 6, we can reach the following results.

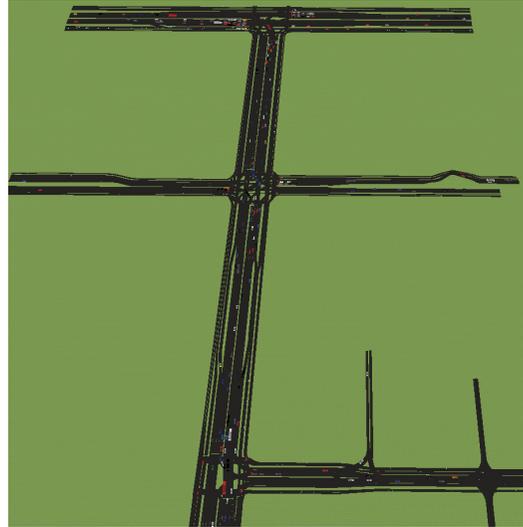


FIGURE 3: Graphical illustration of the simulation environment.

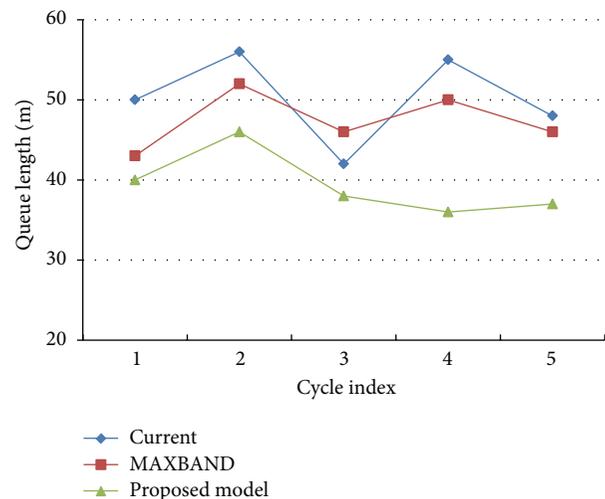


FIGURE 4: Graphical illustration of queue length.

- (1) Generally, the proposed arterial coordination control model yields the best results during all cycles in terms of all three indices, including queue length, delay time, and number of stops.
- (2) The MAXBAND method also outperforms current scheme during most cycles in all three indices, because the current timing plan is not coordinated along the arterial.
- (3) For delay time, the proposed model gets a little worse result than MAXBAND in the first cycle, which is due to some unknown reasons, but it is still better than the current scheme and does not influence the general outstanding performances of the proposed model.

TABLE 2: Comparison of queue length.

Cycle	Current (m)	MAXBAND (m)	Proposed model (m)	Improvement (%)	
				Over current	Over MAXBAND
1	50	43	40	20.0	7.0
2	56	52	46	17.9	11.5
3	42	46	38	9.5	17.4
4	55	50	36	34.6	28.0
5	48	46	37	22.9	19.6
Average	50.2	47.4	39.4	21.0	16.7

TABLE 3: Comparison of delay time.

Cycle	Current (s)	MAXBAND (s)	Proposed model (s)	Improvement (%)	
				Over current	Over MAXBAND
1	29	26	27	6.9	-3.9
2	35	32	30	14.39	6.3
3	28	31	26	7.1	16.1
4	32	30	28	12.5	6.7
5	30	29	27	10.0	6.9
Average	30.8	29.6	27.6	10.2	6.4

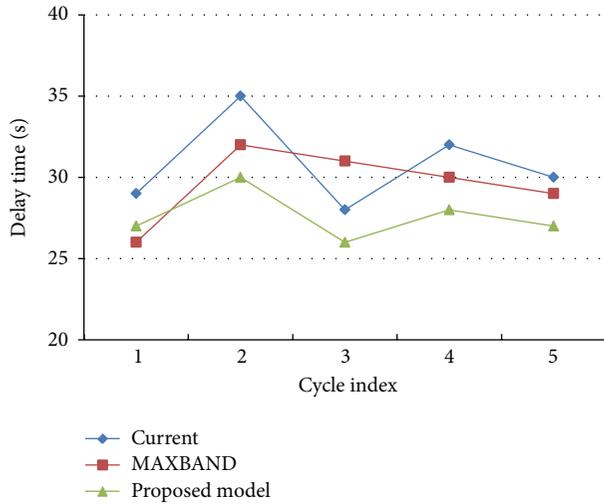


FIGURE 5: Graphical illustration of delay time.

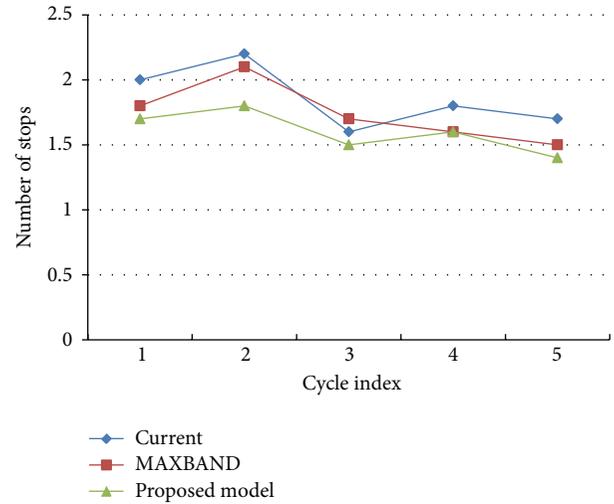


FIGURE 6: Graphical illustration of number of stops.

6. Conclusions

This paper presents a general model framework for real-time arterial coordination control. To provide important input data for the arterial control, this study first proposes a dynamic intersection turning fraction estimation model using the least absolute deviation formulation and designs a genetic algorithm for solution, integrating revised encoding and decoding methods. Based on the estimated time-varying turning proportions, this paper develops a nonlinear optimization model based on minimum delay to optimize the real-time cycle of each single intersection along the arterial road, and the maximum cycle is taken as the public arterial cycle. Furthermore, this paper puts forward a novel

optimization model based on minimum uninterrupted ratio and develops a genetic algorithm to optimize both offset and split of the arterial road, which are also time-varying. Time-dependent public cycle, offset, and split constitute the real-time timing plan of arterial coordination control scheme together. The proposed model removes most of the restrictions of conventional arterial control models and is a real-time control method in nature. The simulation experiments based on field data have confirmed the outstanding performances of the proposed model framework compared with both current scheme and MAXBAND method.

This paper can be enhanced in following directions. The first is to further consider the opposite direction of the arterial corridor and construct a bidirectional arterial coordination

TABLE 4: Comparison of number of stops.

Cycle	Current	MAXBAND	Proposed model	Improvement (%)	
				Over current	Over MAXBAND
1	2	1.8	1.7	15.0	5.6
2	2.2	2.1	1.8	18.2	14.39
3	1.6	1.7	1.5	6.3	11.8
4	1.8	1.6	1.6	11.1	0.0
5	1.7	1.5	1.4	17.7	6.7
Average	1.86	1.74	1.6	13.6	7.7

control model. The second is to extend the model to a wide area and develop a regional coordination control model. And the third is to integrate all three submodels in the framework together and formulate an accurate and efficient combined model.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work has been supported by National Natural Science Foundation of China Project (51208024), Excellent Talents Project of Beijing Municipal Committee Department of Organization (2013D005017000001), and the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions (CIT&TCD201404071). The authors are also thankful to the anonymous reviewers for their valuable comments and suggestions.

References

- [1] N. L. Nihan and G. A. Davis, "Recursive estimation of origin-destination matrices from input/output counts," *Transportation Research B*, vol. 21, no. 2, pp. 149–163, 1987.
- [2] N. L. Nihan and G. A. Davis, "Application of prediction-error minimization and maximum likelihood to estimate intersection O-D matrices from traffic counts," *Transportation Science*, vol. 23, no. 2, pp. 77–90, 1989.
- [3] M. G. H. Bell, "The real time estimation of origin-destination flows in the presence of platoon dispersion," *Transportation Research B*, vol. 25, no. 2-3, pp. 115–125, 1991.
- [4] P. Jiao, H. Lu, and L. Yang, "Real-time estimation of turning movement proportions based on genetic algorithm," in *Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, pp. 484–489, Vienna, Austria, September 2005.
- [5] K. Ashok and M. E. Ben-Akiva, "Estimation and prediction of time-dependent origin-destination flows with a stochastic mapping to path flows and link flows," *Transportation Science*, vol. 36, no. 2, pp. 184–198, 2002.
- [6] M. Bierlaire and F. Crittin, "An efficient algorithm for real-time estimation and prediction of dynamic OD tables," *Operations Research*, vol. 52, no. 1, pp. 116–127, 2004.
- [7] P. Lin and G. Chang, "A generalized model and solution algorithm for estimation of the dynamic freeway origin-destination matrix," *Transportation Research B: Methodological*, vol. 41, no. 5, pp. 554–572, 2007.
- [8] Y. Lou and Y. Yin, "A decomposition scheme for estimating dynamic origin-destination flows on actuation-controlled signalized arterials," *Transportation Research Part C*, vol. 18, no. 5, pp. 643–655, 2010.
- [9] P. Jiao, T. Sun, and H. Wang, "A Bayesian combined model for time-dependent turning movement proportions estimation at intersections," *Mathematical Problems in Engineering*, vol. 2014, Article ID 607195, 8 pages, 2014.
- [10] J. D. C. Little, "The synchronization of traffic signals by mixed-integer linear programming," *Operations Research*, vol. 14, no. 4, pp. 568–594, 1966.
- [11] J. D. C. Little, M. D. Kelson, and N. H. Gartner, "MAXBAND: a program for setting signal on arteries and triangular network," *Transportation Research Record*, no. 795, pp. 40–46, 1981.
- [12] C. K. Wong and S. C. Wong, "A lane-based optimization method for minimizing delay at isolated signal-controlled junctions," *Journal of Mathematical Modelling and Algorithms*, vol. 2, no. 4, pp. 379–406, 2003.
- [13] P. Lowrie, *The Sydney Coordinated Adaptive Control System—Principles, Methodology, Algorithms [Monograph]*, IEE Conference Publication, 1982.
- [14] D. I. Robertson and R. D. Bretherton, "Optimizing networks of traffic signals in real time—the SCOOT method," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 11–15, 1991.
- [15] B. Yu, Z. Yang, and C. Cheng, "Optimizing the distribution of shopping centers with parallel genetic algorithm," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 2, pp. 215–223, 2007.
- [16] B. Yu, Z. Yang, X. Sun, B. Yao, Q. Zeng, and E. Jeppesen, "Parallel genetic algorithm in bus route headway optimization," *Applied Soft Computing*, vol. 11, no. 8, pp. 5081–5091, 2011.
- [17] W. Juan, L. Huapu, Y. Xinxin, and B. Changzhi, "Genetic algorithm for multiuser discrete network design problem under demand uncertainty," *Mathematical Problems in Engineering*, vol. 2012, Article ID 686272, 17 pages, 2012.
- [18] B. Yu, Z. Yang, and B. Yao, "An improved ant colony optimization for vehicle routing problem," *European Journal of Operational Research*, vol. 196, no. 1, pp. 171–176, 2009.
- [19] B. Yu and Z. Z. Yang, "An ant colony optimization model: the period vehicle routing problem with time windows," *Transportation Research E: Logistics and Transportation Review*, vol. 47, no. 2, pp. 166–181, 2011.

- [20] B. Yu, Z. Z. Yang, P. H. Jin, S. H. Wu, and B. Z. Yao, "Transit route network design-maximizing direct and transfer demand density," *Transportation Research C*, vol. 22, pp. 58–75, 2012.
- [21] PTV Planung Transport Verkehr AG, *Vissim 5.30 User Manual*, PTV AG, Karlsruhe, Germany, 2010.

Research Article

Hybrid Model for Early Onset Prediction of Driver Fatigue with Observable Cues

Mingheng Zhang,^{1,2} Gang Longhui,³ Zhe Wang,² Xiaoming Xu,²
Baozhen Yao,^{1,2} and Liping Zhou⁴

¹ State Key Laboratory of Structural Analysis for Industrial Equipment, Dalian University of Technology, Dalian 116024, China

² School of Automotive Engineering, Dalian University of Technology, Dalian 116024, China

³ School of Navigation, Dalian Maritime University, Dalian 116026, China

⁴ Wuxi Mingda Traffic & Technology Consulted Co., Ltd., Wuxi 214125, China

Correspondence should be addressed to Baozhen Yao; yaobaozhen@yahoo.cn

Received 5 May 2014; Accepted 10 June 2014; Published 13 July 2014

Academic Editor: Rui Mu

Copyright © 2014 Mingheng Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a hybrid model for early onset prediction of driver fatigue, which is the major reason of severe traffic accidents. The proposed method divides the prediction problem into three stages, that is, SVM-based model for predicting the early onset driver fatigue state, GA-based model for optimizing the parameters in the SVM, and PCA-based model for reducing the dimensionality of the complex features datasets. The model and algorithm are illustrated with driving experiment data and comparison results also show that the hybrid method can generally provide a better performance for driver fatigue state prediction.

1. Introduction

Fatigue driving is a significant contributing factor to road related crashes worldwide, which will endanger lives of drivers and passengers and can cause serious accidents along major roads [1]. Studies show that 15–20% of car crashes may be attributable to driver sleepiness in high income countries [2]. If symptoms of fatigue could effectively warn the drivers and corrective measures could be taken, then disastrous outcomes would have been prevented. Researchers have made great efforts in this field. At present, the techniques for preventing driver's fatigue can be classified into the following categories.

- (1) Technologies to assess the vigilance capacity of an operator before the work is performed [3].
- (2) Mathematical models of dynamic alert [4].
- (3) Vehicle-based performance technologies that detect the behavior of the driver by monitoring the transportation hardware systems such as steering wheel movements and acceleration [5, 6].

- (4) Real-time technologies for monitoring the driver's state, including intrusive [7, 8] and nonintrusive monitoring systems [9–11].

The method assessing the vigilance before the work is performed investigates driver's subjective feeling of tiredness to determine whether the fatigue happens or not. This method is greatly influenced by individual differences, so it is only taken as a comprehensive qualitative evaluation method for driver fatigue analysis. The method based on vehicle performance uses visual sensor, speed sensor, and angle sensors to monitor vehicle running state such as speed, steering angle, or other features. The driver's spirit state can be inferred from this detected information. This method can make full use of the vehicle devices, so few additional equipment are required, but it is limited with the driver's driving habits, vehicle type, and road conditions. Experiments have demonstrated that the driver in fatigue would exhibit some visual cues [12]. Therefore, at present, the method based on machine vision has been a leading technique due to its hardware's characteristics such as small size, low cost, and nonintrusiveness to drivers.

Various visual parameters, such as eye blinking [9], head movement [13], face orientation [10], and other features, can be extracted when the driver's fatigue occurs. But, it is inaccurate when prewarning is triggered according to a certain specific aspect, because the driver's fatigue is not directly observable but can only be inferred from the information available. There are numbers of reasons for adopting the information fusing method to resolve the driver fatigue detection issues. Firstly, the driver's fatigue derived from the contextual features contains much subjectivity that cannot always reflect the real objectivity; secondly, inferring the driver's fatigue from his/her facial expression is not always reliable because of the following two limitations: (a) the current techniques for image processing cannot always ensure the recognition accuracy; (b) an introverted person might have a tendency to control his or her displays of emotions, which leads to an inaccurate interpretation of the facial expression. Thus, to fuse as many as possible features from uncertain events is necessary to make an accurate inference for monitoring fatigue effectively.

Since there are various nonlinear, stochastic, time-varying factors during the driver fatigue prediction, it is very difficult to predict the fatigue state accurately. Recently, SVM, a novel machine learning algorithm, has been proved, that is, a promising tool for both data classification and regression [14]. It shows high resistance to the over fitting problem, achieving high generalization performance in solving various time series forecasting problems, which has been applied in prediction of time series [15]. These successful applications motivate us to apply SVM in the driver fatigue state prediction.

The parameters selection in SVM, which greatly impacts the performance of SVMs, need to be optimized and set by users. Many literatures have proved that the parameters in SVM play an important role in its performance [16]. Improper selection of the parameters could cause either the overfitting or underfitting of the training data points. These parameters mainly include the penalty factor C and the parameters of kernel function (for instance, parameter γ of RBF kernel function). At present, grid search algorithm [17] is the most reliable method for the offline data training process. However, for large scale or real-time feature practice application, the considerable searching time cannot be accepted. Many literatures suggested that heuristic algorithms were good choice to solve this kind of problems [18–28]. Selakov et al. [29] propose a model for short term electrical load forecasting based on particle swarm optimization (PSO) and SVM. The obtained results show better accuracy compared to results generated with classical methods. A discrete gravitational search algorithm is combined with SVM which is presented for improving classification accuracy with an appropriate feature subset in binary problems [30]. Hou and Li [16] presented evolution strategy with covariance matrix adaptation to identify the parameters in SVM.

The purpose of this paper is to build a hybrid model for driver fatigue state prediction, which is named SVM-GA model. Within the proposed model, several cues are considered including PERCLOS AECS and PNS. The structure of this paper is organized as follows: the problem description

and the architecture of the solution for the driver fatigue state prediction are given out in Section 2. In Section 3, experiments and results are discussed; and lastly, the conclusions are given together with suggestions for further study.

2. Problem Description

As discussed above, fusing as many as possible features from uncertain events is necessary to make an accurate inference for predicting fatigue effectively. According to this problem, fatigue state is obviously the target hypothesis variable that we intend to infer while other visual cues, which are symptoms of fatigue, are information variables. The main purpose of a PSO-SVM model is to predict the unobserved events from the observed or contextual data. Due to there are so many factors reflecting human fatigue as discussed, it is impossible to include all of them into a SVM model. Hence, only the most significant ones are incorporated.

2.1. Observable Cues Analysis. As mentioned above, the driver fatigue can be inferred from the facial expression or observable variables. Various visual cues/parameters can be extracted when the driver's fatigue occurs, as discussed in Section 1. In this research field, the challenge is that we cannot infer the fatigue level through a certain specific aspect for a number of reasons. Therefore, fusing as many information that we can get may be the only way out to resolve the driver fatigue detection issues. But, unfortunately, it is impossible to include all of the visual parameters in the fatigue model. Hence, in this paper, only the following factors that play more contributions to the fatigue are incorporated.

2.1.1. PERCLOS Analysis. PERCLOS is the percentage of eyelid closure over the pupil over time and reflects slow eyelid closures ("drows") rather than blinks. A PERCLOS drowsiness metric was established in a 1994 driving simulator study as the proportion of time in a minute that the eyes are at least 80 percent closed [31]. Based on research by Fairbanks and colleagues, the federal highway administration (FHWA) and National Highway Traffic Safety Administration (NHTSA) consider PERCLOS to be among the most promising known real-time measures of alertness for in-vehicle drowsiness-detection systems [32].

2.1.2. AECS Analysis. AECS refers to the average eye closure speed, which is usually described via the time elapse from eyelids fully open to close completely. Research shows that the closure speed has obvious difference between the driver in awake and fatigue state [33], and this difference can reach more than eight times. In addition, there exists a higher correlation between AECS and PERCLOS. Therefore, the AECS may be a good visual observable variable for the driver fatigue model design.

2.1.3. PERLVO Analysis. PERLVO refers to percentage of mouth large vertical open over time. Research shows that this variable is obviously different when a driver's yawning fatigue or distraction state due to talking or conversation occurred [34]. Yawn may not occur with everyone when he/she falls

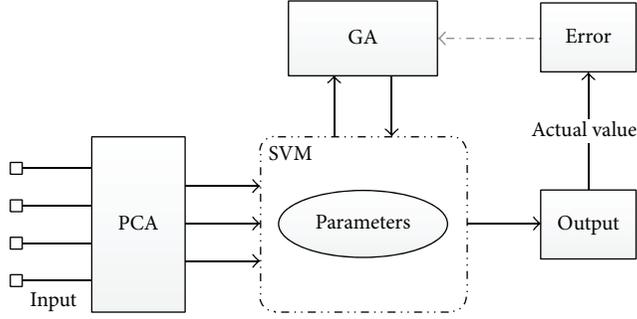


FIGURE 1: Flow-chart of the hybrid model.

into fatigue, but the PERLVO may be necessary supplement for the driver fatigue model design.

2.1.4. PERTITL Analysis. PERTITL refers to percentage of head tilt over time, that is, the percentage that the time of driver's looking up or down accounts for a certain time. It is used to describe the head movement of a driver when he/she falls into fatigue.

2.1.5. PNS Analysis. PNS refers to the percentage of non-steering, that is, the level of the steering wheel without any movement during a period of time. Research result show that, relative to the wake state, the PNS value is significantly increased when the driver is in the fatigue or sleep state. But, unfortunately, it is very difficult to distinguish the state between the fatigue and the sleep [35].

2.1.6. SDLP Analysis. SDLP refers to the standard deviation of lane position, that is, the level of a vehicle transverse the lane mark during a period of time. When the fatigue occurs, before the accident happens, vehicle sometimes could cross the lane. Therefore, researchers worldwide have been developing vision-based systems for lane keeping on unintended lane departures [5]. In these researches, two warning criteria are usually considered: the lateral offsets and the time to lane-crossing (TLC). Therefore, the SDLP variables are useful cues for reflecting the driver's fatigue. It should be considered into the driver fatigue model design.

2.2. Architecture of the Solution. Based on the above analysis, the architecture of the solution can be proposed. The proposed hybrid model consists of three modules: PCA unit, SVM unit, and GA unit. The hybrid model can be described as in Figure 1. In subsequent sections, three submodels will be discussed, respectively.

2.2.1. Support Vector Regression. As a new and promising technique for classification and regression problems, SVM can be adjusted to map the complex input-output relationship for the nonlinear system without being dependent on the specific functions. Unlike other nonlinear optimization methods, the solution of SVM always can achieve the global optimal solution without limitation to a local minimum point and it shows the strong resistance to the overfitting problem and the high generalization performance. Support vector

regression is a common term for implementation of SVM in regression problems.

Given the training data set $(x_1, y_1), (x_2, y_2), \dots, (x_i, y_i)$ ($x_i \in X \in R^n, y_i \in Y \in R, x_i \in X \subseteq R^n, y_i \in Y \subseteq R$), n is the number of samples. $\{x_i, y_i\}$ are pair of input and output vectors. SVM make use of a nonlinear mapping function to map x into a high-dimensional feature space H in which linear approximation is conducted to find the mapping function so that we can get a better approximation for the given data samples. Based on the statistical learning theory, we can get the function as follows: $(x_1, y_1), (x_2, y_2), \dots, (x_i, y_i)$

$$f(x) = \omega \phi(x) + b, \quad (1)$$

where ω is weight factor and b is the threshold value. The input samples are mapped to higher dimensional space by using kernel function ϕ . The kernel function is responsible for nonlinear mapping between input and feature space. The previous researches [36, 37] suggested that radial basis function (RBF) kernel had less numerical difficulties with application and was efficient for traffic state prediction. Thus, RBF kernel function is used for the SVM model in this study.

Regression can be defined as a problem that minimized the risks for a loss function. The optimal regression function is the minimum and regularized generic function under certain constraints:

$$\begin{aligned} \min_{w, b, \xi, \xi^*} \quad & \frac{1}{2} w^T w + C \sum_{i=1}^l \xi_i + C \sum_{i=1}^l \xi_i^* \\ \text{subject to} \quad & w^T \phi(x_i) + b - z_i \leq \epsilon + \xi_i \\ & z_i - w^T \phi(x_i) - b \leq \epsilon + \xi_i^* \\ & \xi_i, \xi_i^* \geq 0, \quad i = 1, \dots, l. \end{aligned} \quad (2)$$

The dual is

$$\begin{aligned} \min_{\alpha, \alpha^*} \quad & \frac{1}{2} (\alpha - \alpha^*)^T Q (\alpha - \alpha^*) \\ & + \epsilon \sum_{i=1}^l (\alpha_i + \alpha_i^*) + \sum_{i=1}^l z_i (\alpha_i - \alpha_i^*) \\ \text{subject to} \quad & \sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0, \\ & 0 \leq \alpha_i, \quad \alpha_i^* \leq C, \quad i = 1, \dots, l, \end{aligned} \quad (3)$$

where $Q_{ij} = K(x_i, x_j) \equiv \theta(x_i)^T \theta(x_j)$.

The approximate function is

$$f(x) = \sum_{i=1}^l (-\alpha_i + \alpha_i^*) K(x_i, x_j) + b, \quad (4)$$

where $K(x_i, x_j) = \phi(x_i)$ denotes the inner product of the vector in the feature space. All of the kernel functions can be directly computed in the input space.

2.2.2. GA for Parameters Optimization. Although SVM is feasible and applicable in predicting the driver fatigue state, there are some parameters, which greatly impact the performance of SVMs, need to be optimized. In general, for the RBF kernel, as a nonlinearly kernel function, the parameters such as C , γ , and ϵ are the key elements and they directly decide about the prediction performance of SVM. Thus, the parameter optimization is important for improving the prediction accuracy. In this paper, genetic algorithm is used for searching the best parameters for the presented SVM model. In general, the process of GA can be briefly described as follows.

(1) *Encoding of Chromosome.* In GA, a standard representation of each candidate solution is as a chromosome that is composed of “genes.” For the SVM parameters optimization problem in this paper, the real encodings were adopted since the parameters C , γ , and ϵ are continuous-valued. Each chromosome consists of gene_1^g , gene_2^g , and gene_3^g , which represent the three parameters, respectively. Here g is the current generation. In order to reduce the search space, the previous literature has given out the recommended searching space which, respectively, attribute to the range $C \in [2^{-5}, 2^5]$, $\epsilon \in [2^{-13}, 2^{-1}]$, and $\gamma \in [0, 2]$.

(2) *Fitness Function.* A fitness function is a particular type of objective function that is used to summarize how close the possible solutions is to achieving the set aims. For the SVM parameters optimization problem in this paper, considering that GA is always finding the maximum fitness of the individual chromosome, mean squared error (MSE) is adopted as follows:

$$\text{fitness} = \frac{1}{l} \sum_{i=1}^l (f(x_i) - y_i)^2, \quad (5)$$

where $f(x_i)$ is the prediction value by the SVM model; y_i is the observed value; l is the number of observation variables.

(3) *Selection Operation.* During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. In this paper, the roulette selection strategy is adopted. Based on the fitness calculation results, the sum fitness value of the entire population and then the ratio corresponding to each chromosome are obtained. In the next step, a random number (range from 0 to 1) is used for determining the range of the cumulative probability. The chromosome falling within the expected range is selected out.

(4) *Genetic Operators.* For each new solution to be produced, a pair of “parent” solutions is selected for breeding from the pool selected previously. A second generation population of solutions is generated from those selected through a combination of genetic operators: crossover (also called recombination), and mutation.

Crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next. It is analogous to reproduction and biological crossover. Cross over is a process of taking more than one parent solutions and producing a child solution from them. In literature [38], an arithmetic crossover operator is used. Consider

$$\begin{aligned} \text{gen}_{k,I}^t &= \alpha_i \text{gen}_{k,I}^{t-1} + (1 - \alpha_i) \text{gen}_{k,II}^{t-1} \\ \text{gen}_{k,II}^t &= \alpha_i \text{gen}_{k,II}^{t-1} + (1 - \alpha_i) \text{gen}_{k,I}^{t-1}, \end{aligned} \quad (6)$$

where gen_k^{t-1} is the “parent” chromosomes; gen_k^t is the “child” chromosomes; α_i is a random probability value with range $(0, 1)$.

Mutation is a genetic operator used to maintain genetic diversity from one generation chromosome to the next. Mutation occurs during evolution according to a user-definable mutation probability. A very small mutation rate may lead to premature convergence of the genetic algorithm and a very high rate may lead to loss of good solutions unless there is elitist selection. In general, the mutation rate is defined with the range $[0.001, 0.1]$. In this paper, according to the previous literature, the mutation rate is set to 0.05.

(5) *Termination.* This generational process is repeated until a termination condition has been reached. In this paper, the search loop continues until $\text{MSE}_n - \text{MSE}_{n-1} < 0.0001$ or the number of generation reaches the maximum number of generations T_{\max} .

2.2.3. PCA for Reducing the Dimensionality of Input Datasets. In order to speed the velocity of SVM training and prediction with maintaining the main information of samples and not changing the distribution of samples, in this paper, PCA method is used before the observable variables are input into the SVM model for training and prediction.

In PCA, which kind of data is used will has a great influence for the result of the analysis. Existing literature shows that the analysis results are different when the covariance matrix and the correlation matrix are used for PCA, respectively. In this paper, the various observable variables can reflect the driver fatigue state from different aspects. Thus, there exists some relevance between the variables. Meanwhile, the dimensional of each variable is different. Therefore, based on the above analysis, we consider using the correlation matrix for the subsequent PCA process.

(1) *Calculate the Correlation Matrix.* Given the original data matrix $X = (x_{ij})_{n \times p}$, n is the number of the observable samples and p is the dimensions for each sample. Then the correlation matrix can be calculated with the following formula:

$$r_{x_i, x_j} = \frac{\text{cov}(X_i, X_j)}{\sqrt{D_{x_i} D_{x_j}}}, \quad (7)$$

where, $\text{cov}(X_i, X_j)$, D_{x_i} and D_{x_j} are the covariance and variance of the original data, respectively.

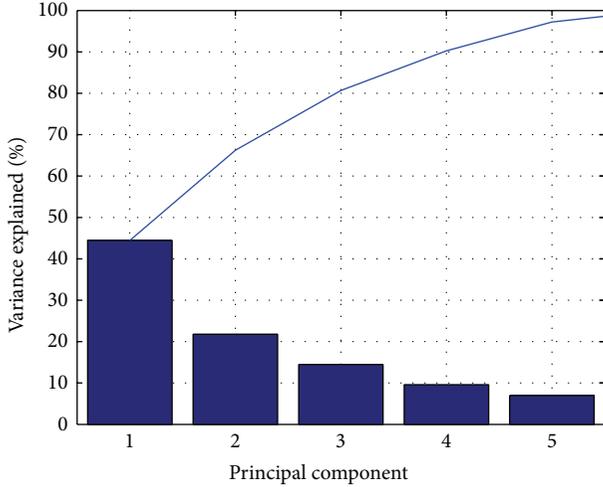


FIGURE 2: Distribution curve for the cumulative energy of PCA with original data.

In this paper, the original data matrix X is composed with the observable variables such as PERCLOS and AECS. Thus, the dimensions of samples are 6 types of the observable variables.

(2) *Calculate the Eigenvectors and Eigenvalues.* Compute the matrix V of eigenvectors which diagonalize the correlation matrix R

$$V^{-1}RV = D, \quad (8)$$

where D is the diagonal matrix of eigenvalues of R . Matrix D will take the form of an $p \times p$ diagonal matrix, where $D(k, l) = \lambda_k$ for $k = l$ is the k th eigenvalue of the correlation matrix R . Matrix V , also of dimension $p \times p$, contains p column vectors, each of length p , which represent the p eigenvectors of matrix R .

(3) *Choosing Components and Deriving the New Data Set.* Once the eigenvectors and eigenvalues of matrix R obtained, the eigenvalues and eigenvectors are sorted in order to decrease eigenvalue. The eigenvalues represent the distribution of the source data's energy among each of the eigenvectors. The cumulative energy content g for the j th eigenvector is the sum of the energy content across all of the eigenvalues from 1 through j . We can use the vector g as a guide in choosing an appropriate value. In this paper, the threshold value for cumulative energy is given as 0.9. Figure 2 is the distribution curve for the cumulative energy of PCA with original data. So the following PCA analysis result can be drowned: with the given threshold value 0.9, the main principal component number is 4. With this operator, the observable variables dimension is changed from 6 to 4.

Based on the above analysis, the new data set can be derived with the following formula:

$$X = V \times R, \quad (9)$$

where X is the new feature dataset matrix which will be input into the GA-SVM model including 4 main principal components with 0.9 cumulative energy loadings.

3. Case Study

In this section, an experiment has been carried out to verify the presented model. Based on the above analysis in Section 2.1, the observable variables used are divided into three types: the eye movement parameters are obtained by eye tracker; the face visual movement parameters are obtained by high-definition camera; and the vehicle running state parameters are collected by a driver simulator. In addition, the parameters obtained from physiological parameter tester are used as the standard that verifies whether the driver is fatigue or not.

A driver simulator equipped with data acquisition sensors is shown as Figure 3. The simulator is equipped with visual and auditory function for real-time simulation. In which, the visual system produces a 3D road scene that can used for simulating the vehicle running state and the visual effect under the driver's operation such as kinds of traffic flow, road lines, different weather, and other traffic things on the road. The auditory system can simulate the sound of different things perfectly, including the engine, the horn, the environment, and the wind. The tester manipulates the steering wheel and sends command to the simulator. Then, various sensors installed on the simulator can interpreter this command and give out the parameters such as the throttle opening and the steering wheel angle. The sampling frequency of the simulator we adopt is 50 Hz.

The experiments lasted from 1:00 P.M to 3:00, sampling at every one minutes, during which each participant was asked to operate the driving simulator at the speed of 90 km/h in highway environment, and his/her dynamic facial image was obtained at the sampling rate of 100 fps. Signals were processed with the corresponding methods to form the observable datasets. The experimental cases and simulation scenario were shown as Figure 4.

3.1. Parameter Identification. In the experiment, there are 100 samples in total. The data is divided into two subsets, which represent training samples and testing samples, respectively. Firstly, about 20% of samples data were set as testing data. Then, 80% of the remaining samples data were assigned to training.

To properly optimize the three parameters C , γ , and ϵ for SVM, GA is used. Before the implementation, four GA parameters, namely, p_c , p_m , p_{size} , and T_{max} , need to be predetermined. In general, p_c varies from 0.3 to 0.9; p_m varies from 0.01 to 0.1; p_{size} is the population size which is set according to the size of the samples. T_{max} is the maximum number of generations which can be determined according to a good convergence of the calculation. Considering the features of this problem and our experiences, the characteristic of GA can be acquired, as can be seen in Table 1. Then GA continues running for searching the best result for parameters C , γ , and ϵ . Figure 5 shows the MSE curve for parameters optimal with given conditions. It can be observed that the optimal fitness

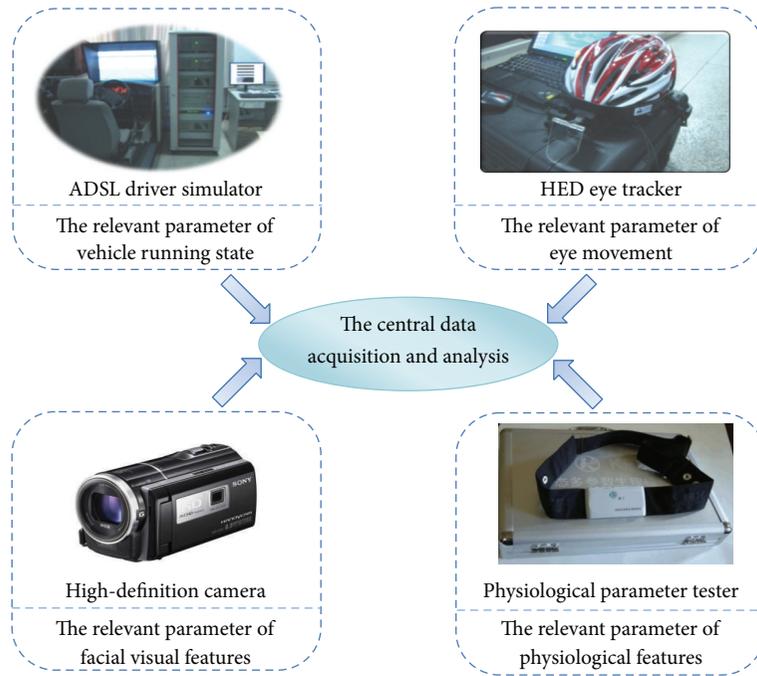
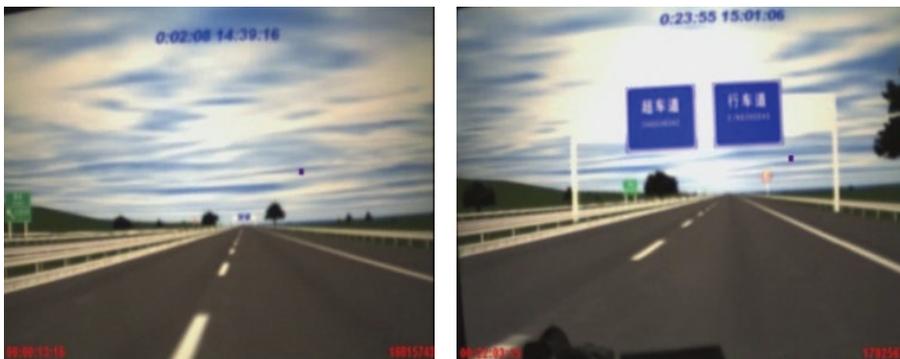


FIGURE 3: The hardware system for driver fatigue experiment.



(a) Participants are driving with simulator



(b) The experiment scenario

FIGURE 4: Experiment cases.

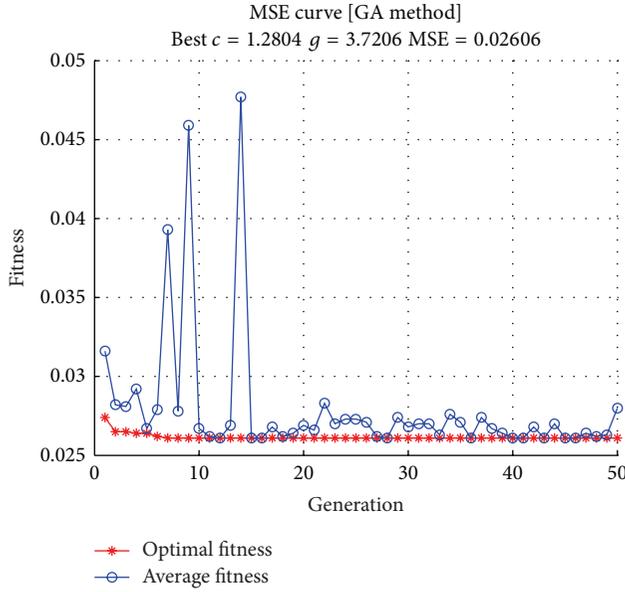


FIGURE 5: MSE curve for SVM parameter searching.

TABLE 1: The characteristics of GA.

Parameter	P_c	P_m	P_{size}	T_{max}
Value	0.6	0.05	5	500

decreases fast before the 5th generation, and then it changes smoothly. The least prediction error appears in about the 45th generation, and it almost remains unchanged, and the process loop terminates in the 50th generation. This means that GA has a good convergence for the SVM parameters searching problem. Finally, the three parameters C , γ , and ϵ were optimized as (1.2804, 3.7206, and 0.0018) with the best MSE value 0.026 for the practical prediction model of the driver fatigue state prediction SVM-GA.

3.2. Results and Analysis. Based on the parameters selected, the final SVM-GA hybrid model for driver fatigue state prediction is confirmed. In order to verify the actual performance of the prediction model presented above, tests were conducted on the train set and test set; respectively, the result was shown as Figure 6. In this test, 20% of the samples data was used for predicting the driver fatigue state. From the result, it is obviously indicated that the prediction curves agree well with the actual observing data. In addition, the analysis result of prediction error is shown that the mean relative error (MSE) is 1.9% with the test set samples. Thus, it can be seen that the presented SVM-GA model is feasible and applicable for driver fatigue state prediction.

In order to evaluate the impact performance with PCA process and different parameters searching method, tests were conducted on the sample data, respectively. The test results are shown as Figures 7-8 and details information of time consumption and MSE are listed in Table 2.

According to the comparison result with/without PCA process in Figure 7, it is indicated that the preprocesses of

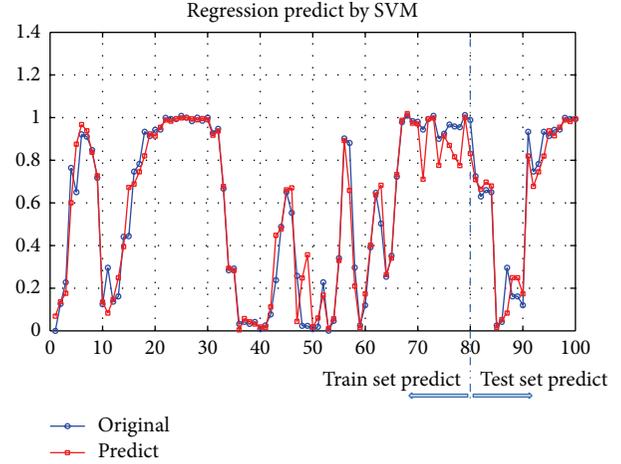


FIGURE 6: Prediction result for SVM-GA model.

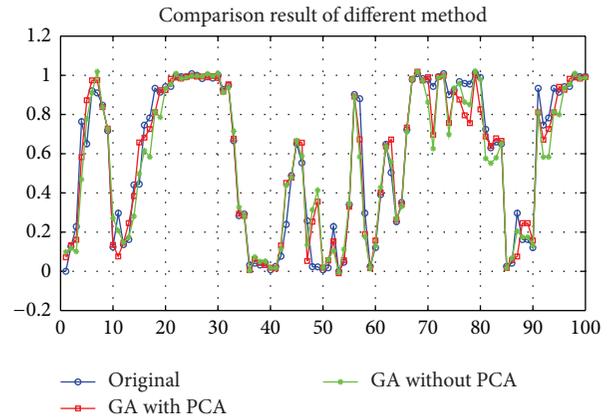


FIGURE 7: Comparison test result with/without PCA process.

TABLE 2: Comparison result for the different methods.

Method	Time elapse (s)	MSE (Test set)	MSE (Train set)	Best MSE
GA with PCA	2.34	0.0094	0.019	0.0284
GA without PCA	2.65	0.0055	0.01	0.0259
CG with PCA	4.62	0.0075	0.0104	0.0281
CG without PCA	5.54	0.0053	0.0094	0.0259

PCA for original data dimension reduction can play a role in improving the prediction result to some extent, that is, in Table 2, the MSE (on test set) for the two methods are 0.0094 and 0.0055, respectively. But, on the contrary, the time consumption for data training and prediction is 2.34 s and 2.65 s. It can be explained in such a way that the PCA process, in fact, is a statistical procedure that uses orthogonal transformation to convert a set of sample data with possibly correlated variables into a set of values of linearly uncorrelated variables. During the transformation, the principal components are extracted for replacing the original sample data and inputting into the SVM model for training and prediction. So there is

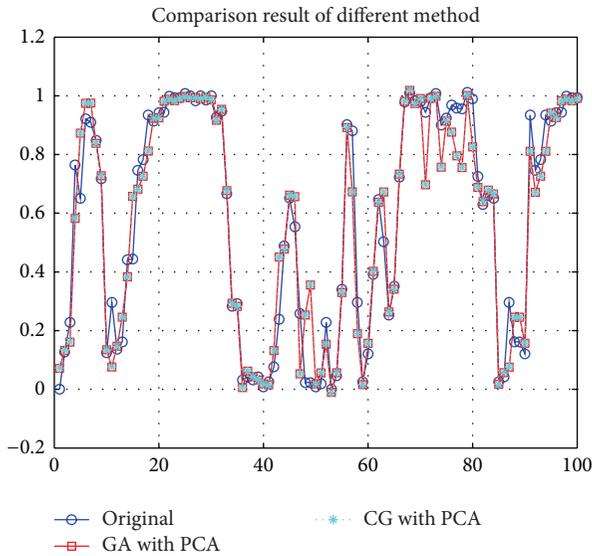


FIGURE 8: Comparison result with GA and CG.

some information loss relative to the original data. This procedure can reduce the data dimension and reveal the internal structure of the data in a way that best explains the variance. But, on the contrary, the loss information may decrease the prediction accuracy in the SVM model applying period. Thus, in practical application, there is a choice must be made between the computation speed and the prediction accuracy.

In order to evaluate the parameters optimization result through GA method proposed in this paper, a cross verification method test, often used for the SVM parameter searching, is conducted. From the result in Figure 8 and Table 2, it is obvious that the results are approximate with the two methods. But the time consumption is significantly different for the SVM model's parameter optimization. There is a great of advantages in SVM-GA method for improving calculation speed.

4. Conclusions

State prediction is an important phase for the driver fatigue early onset prewarning. The state can be inferred from the detected cues such as PERCLOS, AECS, and PNS. Due to the complexity of the information obtained, this paper attempts to develop a hybrid model based on SVM and GA to predict the driver fatigue state during the early onset phases. To evaluate the performance of the proposed method, an experiment with driving experiment data is carried out. The results show that GA has a good convergence and relative stable performance. Furthermore, the comparison of results between different methods suggests that the SVM-GA provides lower prediction errors and time consumption than the other approaches. This indicates that SVM-GA seems to be a powerful tool for driver fatigue state prediction during the early onset phases.

In summary, it is believed that a hybrid model proposed in this paper can provide a better performance for early

onset prediction of driver fatigue. However, there are some unresolved issues to be discussed in the future work. First of all, the driver fatigue has an accumulative property and fatigue is developed over time. Future work is needed to consider this property and extend the current model to be accordant with the practical situation.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was jointly supported by grants from the Humanities and Social Sciences Foundation of the Ministry of Education in China (Project no. 12YJCZH280), the Ph.D. Programs Foundation of Ministry of Education of China (Project no. 20112125120004), the National Natural Science Foundation of China (Project nos. 51208079 and 11272075), and the Fundamental Research Funds for the Central Universities (Project no. 3013-852019).

References

- [1] S. K. L. Lal, A. Craig, P. Boord, L. Kirkup, and H. Nguyen, "Development of an algorithm for an EEG-based driver fatigue countermeasure," *Journal of Safety Research*, vol. 34, no. 3, pp. 321–328, 2003.
- [2] J. L. Connor, "The role of driver sleepiness in car crashes: a review of the epidemiological evidence," in *Drugs, Driving and Traffic Safety*, J. Verster, S. R. Pandi-Perumal, J. Ramaekers, and J. Gier, Eds., pp. 187–205, Birkhäuser, Basel, Switzerland, 2009.
- [3] H. Y. Shu, F. Q. Li, S. P. Yi, H. Fujimoto, and Y. Ueno, "Comprehensive evaluation method for automotive driving fatigue," *Journal of Shanghai Jiaotong University*, vol. 42, no. 8, pp. 1338–1343, 2008.
- [4] D. Dawson, N. Lamond, K. Donkin, and K. Reid, "Quantitative similarity between the cognitive psychomotor performance decrement associated with sustained wakefulness and alcohol intoxication," in *Managing Fatigue in Transportation*, pp. 231–256, 1998.
- [5] J.-G. Wang, C.-J. Lin, and S.-M. Chen, "Applying fuzzy method to vision-based lane detection and departure warning system," *Expert Systems with Applications*, vol. 37, no. 1, pp. 113–126, 2010.
- [6] C. Wu, C. Lin, and C. Lee, "Applying a functional neurofuzzy network to real-time lane detection and front-vehicle distance measurement," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 42, no. 4, pp. 577–589, 2012.
- [7] M. V. M. Yeo, X. P. Li, K. Q. Shen, and E. P. V. Wilder-Smith, "Can SVM be used for automatic EEG detection of drowsiness during car driving?" *Safety Science*, vol. 47, no. 1, pp. 115–124, 2009.
- [8] S. Hu and G. Zheng, "Driver drowsiness detection with eyelid related parameters by Support Vector Machine," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7651–7658, 2009.
- [9] A. Picot, S. Charbonnier, and A. Caplier, "Drowsiness detection based on visual signs: Blinking analysis based on high frame rate video," in *Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC '10)*, pp. 801–804, Austin, Tex, USA, May 2010.

- [10] M. F. Hansen, G. A. Atkinson, L. N. Smith, and M. L. Smith, "3D face reconstructions from photometric stereo using near infrared and visible light," *Computer Vision and Image Understanding*, vol. 114, no. 8, pp. 942–951, 2010.
- [11] S. Vitabile, A. de Paola, and F. Sorbello, "A real-time non-intrusive FPGA-based drowsiness detection system," *Journal of Ambient Intelligence and Humanized Computing*, vol. 2, no. 4, pp. 251–262, 2011.
- [12] Q. Ji, Z. Zhu, and P. Lan, "Real-time nonintrusive monitoring and prediction of driver fatigue," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 4, pp. 1052–1068, 2004.
- [13] A. Doshi and M. Trivedi, "A comparative exploration of eye gaze and head motion cues for lane change intent prediction," in *Proceedings of the 2008 IEEE Intelligent Vehicles Symposium*, pp. 49–54, June 2008.
- [14] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [15] L. J. Cao and F. E. H. Tay, "Support vector machine with adaptive parameters in financial time series forecasting," *IEEE Transactions on Neural Networks*, vol. 14, no. 6, pp. 1506–1518, 2003.
- [16] S. Hou and Y. Li, "Short-term fault prediction based on support vector machines with parameter optimization by evolution strategy," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12383–12391, 2009.
- [17] C. W. Hsu, C. C. Chang, and C. J. Lin, *A Practical Guide to Support Vector Classification*, Technical Report, National Taiwan University, 2003.
- [18] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research C*, vol. 43, pp. 233–248, 2014.
- [19] B. Z. Yao, P. Hu, M. H. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem," *Simulation*, vol. 89, no. 6, pp. 762–770, 2013.
- [20] B. Yao, C. Yang, J. Yao, and J. Sun, "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.
- [21] B. Yao, C. Yang, J. Hu, J. Yao, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6–7, pp. 2127–2131, 2011.
- [22] B. Z. Yao, P. Hu, M. H. Zhang, and X. M. Tian, "Improved ant colony optimization for seafood product delivery routing problem," *ROMET-TRAFFIC & Transportation*, vol. 26, no. 1, pp. 1–10, 2014.
- [23] B. Z. Yao, J. B. Yao, M. H. Zhang, and L. Yu, "Improved support vector machine regression in multi-step-ahead prediction for tunnel surrounding rock displacement," *Scientia Iranica*, 2013.
- [24] B. Yu, Z. Yang, K. Chen, and B. Yu, "Hybrid model for prediction of bus arrival times at next station," *Journal of Advanced Transportation*, vol. 44, no. 3, pp. 193–204, 2010.
- [25] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus arrival time prediction using support vector machines," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 10, no. 4, pp. 151–158, 2006.
- [26] B. Yu, Z. Z. Yang, and B. Z. Yao, "A hybrid algorithm for vehicle routing problem with time windows," *Expert Systems with Applications*, vol. 38, no. 1, pp. 435–441, 2011.
- [27] B. Yu, Z. Yang, and J. Yao, "Genetic algorithm for bus frequency optimization," *Journal of Transportation Engineering*, vol. 136, no. 6, pp. 576–583, 2010.
- [28] B. Yu, H. B. Zhu, W. J. Cai, N. Ma, and B. Z. Yao, "Two-phase optimization approach to transit hub location—the case of Dalian," *Journal of Transport Geography*, vol. 33, pp. 62–71, 2013.
- [29] A. Selakov, D. Cvijetinović, L. Milović, S. Mellon, and D. Bekut, "Hybrid PSOSVM method for short-term load forecasting during periods with significant temperature variations in city of Burbank," *Applied Soft Computing*, vol. 16, pp. 80–88, 2014.
- [30] S. Sarafrazi and H. Nezamabadi-pour, "Facing the classification of binary problems with a GSA-SVM hybrid system," *Mathematical and Computer Modelling*, vol. 57, no. 1–2, pp. 270–278, 2013.
- [31] R. J. Fairbanks, S. E. Fahey, and W. W. Wierwille, "Research on vehicle-based driver status/performance monitoring," Seventh Semi-Annual Research Report, U.S. Department of Transportation, National Highway Traffic Safety Administration, 1994.
- [32] D. F. Dinges and R. Grace, "Perclous: a valid psychophysiological measure of alertness as assessed by psychomotor vigilance," 1988.
- [33] Q. Ji and X. J. Yang, "Real time visual cues extraction for monitoring driver vigilance," in *Computer Vision Systems*, B. Schiele and G. Sagerer, Eds., pp. 107–124, Springer, Berlin, Germany, 2001.
- [34] W. Rongben, G. Lie, T. Bingliang, and J. Lisheng, "Monitoring mouth movement for driver fatigue or distraction with one camera," in *Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems (ITSC '04)*, pp. 314–319, October 2004.
- [35] X. B. Zhang, B. Cheng, and R. J. Feng, "Real-time detection of driver drowsiness based on steering performance," *Journal of Tsinghua University*, vol. 50, pp. 1072–1076, 1081, 2010.
- [36] V. Tyagi, S. Kalyanaraman, and R. Krishnapuram, "Vehicular Traffic Density State Estimation Based on Cumulative Road Acoustics," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, pp. 1156–1166, 2012.
- [37] Q. Li, "Short-time traffic flow volume prediction based on support vector machine with time-dependent structure," in *Proceedings of the 2009 IEEE Instrumentation and Measurement Technology Conference (I2MTC '09)*, pp. 1730–1733, May 2009.
- [38] B. Yu, Z. Yang, and C. Cheng, "Optimizing the distribution of shopping centers with parallel genetic algorithm," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 2, pp. 215–223, 2007.

Research Article

Research of Obstacle Recognition Technology in Cross-Country Environment for Unmanned Ground Vehicle

Zhao Yibing, Xu Hongbin, Guo Lie, Li Linhui, and Zhang Mingheng

State Key Laboratory of Structural Analysis for Industrial Equipment, School of Automotive Engineering,
Dalian University of Technology, Dalian, Liaoning 116023, China

Correspondence should be addressed to Guo Lie; guo_lie@dlut.edu.cn

Received 29 April 2014; Revised 18 June 2014; Accepted 19 June 2014; Published 10 July 2014

Academic Editor: Rui Mu

Copyright © 2014 Zhao Yibing et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Being aimed at the obstacle recognition problem of unmanned ground vehicles in cross-country environment, this paper uses monocular vision sensor to realize the obstacle recognition of typical obstacles. Firstly, median filtering algorithm is applied during image preprocessing that can eliminate the noise. Secondly, image segmentation method based on the Fisher criterion function is used to segment the region of interest. Then, morphological method is used to process the segmented image, which is preparing for the subsequent analysis. The next step is to extract the color feature S , color feature a and edge feature “verticality” of image are extracted based on the HSI color space, the Lab color space, and two value images. Finally multifeature fusion algorithm based on Bayes classification theory is used for obstacle recognition. Test results show that the algorithm has good robustness and accuracy.

1. Introduction

Unmanned ground vehicle has been a research hotspot, and it has broad application prospect in military, civil, scientific research, and other fields [1]. The working environment of UGV has expanded to the more complex cross-country environment from the indoor environment. So in order to make the UGV safe and fast drive and finish the task in unknown environment, UGV must be able to quickly and accurately detect all kinds of obstacles and identify their categories in the environment, namely, having the ability of obstacle detection and recognition. That can provide the decision basis for how to plan a path, and then plan an optimal path according to a certain evaluation standard [2].

For the obstacle detection and recognition of UGV in cross-country environment, domestic and foreign researchers have done a lot of research. Mnaduchi et al. put forward an obstacle detection method, using the disparity map based on stereo vision, and test results proved that the method has better robust [3]. Mnaduchi et al. designed a remote obstacle detection system based on the ultrasonic radar, which has good detection range and precision, so

it can improve the environmental awareness of UGV [4]. Hu and Wu used laser rangefinder to realize the obstacle detection and recognition, which can quickly detect the static and dynamic obstacles in vehicle driving environment [5]. Huihai et al. combined machine vision and ultrasonic sensors to detect obstacles, and test results show that the obstacle detection algorithm is effective and practical [6]. Yanmin et al. proposed an obstacle detection algorithm based on stereo vision and laser radar, which can separate the high grass and other obstacles (such as trunks and stones) [7]. According to the optical flow model, Zhao et al. proposed an obstacle detection method based on optical flow field [8].

Kinds of sensors are used to obstacle detection and recognition, and all of them have both advantages and disadvantages. The principle and structure of visual sensor are similar to the human sensory organs, and machine vision sensor has the advantages of small volume, low cost, and convenient installation [9–11]. Compared with other sensors, because it belongs to the passive measurement, its concealment is better, and it also has the advantages of having wide detection range and containing rich information [12]. Based on the requirements of research topic and hardware,



FIGURE 1: The type of obstacles studied in this paper.

the main work of this paper is to research obstacle detection and recognition technology based on monocular vision in cross-country environment.

This paper mainly researches the detection and recognition technology of trunk and shrub. And shrubs are divided into high shrubs and dwarf shrubs. The types of obstacles researched in this paper are shown in Figure 1.

The main contents of this paper include the following aspects.

Chapter 1: Chapter 1 includes an introduction; it introduces the research status of unmanned vehicle obstacle detection and recognition technology, the meaning and main contents of this paper.

Chapter 2: obstacle detection is included in Chapter 2 that introduces an obstacle detection method based on monocular visual images, including image pre-processing, image segmentation, and morphological image processing.

Chapter 3: feature extraction is included in Chapter 3 that introduces the features extraction method based on HSI color space, Lab color space and binary image.

Chapter 4: Chapter 4 tackles identifying obstacles; it introduces an obstacle recognition method based on Bayes classification theory and calculates the parameters of Bayesian classifier by processing a large number of training samples.

Chapter 5: Chapter 5 includes test platform and the relevant conclusions.

2. Monocular Vision-Based Obstacle Detection Methods

2.1. Image Preprocessing Based on Median Filtering Method.

In the process of image acquisition, because of the difference of image acquisition environment, such as the light and shade degree of illumination and equipment performance, there are often noise and low contrast in the original image, which can reduce the image quality. So it is very necessary to image preprocessing before image segmentation, feature extraction, and pattern recognition [13, 14].

Median filtering belongs to nonlinear filtering method, and it is used to process the one-dimensional data in the earliest time. With the passage of time, it is gradually applied to process the two-dimensional image. Because of this, the template shape and size have great influence to the effect of [15]. When the template size increases, the effects of noise filtering will be better, but the details of the image will become increasingly blurred. So aiming at different quality and actual requirements, different shape and size of the template should be selected to realize the better noise filtering effect as far as possible and not lose the image edge details [16].

The main idea of median filtering method is to make the abrupt pixel values take the similar values with its neighborhood, so as to eliminate the isolated noise points



FIGURE 2: The comparison diagram before and after median filtering.

in the image. The specific steps of the method are shown as follows.

- (1) Put the template (3×1 , 1×3 or 3×3) to traverse the whole image and template center to coincide with the position of one pixel.
- (2) Sort all of the pixel values under in the template.
- (3) Find the middle values in this column.
- (4) Put the calculated middle values to the pixel corresponding to the center of template.

In this paper, median filtering method is used to eliminate the noise interference of original image, and the example is shown in Figure 2.

2.2. The Improved Image Segmentation Method Based on Fisher Criterion Function. Image segmentation is the most important and basic processing steps in the process of image processing, and it is also the key technology used in pattern recognition, image analysis, and understanding. Its main objective is to separate objects from a complex environment, and the effects of image segmentation have great influence on target recognition [17].

Threshold-based method is the simplest and most effective method in all the methods of image segmentation. Image segmentation method based on the criterion function is one of the very wide application methods, and its basic design idea is that based on some standard function, criterion function value is calculated in each threshold value, so the best threshold value is selected to which achieve the maximum threshold value or the minimum value. The classical image segmentation methods based on criterion function include minimum error criterion method, Otsu method, and maximum entropy method [18, 19]. Afterwards, Fisher criterion function is applied to the image segmentation method, and it is used in this paper. Its basic principle is as follows.

Assuming the distribution range of gray value $0 \sim G$, and there are only foreground and background in one image. $h(i)$, ($i = 0 \sim G$) is the normalized histogram. The priori

probability of objective and background are respectively $\theta(t)$ and $1 - \theta(t)$

$$\theta(t) = \sum_{i=0}^t h(i). \quad (1)$$

The mean value of objective and background is μ_1 and μ_2

$$\mu_1(t) = \sum_{i=0}^t \frac{h(i) i}{\theta(t)}, \quad (2)$$

$$\mu_2(t) = \sum_{i=t+1}^G \frac{h(i) i}{(1 - \theta(t))}.$$

The variance of objective and background is σ_1^2 and σ_2^2

$$\sigma_1^2 = \sum_{i=0}^t \frac{[i - \mu_1(t)]^2 h(i)}{\theta(t)}, \quad (3)$$

$$\sigma_2^2 = \sum_{i=t+1}^G \frac{[i - \mu_2(t)]^2 h(i)}{(1 - \theta(t))}.$$

The formula of Fisher criterion function is $J(t)$ which is expressed by

$$J(t) = \frac{|\theta(t) \mu_1(t) - [1 - \theta(t)] \mu_2(t)|^2}{\theta(t) \sigma_1^2(t) + [1 - \theta(t)] \sigma_2^2(t)}. \quad (4)$$

According to the above formula, the optimal threshold is

$$t^* = \underset{0 \leq t \leq G}{\text{Argmax}} [J(t)]. \quad (5)$$

When using image segmentation method based on Fisher criterion function, only the gray information is used, while the other color information is not used. So the improved method is that converting the gray image to other color spaces firstly, such as HSI and Lab color space, and then using Fisher criterion function method to process image. The process of this method is shown in Figure 3.

The effect of improved image segmentation method based on Fisher criterion function is shown in Figure 4.

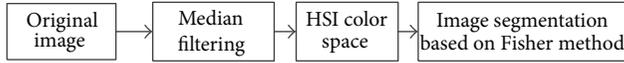


FIGURE 3: The flowchart of improved Fisher segmentation method.

In Figure 4, comparing (a) and (c) shows that the objective region has been segmented, and the interference is obviously reduced, which is convenient for subsequent processing. Otherwise the large interference will affect the shape of objective region and edge information, which have great influence on the final results of obstacle recognition.

2.3. Image Morphology. After the image segmentation, region of interest has been segmented, but at the same time, a part of the region of noninterest is also segmented, which may influence the subsequent image analysis. So this paper adopts the outlier elimination method, small areas elimination method, and the holes filling method to eliminate this interference.

2.3.1. Outlier Elimination Method. There will be some isolated spots in the segmented image, which have influence on the image recognition, though it is not much, so it should be eliminated. The main idea of outlier elimination method is that to process the neighborhood of a single pixel, such as 4-neighborhood and 8-neighborhood. Its specific steps are shown as follows.

- (1) Put the 3×3 template to traverse the whole image and read the values of all the pixels under template.
- (2) Calculate the average value of these gray values.
- (3) Compare the average value with the threshold.
- (4) If the average value is less than the threshold, this pixel value is set to 0 (background); otherwise, it is set to 1 (objective).

In this paper, the results of outlier elimination method are shown in Figure 5, and (b) shows the isolated spots that have been eliminated.

2.3.2. Small Areas Elimination Method. Although isolated points have been eliminated, there are still many interference areas that have not been eliminated, so only eliminating outliers are not enough. This paper uses small areas elimination method to eliminate the large interference areas. The main idea of this method is that according to the area size of the connected region the regions that their area is less than a certain threshold are eliminated. The specific steps of small areas elimination method are shown as follows.

- (1) Count all contained regions in image.
- (2) Calculate the area of connected regions.
- (3) Find the largest area in all the connected regions, and set it as the threshold.
- (4) Set all pixels value in the connected region that its area is smaller than the threshold as 0.

In this paper, the results of small areas elimination method are shown in Figure 6, and shows that the smaller connected regions exist in (a) have been eliminated, and only the region of interest exists in (b).

2.3.3. The Holes Filling Method. Because of the uneven illumination, the presence of noise, and other factors, there will be holes in the segmented image, which makes the region of interest discontinuous. So this paper uses the holes filling method to fill the holes, and its main idea is that to set all the pixels within the connected region as the specified gray value. The specific steps of this method are shown as follows.

- (1) Scan the whole image following a particular order (left to right, top to bottom).
- (2) In each scanning line, when a pixel value from 1 to 0, marking the point. And scanning continues until the pixel value from 0 to 1, marking this point too.
- (3) After scanning the whole image, set the pixels value between the two marker points according to the order from left to right as 1.

In this paper, the results of the holes filling method are shown in Figure 7, and (b) shows the holes in (a) have been effectively filled, and it cannot cause the loss of the shape feature of region of interest.

3. Monocular Vision-Based Feature Extractions of Obstacle Method

Monocular vision-based features include color, texture, and shape (edge, wide, and high). Color and texture features belong to the internal features, so the segmented image and the original image should be combined to achieve feature extraction. Shape feature belongs to the external image, so it can be directly extracted in the segmented image. This paper uses HSI and Lab color space to extract color feature and uses binary image to extract edge feature [20].

3.1. Color Feature of Obstacle Extraction. HSI color space that reflects the feeling of people to color was first proposed by Munseu and hue, saturation, and intensity are used to describe the characteristics of color [21]. In the HSI color space model, the concept of hue and saturation is independent, so it is suitable for the application to the process of image processing and analysis. Due to the fact that hue represents color wavelength, saturation represents the shades of color, and brightness represents the reflection effect, so hue and saturation are related to the image color information, while brightness is not related to image color information [22].

Lab color space is released by CIE in 1976, and it uses digital method to describe visual induction of human. Three components, L , a , and b , are used to describe the characteristics of color. The L component represents luminosity, and it does not contain color information. The other components a and b represent the color information [23]. Lab color space has the advantages of wide color gamut, so long as the human visual perception of color can be shown through

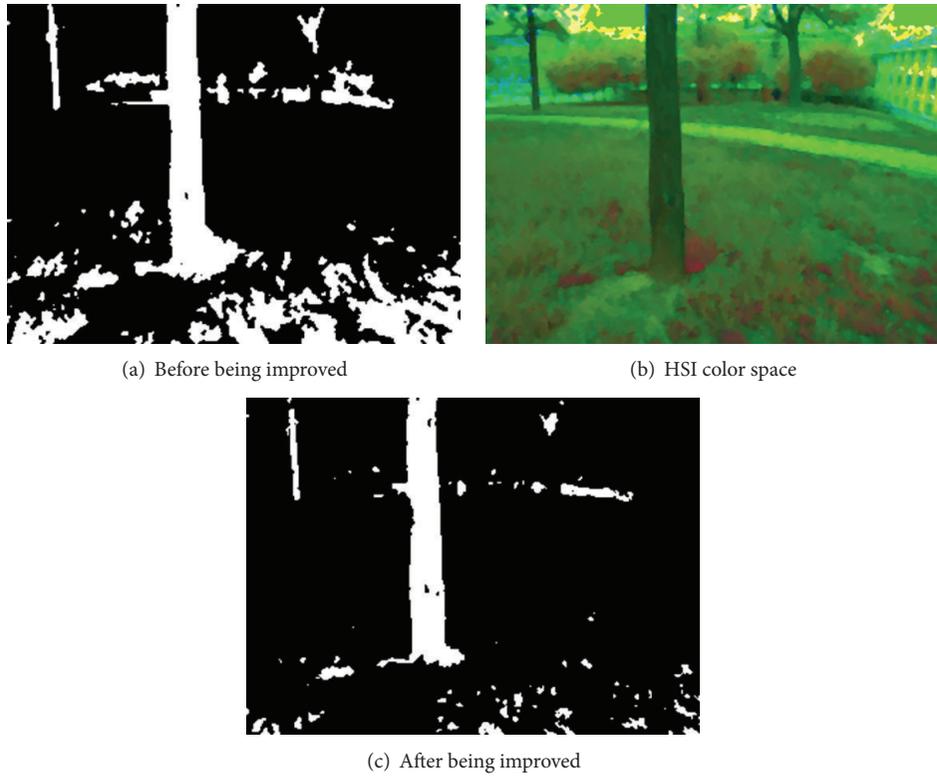


FIGURE 4: The results of improved Fisher segmentation method.

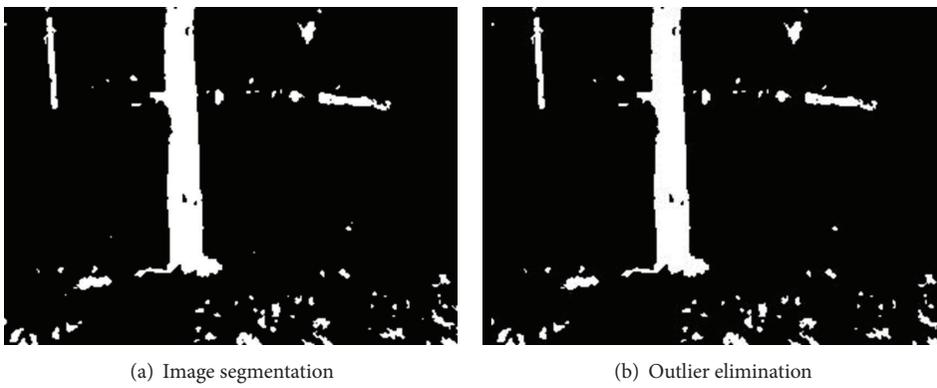


FIGURE 5: The results of outlier elimination.

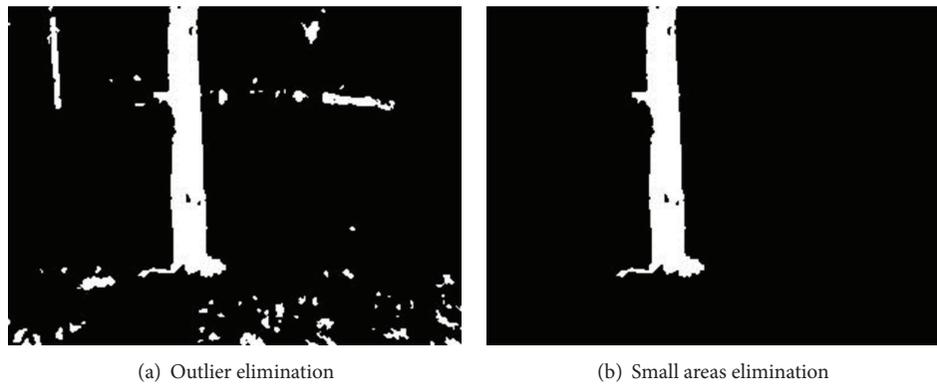


FIGURE 6: The results of small areas elimination.

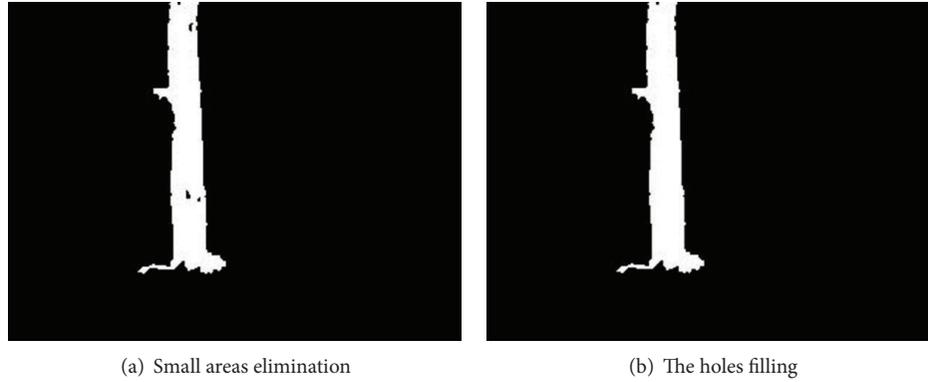


FIGURE 7: The results of the holes filling.

the Lab color space model. In addition, it also makes up for the shortcomings of color uneven distribution of RGB color space model [24].

The RGB images were converted to HSI and Lab color space, and then this paper count lots of S -values, as shown in Figure 8(a), and a -values, as shown in Figure 8(b), of trunk, high shrubs, and dwarf shrubs. The horizontal axis represents the number of all kinds of obstacles, and the vertical axis represents the statistical characteristic value. The red dot (\cdot) represents the statistical values of trunk, and the green plus sign ($+$) represents the statistical values of high shrubs. The blue asterisk ($*$) represents the statistical values of dwarf shrubs. As can be seen from Figures 8(a) and 8(b), S -feature can be used to separate trunk and the other obstacles, and a -feature can be used to separate high shrubs and the other obstacles.

3.2. Edge Feature of Obstacle Extraction. There are only 0 and 1 in the pixel values of binary image, and the segmented image has been binary image. In order to extract the edge information of image this paper uses edge detection method and boundary representation method to further image processing.

In the process of edge detection, the ability of antinoise and edge localization is contradictory, that is to say, by enhancing one the other one will be reduced. But the main idea of the Canny operator is to search of the best solution between the ability of antinoise and edge localization, so that both the ability of antinoise and edge detection are strong as far as possible.

In this paper, the results of edge detection method based on Canny operator is shown in Figure 9, and Figure 9(b) shows the edge has been fully extracted, which is continuous single pixel edge and has much influence on boundary representation. So the results show that this method is very effective.

Chain code was proposed by Freeman in 1961, and it is a kind of representation of the boundary point. Its main idea is using a series of connected line segment that has a specific length and direction to represent boundary [25]. Due to that, only the starting point needs to be represented by coordinate, while the other points can be represented

by direction. Storing a direction required less space than a coordinate, so it can greatly reduce the number of data, when using chain code to represent boundary instead of coordinate. Now, the commonly used chain codes include 4-direction chain code, 6-direction chain code, 8-direction chain code, and 16-direction chain code, as shown in Figure 10.

This paper uses 16-direction code chain and counts lots of “verticality” values of trunk, high shrubs, and dwarf shrubs as shown in Figure 11. Verticality is defined as the ratio of the numbers of vertical direction and the total numbers of all directions, that is to say, it is the proportion of vertical direction. The horizontal axis represents the number of all kinds of obstacles, and the vertical axis represents the statistical characteristic value. The red dot (\cdot) represents the statistical values of trunk and the green plus sign ($+$) represents the statistical values of high shrubs. The blue asterisk ($*$) represents the statistical values of dwarf shrubs. As can be seen from Figure 11, the “verticality” feature can be used to separate trunk and dwarf shrubs.

4. Obstacle Recognition Methods Based on Bayes Classification Theory

The flowchart of statistical pattern recognition system is shown in Figure 12, and it can be seen that the recognition system includes two parts: the training stage and the recognition stage. The training stage is used to analyze known samples, so as to formulate the classification criterion, which is the basis of obstacle recognition of unknown samples. The recognition stage is used to achieve the classification and recognition of unknown samples.

4.1. Bayes Classifier

4.1.1. The Theory of Bayes Classifier. The main ideas of Bayes classifier is to make the classification error rates of the results minimum in the given condition, namely, the error rate is set as the basis of classification and recognition [26]. There are M objects, $\omega_i, i = 1, 2, \dots, M$, so for the X -feature of the samples, Bayes formula is as follows:

$$P(\omega_i | X) = \frac{P(X | \omega_i) P(\omega_i)}{P(X)}, \quad (6)$$

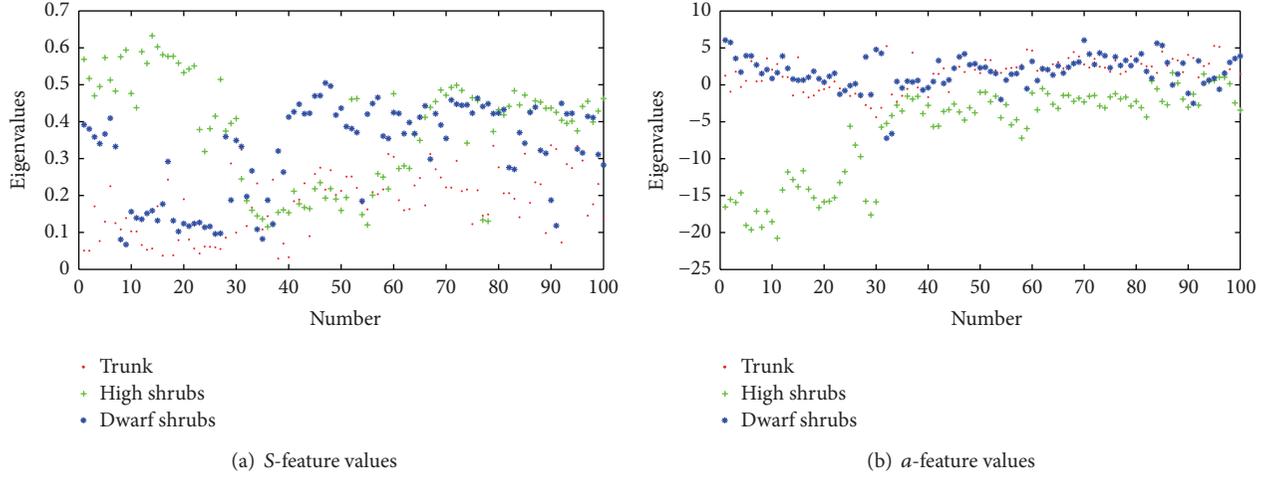


FIGURE 8: The statistical diagram of color feature.

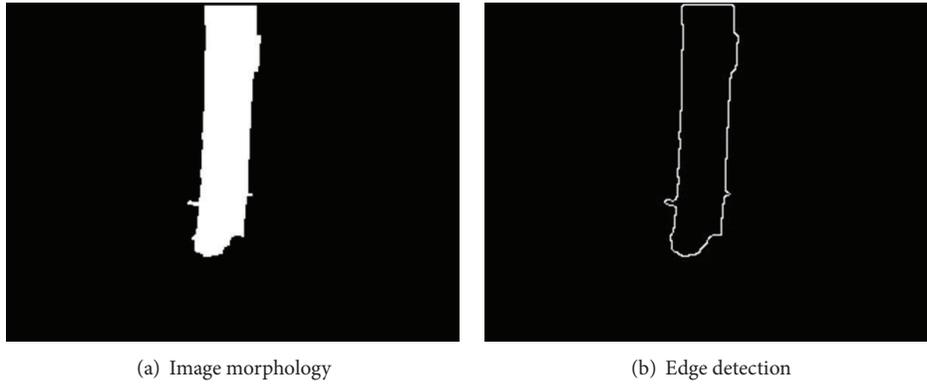


FIGURE 9: The results of edge detection based on Canny operator.

according to the total probability formula:

$$P(X) = \sum_{j=1}^M P(X | \omega_j) P(\omega_j). \quad (7)$$

In formula (6), $P(\omega_i)$ represents the prior probability, which is the possibility of all events occurring without considering any conditions. $P(X | \omega_i)$ represents the conditional probability density function. According to the former researches, the statistical data of many problems mostly belong to normal distribution, so the density function of normal distribution is selected as the calculation form of conditional probability density function. The unilabiate normal distribution probability density function is as follows:

$$P(x) = \frac{1}{\sqrt{2\pi}\delta} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\delta}\right)^2\right]. \quad (8)$$

There, μ represents mathematical expectation, and its formula is shown as follows:

$$\mu = E(x) = \int_{-\infty}^{\infty} xP(x) dx. \quad (9)$$

δ^2 represents variance, and its formula is shown as follows:

$$\delta^2 = E[(x - \mu)^2] = \int_{-\infty}^{\infty} (x - \mu)^2 P(x) dx. \quad (10)$$

According to formulas (8), (9), and (10), in order to determine the specific expression of a conditional probability density function, expectation and variance should be calculated, which can be achieved by studying a large number of samples.

In formula (6), $P(\omega_i | X)$ represents the posterior probability, which is the criterion for the final recognition. The flowchart of Bayes classifier based on minimum error rate is shown in Figure 13.

Figure 13 shows that the recognition process of Bayes classifier can be briefly described as follows. Calculate posterior probability $P(\omega_1 | X), P(\omega_2 | X), \dots, P(\omega_M | X)$, and then calculate the maximum value as follows:

$$P(\omega_i | X) = \max_{1 \leq j \leq M} \{P(\omega_j | X)\}, \quad i = 1, 2, \dots, M. \quad (11)$$

If class i has the maximum posterior probability, so the tested sample belongs to the class i .

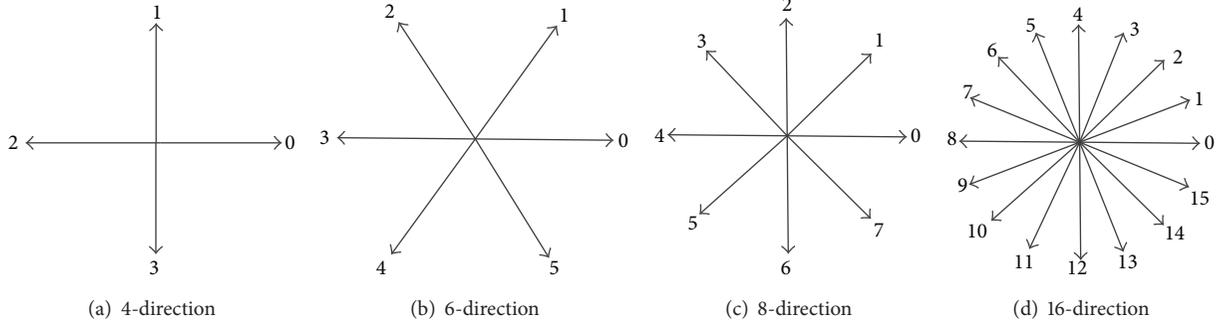


FIGURE 10: The common forms of chain code.

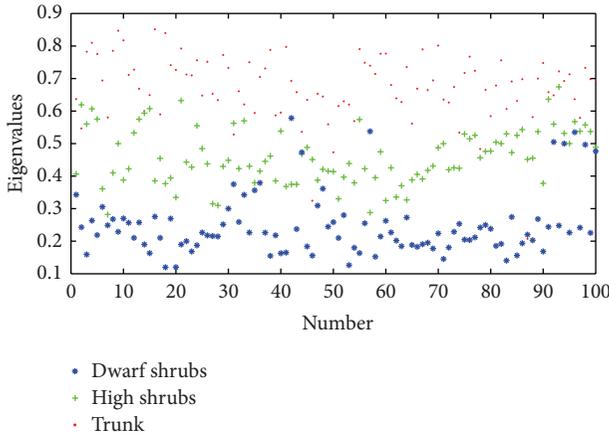


FIGURE 11: The statistical diagram of verticality.

4.1.2. *The Design of Bayes Classifier.* The obstacles researched in this paper are trunk, high shrubs, and dwarf shrubs, which are set to ω_1 , ω_2 , and ω_3 . And their priori probabilities are set to $P(\omega_1)$, $P(\omega_2)$, and $P(\omega_3)$. The prior probabilities are shown as follows:

$$P(\omega_1) = P(\omega_2) = P(\omega_3) = \frac{1}{3}. \quad (12)$$

For the same sample, this paper extracted three different features. X_1 represents saturation S , X_2 represents color features a , and X_3 represents “verticality” feature based on 16-direction chain code. For each feature value, this paper need calculate three conditional probability density functions. According to the features, MATLAB is used to estimate the parameters of conditional probability density function. If all the features belong to normal distribution, the conditional probability density functions are shown as follows:

$$P(X_j | \omega_i) = \frac{1}{\sqrt{2\pi}\delta} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\delta}\right)^2\right], \quad i, j = 1, 2, 3. \quad (13)$$

In order to see the distribution of data more intuitively, the histogram is the most commonly used method. Its main idea is that to divide the data into some interval, which has the same interval between them. The histogram is defined as

the rectangle. Figure 14(a) shows the histogram of S -feature of trunk, and Figure 14(b) shows the fitted normal distribution probability density function curve based on S -feature data of trunk samples.

The intuitionistic method that testing whether the data belong to normal distribution is shown in Figure 15. The red line is transformed from the normal distribution, and the blue points present the detected data. The linear approximation of data accord with normal distribution. As it can be seen from the figure, the data points are mostly falling in line, so The preliminary conclusion is that the data fit a normal distribution.

The specific hypothesis testing results for the data of trunk samples based on Lilliefors are shown in Table 1. The statistical value is 0.0596, which is less than the critical value that is 0.0947, so the original hypothesis is accepted, the conclusion is that the data fit a normal distribution. According to the expectation and variance, the S -feature values of trunk are in accord with normal distribution $N(0.1960, 0.0753^2)$, so the conditional probability density functions are as follows:

$$P(X_1 | \omega_1) = \frac{1}{\sqrt{2\pi} * 0.0753} \exp\left[-\frac{1}{2}\left(\frac{x - 0.1960}{0.0753}\right)^2\right]. \quad (14)$$

In addition, the 95% confidence interval of expectation is [0.1800, 0.2120], and the 95% confidence interval of variance is [0.0656, 0.0885].

Similarly, the S -feature values of high shrubs are in accord with normal distribution $N(0.4591, 0.0799^2)$, so the conditional probability density functions are as follows:

$$P(X_1 | \omega_2) = \frac{1}{\sqrt{2\pi} * 0.0799} \exp\left[-\frac{1}{2}\left(\frac{x - 0.4591}{0.0799}\right)^2\right]. \quad (15)$$

The S -feature values of dwarf shrubs are in accord with normal distribution $N(0.3768, 0.0579^2)$, so the conditional probability density functions are as follows:

$$P(X_1 | \omega_3) = \frac{1}{\sqrt{2\pi} * 0.0579} \exp\left[-\frac{1}{2}\left(\frac{x - 0.3768}{0.0579}\right)^2\right]. \quad (16)$$

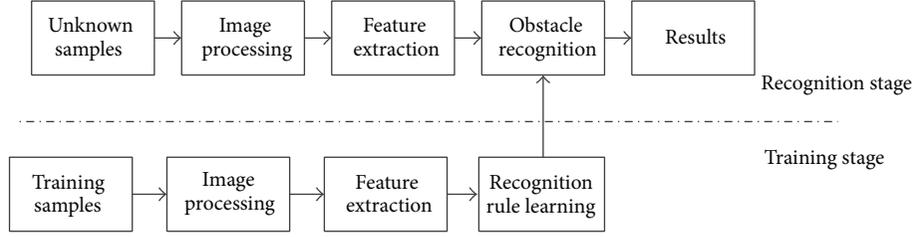


FIGURE 12: The flowchart of statistical pattern recognition system.

TABLE 1: The results of hypothesis testing.

Logical value	Probability	Statistical value	Critical value	Expectation	Variance
0	0.5000	0.0596	0.0947	0.1960	0.0753

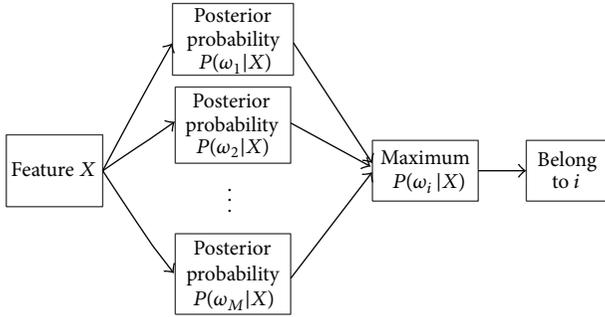


FIGURE 13: The flowchart of Bayes classifier.

The a -feature values of trunk are in accord with normal distribution $N(2.4879, 1.2802^2)$, so the conditional probability density functions are as follows:

$$P(X_2 | \omega_1) = \frac{1}{\sqrt{2\pi} * 1.2802} \exp \left[-\frac{1}{2} \left(\frac{x - 2.4879}{1.2802} \right)^2 \right]. \quad (17)$$

The a -feature values of high shrubs are in accord with normal distribution $N(-2.4449, 1.8926^2)$, so the conditional probability density functions are as follows:

$$P(X_2 | \omega_2) = \frac{1}{\sqrt{2\pi} * 1.8926} \exp \left[-\frac{1}{2} \left(\frac{x - (-2.4449)}{1.8926} \right)^2 \right]. \quad (18)$$

The a -feature values of dwarf shrubs are in accord with normal distribution $N(1.5453, 1.6136^2)$, so the conditional probability density functions are as follows:

$$P(X_2 | \omega_3) = \frac{1}{\sqrt{2\pi} * 1.6136} \exp \left[-\frac{1}{2} \left(\frac{x - 1.5453}{1.6136} \right)^2 \right]. \quad (19)$$

TABLE 2: The range of color feature for three kinds of obstacles.

Obstacle	S-feature	a -feature	Verticality
Trunk	[0.1800, 0.2120]	[2.1973, 2.7785]	[0.6658, 0.6988]
High shrubs	[0.4402, 0.4780]	[-2.8929, -1.9970]	[0.4552, 0.4936]
Dwarf shrubs	[0.3642, 0.3893]	[1.1993, 1.8912]	[0.2102, 0.2320]

The “verticality” values of trunk are in accord with normal distribution $N(0.6826, 0.0822^2)$, so the conditional probability density functions are as follows:

$$P(X_3 | \omega_1) = \frac{1}{\sqrt{2\pi} * 0.0822} \exp \left[-\frac{1}{2} \left(\frac{x - 0.6823}{0.0822} \right)^2 \right]. \quad (20)$$

The “verticality” values of high shrubs are in accord with normal distribution $N(0.4744, 0.0861^2)$, so the conditional probability density functions are as follows:

$$P(X_3 | \omega_2) = \frac{1}{\sqrt{2\pi} * 0.0861} \exp \left[-\frac{1}{2} \left(\frac{x - 0.4744}{0.0861} \right)^2 \right]. \quad (21)$$

The “verticality” values of dwarf shrubs are in accord with normal distribution $N(0.2212, 0.0514^2)$, so the conditional probability density functions are as follows:

$$P(X_3 | \omega_3) = \frac{1}{\sqrt{2\pi} * 0.0514} \exp \left[-\frac{1}{2} \left(\frac{x - 0.2212}{0.0514} \right)^2 \right]. \quad (22)$$

The 95% confidence intervals of color and edge features for three kinds of obstacles are shown in Table 2.

4.2. Multifeature Fusion Algorithm Based on the Bayes Classifier. According to the obtained prior probability and conditional probability density functions, combined with the Bayes formula, the posterior probability under the single characteristic value can be calculated as follows:

$$P(\omega_i | X_j) = \frac{P(X_j | \omega_i) P(\omega_i)}{P(X_j)}, \quad i, j = 1, 2, 3, \quad (23)$$

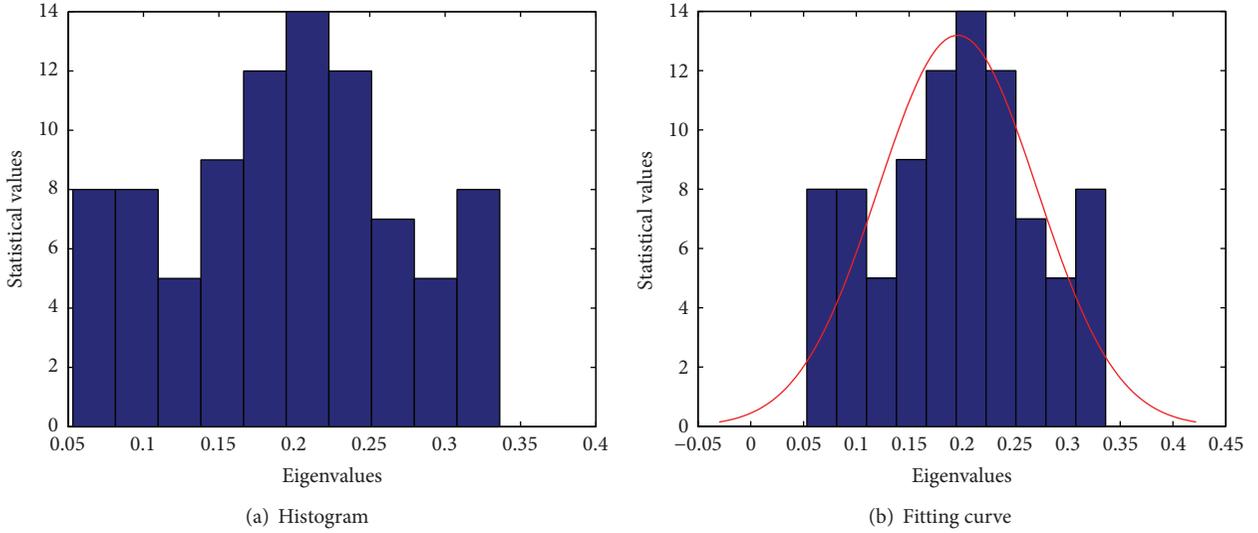


FIGURE 14: The fitting curve of normal distribution.

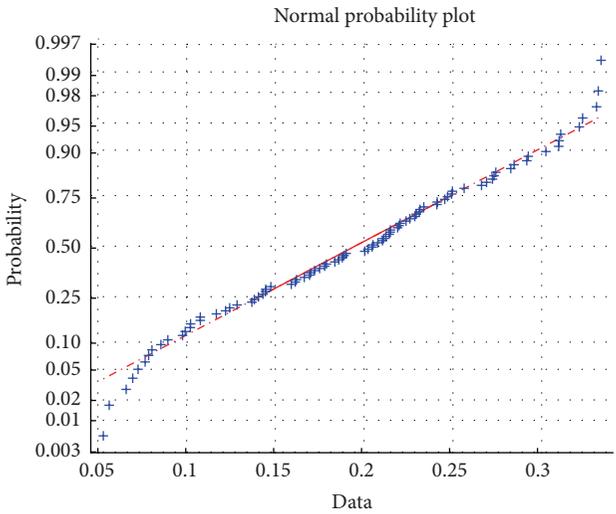


FIGURE 15: The normal distribution test of data.

according to the total probability formula:

$$\begin{aligned}
 P(X_j) &= \sum_{i=1}^3 P(X_j | \omega_i) P(\omega_i) \\
 &= P(X_j | \omega_1) P(\omega_1) + P(X_j | \omega_2) P(\omega_2) \\
 &\quad + P(X_j | \omega_3) P(\omega_3),
 \end{aligned} \quad (24)$$

where, ω_1 , ω_2 , and ω_3 , respectively, present trunk, high shrubs, and dwarf shrubs and X_1 , X_2 , and X_3 , respectively, present S-feature, a-feature, and "verticality."

Then, put the prior probability in formula (12) and conditional probability density function in formula (14)~(22) into formula (23), the posteriori probability can be calculated for each feature. So tested samples can be classified based



FIGURE 16: The test platform of UGV.

on these values; namely, it achieves the task of obstacle recognition.

The above method is focused on single feature, and then this paper presents obstacle recognition of multifeature fusion based on Bayes classification theory. Its basic is Bayes classification theory and then operating the obstacle recognition combined with multifeature fusion.

There are n features and m class; then, the steps of this method are shown as follows.

- (1) According to the actual situation, get the prior probability of each class.
- (2) Calculate the conditional probability density function of the multifeature fusion. If they are independent of each other between n features, namely, they do not interfere with each other, so the conditional probability function is shown as follows:

$$\begin{aligned}
 &P(X_1 \wedge X_2 \cdots \wedge X_n | \omega_i) \\
 &= P(X_1 | \omega_i) * P(X_2 | \omega_i) \cdots * P(X_n | \omega_i), \quad (25) \\
 &\quad i = 1, 2, \dots, m.
 \end{aligned}$$

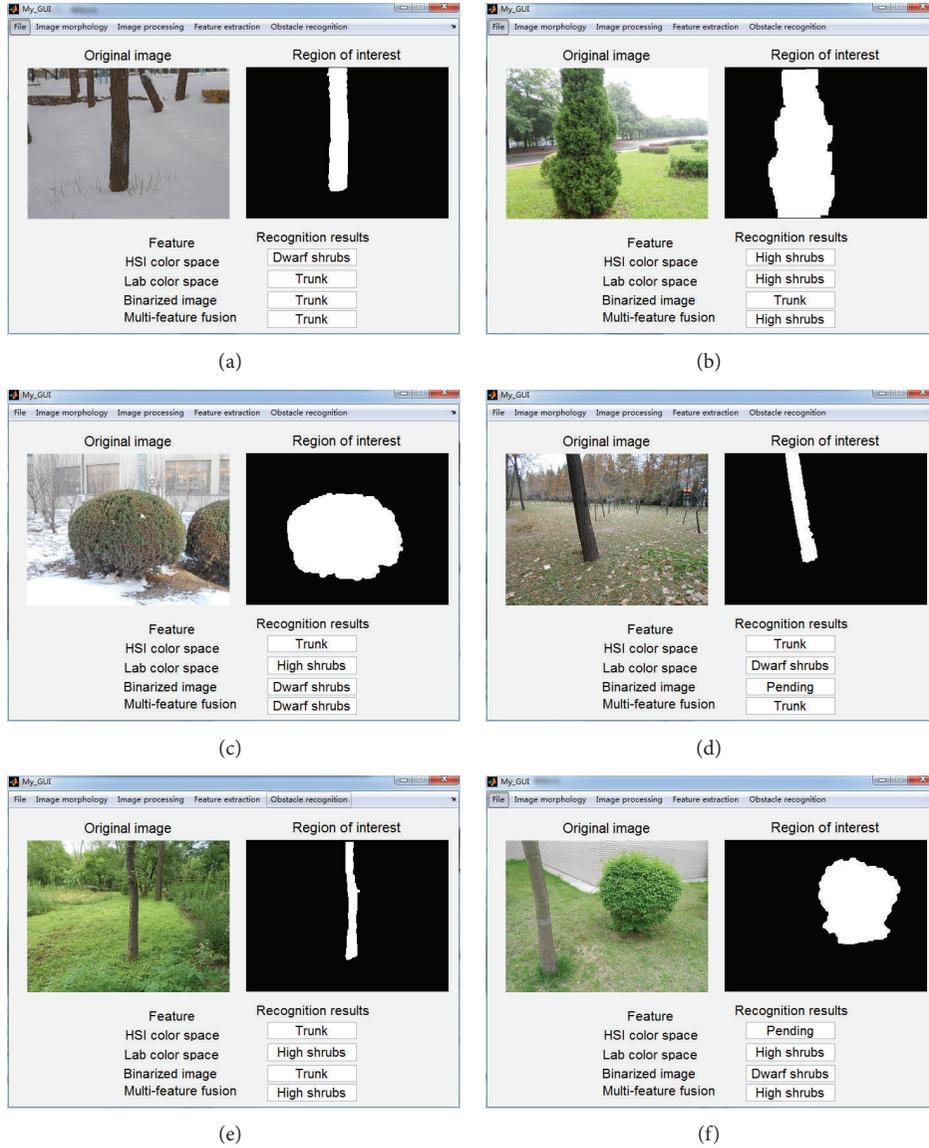


FIGURE 17: The results of obstacle recognition.

(3) The total probability formula is employed to calculate the total probability:

$$\begin{aligned}
 &P(X_1 \wedge X_2 \cdots \wedge X_n) \\
 &= \sum_{i=1}^m P(X_1 \wedge X_2 \cdots \wedge X_n | \omega_i) P(\omega_i) \\
 &= P(X_1 \wedge X_2 \cdots \wedge X_n | \omega_1) P(\omega_1) \\
 &\quad + P(X_1 \wedge X_2 \cdots \wedge X_n | \omega_2) P(\omega_2) \\
 &\quad + P(X_1 \wedge X_2 \cdots \wedge X_n | \omega_m) P(\omega_m).
 \end{aligned} \tag{26}$$

(4) Bayes formula is employed to calculate the posterior probability:

$$\begin{aligned}
 &P(\omega_i | X_1 \wedge X_2 \cdots \wedge X_n) \\
 &= \frac{P(X_1 \wedge X_2 \cdots \wedge X_n | \omega_i) P(\omega_i)}{P(X_1 \wedge X_2 \cdots \wedge X_n)}, \quad i = 1, 2, \dots, m.
 \end{aligned} \tag{27}$$

(5) Classifying according to the classification rules, classification rules are shown as follows:

$$\begin{aligned}
 &P(\omega_i | X_1 \wedge X_2 \cdots \wedge X_n) \\
 &= \max_{1 \leq j \leq m} \{P(\omega_j | X_1 \wedge X_2 \cdots \wedge X_n)\}, \quad i = 1, 2, \dots, m.
 \end{aligned} \tag{28}$$

Combined with the research content of this paper, which mainly includes three features and three kinds of obstacles, the posterior probability of multifeature fusion based on Bayes classification theory is as follows:

$$P(\omega_i | X_1 \wedge X_2 \wedge X_3) = \frac{P(X_1 \wedge X_2 \wedge X_3 | \omega_i) P(\omega_i)}{P(X_1 \wedge X_2 \wedge X_3)}, \quad i = 1, 2, 3, \quad (29)$$

where conditional probability density function is

$$P(X_1 \wedge X_2 \wedge X_3 | \omega_i) = P(X_1 | \omega_i) * P(X_2 | \omega_i) * P(X_3 | \omega_i), \quad i = 1, 2, 3. \quad (30)$$

The total probability formula is as followed:

$$P(X_1 \wedge X_2 \wedge X_3) = \sum_{i=1}^3 P(X_1 \wedge X_2 \wedge X_3 | \omega_i) P(\omega_i) = P(X_1 \wedge X_2 \wedge X_3 | \omega_1) P(\omega_1) + P(X_1 \wedge X_2 \wedge X_3 | \omega_2) P(\omega_2) + P(X_1 \wedge X_2 \wedge X_3 | \omega_3) P(\omega_3). \quad (31)$$

Finally, put the prior probability in formula (12) and conditional probability density function in formula (14)~(22) into formula (29), the posteriori probability combined three features can be calculated. So tested samples can be classified based on these values; namely, it achieves the task of obstacle recognition.

5. Test and Result Analysis

In this paper, the UGV platform of Dalian University of Technology is used to image acquisition. Then VC++ software is used to image processing and segment the region of interest. Finally, MATLAB software is used to feature extraction and obstacle recognition.

5.1. Test Platform. The UGV platform of DLUT is shown in Figure 16. The perception system of the UGV includes two AVT F-033B/C color cameras, one real-time laser radar (SICK-221), one American UNIQ USS-301 infrared camera, and one ADVANTECH IPC-610H industrial PC. Both the color camera and infrared camera are mounted on the top of the UGV's platform with shims that provided a 10° down tilt. Laser radar is mounted to the bumper with horizontal forward-looking scanning field of view [27].

5.2. Test Results. This paper collected 300 pieces of obstacle images, including 100 pieces of trunk images, 100 pieces of high shrubs images, and 100 pieces of dwarf shrubs images. For any one obstacle image, its probability belonging to the tree, high shrubs, and dwarf shrubs can be calculated, respectively, and when a probability value satisfies certain

TABLE 3: The accuracy of obstacle recognition.

Feature	S (X_1)	a (X_2)	Verticality (X_3)	Fusion ($X_1 \wedge X_2 \wedge X_3$)
Accuracy	59.33%	57%	87%	92.33%

conditions, the recognition results can be determined; otherwise; the recognition result is pending. The conditions are shown as follows.

- (1) It is the maximum posteriori probability.
- (2) The difference between it and other posterior probability values is greater than a certain threshold (the threshold is setting for 5%).

A part of recognition results is shown in Figure 17.

5.3. Result Analysis. This paper tested 300 images and obtained the recognition results. Then the accuracy of single feature and multifeature fusion can be obtained, as shown in Table 3.

As can be seen from the table, when using a single feature to recognize obstacle, the correct rate is relatively low, while the correct rate of multifeature fusion reached more than 90% and it is the highest in the four methods. Therefore, the test results show that the effects of using multifeature fusion to recognize obstacle is better than using a single feature, and the recognition accuracy is significantly improved, so it verifies the feasibility and validity of the method.

6. Conclusion

According to the type of obstacle that UGV may be encountered with in the off-road environment, such as trunk, high shrubs, and dwarf shrubs, as well as a large number of related research achievements, this paper presented a monocular-vision obstacle recognition method based on Bayes classification theory. The main content is feature extraction methods based on HIS color space, Lab color space, and binary image and multifeature fusion obstacle recognition method based on Bayes classification theory. Test results show the proposed method has better robust and accuracy. In addition, there are some shortcomings in this paper. For example, the type of researched obstacles is not enough, and the real-time performance of obstacle recognition algorithm is not discussed, so the subsequent work will consider rocks, pits, water, and other obstacles, which can improve the obstacle recognition system of UGV in the off-road environment, and the real-time performance of this method.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is sponsored by the Specialized Research Fund for the Doctoral Program of Higher Education (20110041120024), National Natural Science Foundation Project (51205038), and Fundamental Research Funds for the Central Universities DUT13JS14 and DUT13JS02. Finally, the authors are grateful for all the friends cooperating with us during the research and the events, and the journalists who covered this event.

References

- [1] W. Wei, *Obstacle Detection and Road Segmentation for Robots Based on Information Fusion*, Zhejiang University, Zhejiang, China, 2010.
- [2] Q. Weigao and X. Xuejin, "Development status and direction of driverless vehicle," *Shanghai Automotive*, vol. 7, pp. 40–43, 2007.
- [3] R. Manduchi, A. Castano, A. Talukder, and L. Matthies, "Obstacle detection and terrain classification for autonomous off-road navigation," *Autonomous Robots*, vol. 18, no. 1, pp. 81–102, 2005.
- [4] R. Mnaduchi, A. Castano, A. Talukder et al., "Real-time moving obstacle detection using optical flow models," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 466–471, Tokyo, Japan, 2006.
- [5] T. Hu and T. Wu, "A fast and robust obstacle detection algorithm for off-road autonomous mobile robots," *Robot*, vol. 33, no. 3, pp. 287–298, 2011.
- [6] C. Huihai, L. Yan, and Z. Xin, "A large range sonar obstacle detection system for cross-country autonomous vehicle," *Electronic Design Engineering*, vol. 19, no. 16, pp. 64–67, 2011.
- [7] L. Yanmin, Z. Qidan, Z. Xunyu et al., "Study on obstacle detection based on laser range finder," *Computer Engineering and Design*, vol. 33, no. 2, pp. 718–723, 2012.
- [8] X. Zhao, P. Liu, M. Zhang, L. Yang, and J. Shi, "A fast obstacle detection algorithm for mobile robots and its application," *Robot*, vol. 33, no. 2, pp. 198–214, 2011.
- [9] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research Part C*, vol. 43, pp. 233–248, 2014.
- [10] B. Yao, P. Hu, M. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for the periodic vehicle routing problem," *Simulation*, vol. 89, no. 6, pp. 762–770, 2013.
- [11] B. Z. Yao, C. Y. Yang, J. B. Yao, and J. Sun, "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.
- [12] İ. K. İyidir, F. B. Tek, and D. Kırcaı, "Adaptive visual obstacle detection for mobile robots using monocular camera and ultrasonic sensor," in *Proceedings of the 12th International Conference on Computer Vision (ECCV '12)*, vol. 2, pp. 526–535, Florence, Italy, October 2012.
- [13] B. Z. Yao, C. Y. Yang, J. B. Yao, J. J. Hu, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6-7, pp. 2127–2131, 2011.
- [14] B. Z. Yao, J. B. Yao, M. H. Zhang, and L. Yu, "Improved support vector machine regression in multi-step-ahead prediction for tunnel surrounding rock displacement," *Scientia Iranica*, 2013.
- [15] B. Yu, Z. Yang, K. Chen, and B. Yu, "Hybrid model for prediction of bus arrival times at next station," *Journal of Advanced Transportation*, vol. 44, no. 3, pp. 193–204, 2010.
- [16] Y. Bin, Y. Zhongzhen, and Y. Baozhen, "Bus arrival time prediction using support vector machines," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 10, no. 4, pp. 151–158, 2006.
- [17] B. Yu, Z. Z. Yang, and B. Z. Yao, "A hybrid algorithm for vehicle routing problem with time windows," *Expert Systems with Applications*, vol. 38, no. 1, pp. 435–441, 2011.
- [18] B. Yu, Z. Z. Yang, and J. B. Yao, "Genetic algorithm for bus frequency optimization," *Journal of Transportation Engineering*, vol. 136, no. 6, pp. 576–583, 2010.
- [19] B. Yu, H. B. Zhu, W. J. Cai, N. Ma, and B. Z. Yao, "Two-phase optimization approach to transit hub location—the case of Dalian," *Journal of Transport Geography*, vol. 33, pp. 33–62, 2013.
- [20] X. Pang, Z. Min, and J. Kan, "Color image segmentation based on HSI and LAB color space," *Journal of Guangxi University: Natural Science Edition*, vol. 36, no. 6, pp. 976–980, 2011.
- [21] H. Shaojia, L. Ziyang, and S. Jianqing, "Obstacle detection of indoor robots based on monocular vision," *Journal of Computer Applications*, vol. 32, no. 9, pp. 2556–2559, 2012.
- [22] P. Fei and W. Henghua, "Study on obstacle distance detection based on monocular humanoid robot," *Computer Systems & Applications*, vol. 22, no. 8, pp. 88–91, 2013.
- [23] B. Yu, L. Guo, X. Qian, T. Zhao, and G. Cheng, "An effective denoising algorithm in CIE-lab color space using hybrid filtering," *Journal of Northwestern Polytechnical University*, vol. 30, no. 6, pp. 941–945, 2012.
- [24] C. Changtao, Q. Guoqing, and Y. Ping, "Application of Lab spaces color segmentation in fast vehicle license plate location," *Application Research of Computers*, vol. 27, no. 8, pp. 3191–3193, 2010.
- [25] D. Jingyi and Y. Jiao, "Using improved Freeman chain code to RMB denomination identification," *Computer Engineering and Design*, vol. 33, no. 12, pp. 4643–4646, 2012.
- [26] H. Song, D. He, and X. Xin, "Unstructured road detection and obstacle recognition algorithm based on machine vision," *Transactions of the Chinese Society of Agricultural Engineering*, vol. 27, no. 6, pp. 225–230, 2011.
- [27] Y. Zhao, J. Li, L. Li, M. Zhang, and L. Guo, "Environmental perception and sensor data fusion for unmanned ground vehicle," *Mathematical Problems in Engineering*, vol. 2013, Article ID 903951, 12 pages, 2013.

Research Article

Road Network Vulnerability Analysis Based on Improved Ant Colony Algorithm

Yunpeng Wang,¹ Yuqin Feng,^{1,2} Wenxiang Li,³ William Case Fulcher,⁴ and Li Zhang⁴

¹ School of Transportation Science and Engineering, Beihang University, Beijing 100191, China

² School of Automobile Engineering, Heilongjiang Institute of Technology, Harbin 150050, China

³ Key Laboratory of Road Traffic Engineering of the Ministry of Education, Tongji University, Shanghai 201804, China

⁴ Department of Civil and Environmental Engineering, Mississippi State University, Starkville, MS 39759, USA

Correspondence should be addressed to Yuqin Feng; promise1973@163.com

Received 27 March 2014; Accepted 7 May 2014; Published 22 June 2014

Academic Editor: Baozhen Yao

Copyright © 2014 Yunpeng Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present an improved ant colony algorithm-based approach to assess the vulnerability of a road network and identify the critical infrastructures. This approach improves computational efficiency and allows for its applications in large-scale road networks. This research involves defining the vulnerability conception, modeling the traffic utility index and the vulnerability of the road network, and identifying the critical infrastructures of the road network. We apply the approach to a simple test road network and a real road network to verify the methodology. The results show that vulnerability is directly related to traffic demand and increases significantly when the demand approaches capacity. The proposed approach reduces the computational burden and may be applied in large-scale road network analysis. It can be used as a decision-supporting tool for identifying critical infrastructures in transportation planning and management.

1. Introduction

As a primary infrastructure of the modern city, a robust transportation network is one of the preconditions of a flourishing economy and a high standard of living-class life. However, there are many events emergencies, such as traffic congestion, traffic accidents, road maintenance, bad weather, and terrorist activities, that can have a tremendous impact on the operational performance of the road network and can make some road segments impassible. These events will cause the traffic flow to be redistributed, which could cause congestion in other road segments to occur. In these cases, the vulnerabilities of the road network reveal themselves constantly [1–3]. It is uneconomical to solve these problems merely by constructing the road infrastructure to increase the capacity of the entire road network [4]. Instead, we should explore the characteristics and laws of the exposed problems, identify the vulnerable segments of the road network, and understand the operational performance variation of these segments in adverse conditions and the consequences

are produced. Thus it is necessary and important to study the vulnerability of road networks.

1.1. State of the Art. The concept of vulnerability was first proposed by Timmerman in 1981 [5]. The research was focused in the field of geology. Since the 1990s, a multitude researches on vulnerability have emerged, where vulnerability was used in the field of disaster management, public safety, economics, sustainability, science, and so on. Vulnerability may be the most fundamental and essential problem in our daily life especially in emergency evacuation situations [6, 7]. Berdica defined the conception as the characteristic of the degenerating transportation system's accessibility in different cases and the characteristic that can be susceptible to unusual events [8]. Taylor and D'Este believed the vulnerability, reliability, and risk of the transportation network were the closely related concepts [9]. Murray-Tuite and Mahmassani established a bilevel programming model to excavate the vulnerable segments of the transportation network, in which

traffic managers supervise the traffic to achieve optimization in the underlying programming model while the damage factors, such as traffic accidents, and bad weather, maximize the degradation of the transportation network [10]. Husdal considered the vulnerability of the transportation network as a prime, and the impact of vulnerability and reliability should be included in the cost-benefit analysis [11]. Chen et al. tried to connect vulnerability with probability and the consequence of risk [12]. Bell et al. analyzed vulnerability based on the probability of capacity degradation with game theory [13]. Erath et al. did research on the vulnerability of the transportation network in Switzerland, where vulnerability was defined as the product of the probability of capacity degradation and the combination of the direct and indirect results caused by interruption [14].

In short, the most current road network vulnerability researches focus on the consequence of some road network units' failure. Those researches can be separated into two kinds: one believes that vulnerability is only related to the consequence that some units' fail, while it has nothing to do with the probability of the failure; another believes that vulnerability is closely related to the risk, in other words, the product of the probability of the failure and its consequences. The shortcoming of this focus is that it cannot reflect the intrinsic quality of vulnerability, that is to say the sensitivity of the traffic state to the disturbance. In this paper, we will define the vulnerability based on the easiness of disruption to traffic operational performance.

Several studies have engaged vulnerability analysis for large-scale road networks (see, e.g., [2, 15–17]). A measure of road network vulnerability involves solving the traffic assignment problem repeatedly. As a result, computational burden has long been recognized as one of the most restrictive issues in this analysis. Therefore, many other algorithms are also proposed to conduct traffic assignment, such as the Frank-Wolfe algorithm, neural network algorithm, genetic algorithm, and fuzzy algorithm. However, it is impossible for these algorithms to simulate the process of how vehicles select routes, veritably and dynamically. A difference between the traffic assignment outcomes and reality also exists. To address this problem, the improved ant colony algorithm is introduced to conduct traffic assignment. The ant colony algorithm not only owns an ability of stochastic searching optimization, but also has the attribute of adaptive distributed computation. Moreover, its optimal path searching procedure is very similar to the process of how vehicles select routes; consequently, we will use the ant colony algorithm to solve the traffic assignment problem.

1.2. The Objectives and Organization of the Study. The assessment of road network vulnerability is a new hot spot and difficult in present research. Discriminating and supervising the vulnerability of the transportation network effectively is a significant study to help improve resilience in emergency events. Therefore, developing methods for the evaluation of road network vulnerability would be interesting and meaningful. This will constitute a more useful performance measure of a road network. This measure will be used for

robust network planning and design. The objective of the proposed study is to develop methods for evaluating and determining the vulnerable road segments in a road network.

This paper is organized as follows. The next section presents the definition of road network vulnerability. Then, Section 3 builds up the vulnerability model. In Section 4, the improved ant colony algorithm is developed to solve the traffic assignment problem for vulnerability analysis. Section 5 provides two examples to demonstrate the applications of the proposed model and methodology. In Section 6, conclusions are presented.

2. Definition

Vulnerability was defined by many scholars from different angles. Timmerman took the vulnerability as a kind of sensitivity of road network system to disasters. The sensitivity depends on the flexibility of the system, which indicates the ability of the system to recover from the disaster [5]. Jenelius and Mattsson divided the concept of vulnerability into two kinds; one is the probability of dangerous incidents, while the other is the result of the events that occurred at a specified place [18]. Husdal defined the road network vulnerability as the function of the degradation of the road network in some certain circumstances [11].

The implication of vulnerability is that some part or portion of a system disturbed by internal and/or external causes will break down. The result is that some portion of the system, perhaps the whole system, would be affected directly or indirectly, and eventually the whole system will be destroyed. This characteristic is called the vulnerability of a complicated system. This paper defines vulnerability as the sensitivity of some part of the system disturbed by internal and/or external causes leading to a change in other parts of the system or the system in its entirety, which reflects the adaptation to the traffic environment. From another viewpoint, vulnerability shows the extent and ease of interference to the traffic condition.

2.1. Definition Demarcation

2.1.1. The Origins of Vulnerability. Vulnerability is caused by internal and external factors and is the result of interconnections between both causes under specified space-time conditions. The internal causes relate to the reasonability of the road network structure, while the external causes relate to the disturbances of the external environment such as traffic accidents, road maintenance, bad weather, terrorist activities, and many other emergency events.

2.1.2. Evaluation Criterion. According to the definition proposed in this paper, the core of the evaluation criterion is the degree of sensitivity to disturbances of the traffic conditions, that is to say the extent and ease of interference to the traffic conditions. Although most definitions focus on the result produced by a disturbance, we will pay more attention to the degree of sensitivity in this paper.

3. Establishment of the Vulnerability Model

3.1. Utility Index of Road Network Unit. Once a road network unit loses or decreases its efficacy, the traffic on it will be redistributed to other alternative routes [10]. This traffic redistribution will induce the loss or degradation of other units' efficacy, triggering the dominoes phenomenon. In other words, this will cause other units' or even the whole traffic system's operational performance to decrease rapidly. In this way, the vulnerability is spread throughout the system. The consequence of this phenomenon is the travel time of the units or even the whole transportation system would vastly increase to a point where on some roads paralysis could occur which would make some specific routes impassible. For this purpose, a utility index is introduced to measure the influence of the degraded unit on all of the other road network units [10]. Consider

$$e_a = \frac{\Delta C_a}{C_a} \cdot \frac{c_a^0}{c_a}, \quad (1)$$

where C_a is the capacity of the road network unit a , pcu/h; ΔC_a is the surplus capacity of the road network unit a , $\Delta C_a = C_a - x_a$, pcu/h; x_a is the traffic volume on the road network unit a , pcu/h; c_a^0 is the travel cost of the road network unit a in free flow; c_a is the travel cost of the road network unit a ; and e_a is the utility index of the road network unit a . While e_a approaches zero traffic conditions are worsening. When the traffic volume x_a is equal to the capacity C_a , $\Delta C_a = C_a - x_a \rightarrow 0$, the utility index is zero. On the contrary, this value tends to one when the traffic conditions approach the free-flow condition.

Travel time is the dominant element of travel cost. Most other elements of travel cost are closely related to travel time. Thus, in this paper, travel time is selected to measure travel cost, and the following BPR (Bureau of Public Roads) function is used to calculate it:

$$T_a = t_a \left[1 + \beta \left(\frac{x_a}{C_a} \right)^n \right], \quad (2)$$

where T_a is the travel time of road segment a , s; t_a is the free-flow travel time on road segment a , s; x_a and C_a are, respectively, the traffic volume and capacity on road segment a , pcu/h; and β and n are parameters with recommended values of $\beta = 0.15$, $n = 4$.

3.2. Vulnerability Model. The utility index was introduced to characterize traffic conditions. Then the vulnerability was defined as the ratio of the variation of the utility index to the initial utility index to reflect the sensitivity of traffic conditions to disturbance. The formulation can be written as

$$V_a = \frac{e'_a - e_a}{e_a}, \quad (3)$$

where e'_a is the initial utility index of the road network unit a ; e_a is the utility index of the disturbed road network unit a ; and V_a is the vulnerability index of the road network unit a . It is obvious that a smaller V_a results in a lower vulnerability

of this road unit or higher resilience. Otherwise when V_a is higher, this road unit is more vulnerable and is more likely to be disturbed.

4. Traffic Assignment Based on Improved Ant Colony Algorithm

A measure of road network vulnerability involves solving traffic assignment problems repeatedly. Traffic will be redistributed in the road network when the network is disturbed under conditions such as adverse weather, traffic accidents, and the failure of one road segment or a collection of several road segments. As a result, computational burden has long been recognized as one of the most restrictive issues in this type of analysis. It is well known that ants are good at finding food and that they leave behind a pheromone trail that other ants can follow to reach the food. The ant colony algorithm simulates the way that the ant releases pheromone on the route and that its behavior is influenced by other ants. The ant colony algorithm has been successfully used to improve traffic assignment by many scholars [19–23]. Concrete steps to determine the optimal path with the primary ant colony algorithm are as follows.

- (1) Parameter initialization: set the iteration number as $n = 0$; $\tau_{ij}(t)$ is the pheromone density on connection nodes and roads at t , $\tau_{ij}(0) = C$ (constant). η_{ij} is the heuristic information of the road segment (i, j) , in this problem; $\eta_{ij} = 1/d_{ij}$, where d_{ij} is the length of the road segment (i, j) .
- (2) Dropping m ants on original point and putting it in present disaggregation, for ant k , there is a possibility of $p_{ij}^k(t)$ to move to next vertex j and then put vertex j in present disaggregation as well. Consider

$$p_{ij}^k = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\gamma}{\sum_{j \in \text{allowed}} [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\gamma}, & j \in \text{allowed} \\ 0, & \text{otherwise,} \end{cases} \quad (4)$$

where α is the relative importance of the pheromone ($\alpha \geq 0$); γ is the relative importance of the heuristic information ($\gamma \geq 0$).

- (3) Calculating the objective function of the minimum travel cost of every ant and recording the present optimal solution.
- (4) Modifying the pheromone density on every road segment according to the pheromone renewal equation. The pheromone renewal equation of road segment (i, j) at t is as follows:

$$\tau_{ij} = \rho \tau_{ij}(t-1) + \sum_k \Delta \tau_{ij}^k(t-1), \quad (5)$$

where ρ is the retention rate of the pheromone on road segment (i, j) ; $\Delta \tau_{ij}^k(t-1)$ is the pheromone quantity that ant k left on segment (i, j) per unit

length at $(t - 1)$. The ant colony algorithm can be divided into three forms: ant-cycle model, ant-density model, and ant-quantity model. Since the pheromone density released by ants is independent with the length d_{ij} of road segment (i, j) in ant-cycle model and ant-density model, in this paper, we adopt the following ant-quantity model to calculate the pheromone density released by ants on road segment (i, j) . Consider

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{d_{ij}}, & (i, j) \text{ is on the optimal route} \\ 0, & \text{else,} \end{cases} \quad (6)$$

where Q is a constant of the pheromone density released by ants.

- (5) Set $\Delta\tau_{ij}^k(t) = 0$, $n = n + 1$ for every road segment (i, j) in the road network.
- (6) If n is less than the number of predetermined iterations, then return to step (2).
- (7) Output the optimal path.

4.1. Improved Ant Colony Algorithm. Traffic volume on road segments will increase with each execution of traffic assignment. At the same time, the impedance of road segment (i, j) will be changed as well. As more traffic would be distributed to the shorter road segments, which may lead to an increase in travel time on these road segments. On the other hand, some road segments are longer, but due to low traffic volumes, the travel time might be less than that of shorter routes. Therefore, it is necessary to modify the primary ant colony algorithm in order to accommodate this case.

The modification of ant colony algorithm focuses on changing the $\Delta\tau_{ij}^k(t)$ in the pheromone renewal equation; the concrete steps remain unchanged. In route choice, travelers would pay more attention to travel time, but not the distance. So we should replace distance d_{ij} with travel time T_{ij} , which could be calculated with the BPR function. Then $\Delta\tau_{ij}^k(t)$ may be expressed as

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{T_{ij}}, & (i, j) \text{ is on the optimal route} \\ 0, & \text{else.} \end{cases} \quad (7)$$

Bringing the BPR function into (7), then

$$\begin{aligned} \Delta\tau_{ij}^k(t) &= \begin{cases} \frac{Q}{t_{ij}^0(1 + \beta(q_{ij}/C_{ij})^n)}, & (i, j) \text{ is on the optimal route} \\ 0, & \text{else.} \end{cases} \\ & \quad (8) \end{aligned}$$

In the ant colony algorithm, the heuristic information $\eta_{ij} = 1/d_{ij}$ on the road segment (i, j) only reflects the distance between the present nodes and connected points

without considering the distance between the next node and destination point. Therefore a new parameter d_{jE} is introduced which is the optimal route distance between node j and destination point (food source) E . Then, set $\eta_{ij} = 1/(d_{ij} + d_{jE})$ which will allow the optimal solution to be determined. This new heuristic information η_{ij} could strengthen the search directionality. Note that the distance used in the improved ant colony algorithm should be replaced by travel time, so $\eta_{ij} = 1/(d_{ij} + d_{jE})$ should be expressed as $\eta_{ij} = 1/(T_{ij} + T_{jE})$.

The optimal solution can be figured out from the fact that the new heuristic information η_{ij} could strengthen the directionality in searching.

4.2. Traffic Assignment Model Based on the Improved Ant Colony Algorithm. The estimated minimum travel cost is assumed as the route choice criterion for travelers. Due to the complexity of the practical road network and the randomness of the traffic condition, the route choice is always random. As a result, the route choice behavior can be described by SUE (stochastic user equilibrium). The improved ant colony algorithm is used to solve the traffic assignment problem. The concrete steps for the incremental traffic assignment algorithm are as follows.

Step 0. Initialization: divide the OD traffic volume into N shares, according to a certain distribution rate λ . Each share is $q_{rs}^n = q_{rs} \cdot \lambda$. At the same time, set $n = 1$ and $q_{ij}^0 = 0$, $\forall(i, j)$, while every parameter in the primary ant colony algorithm should be initialized.

Step 1. Use the primary ant colony algorithm to identify the shortest path in every OD pair. Then load the traffic volume q_{rs}^n on the shortest path to conduct the first traffic assignment, and set $n = 2$.

Step 2. Renew the travel time of every road segment, $t_{ij}^n = t_{ij}(q_{ij}^{n-1})$, $\forall(i, j)$, then identify the shortest path in every OD pair with the improved ant colony algorithm.

Step 3. Assign the traffic volume q_{rs}^n to every road segment in the road network and the additional traffic volume e_{ij}^n can be calculated.

Step 4. Renew the traffic volume by letting $q_{ij}^n = q_{ij}^{n-1} + e_{ij}^n$, $\forall(i, j)$.

Step 5. Make a judgment about whether this calculation ended or not, according to the number of iterations. If $n = N$, then q_{ij}^n is the result; otherwise, $n < N$ and set $n = n + 1$; return to Step 2.

5. Numerical Examples

5.1. Test Road Network. A simple test road network is used to demonstrate the performance of the proposed model and methodology. This numerical network consists of four nodes and five arcs, as shown in Figure 1, where traffic demand

TABLE 1: Property data of arcs.

Arc number	1	2	3	4	5
t_a (min)	1	2.0	1.1	1.2	1.1
C_a (veh/min)	17	13	13	19	12

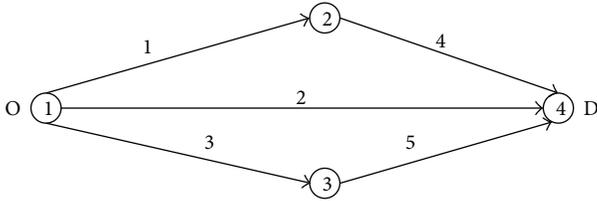


FIGURE 1: Test road network.

among OD(1,4) is 24 veh/min. $\beta = 0.15$ and $n = 4$ in the BPR. The free-flow travel time and capacity are shown in Table 1.

To illustrate the effect of traffic demand on vulnerability, we assume that the capacity of every arc is constant and traffic demand increases by 10%, 20%, 30%, and 40%, respectively. Traffic assignment was performed for the above scenarios with the improved ant colony algorithm. Firstly, according to formula (2), we can get the travel cost of all road segments with the results of traffic assignment. Then according to formula (1), the utility index may be calculated with the results of traffic assignment and travel cost. Finally, according to formula (3), we can get the vulnerability with the results of traffic utility index. The results are shown in Table 2 and Figure 2. The results show that (1) with increasing traffic demand, the vulnerability also increases and (2) due to the differences of every arc, the vulnerability is different for each of them under the same traffic demand disturbance. For example, the curve of arc 4 has the lowest vulnerability index and smallest slope, which means that both the vulnerability and the sensitivity to traffic demand increases are the lowest, followed by arc 3, arc 1, and arc 5. The curve of arc 2 has the highest vulnerability index and the steepest slope. This means that the vulnerability and the sensitivity to traffic demand increases in arc 2 are the highest. This is closely related to the fact that the capacity of arc 2 is the smallest. In general, a greater road capacity contributes to a stronger ability to handle growing traffic demand disturbances. In other words, a greater road capacity tends to an arc that is less sensitive to increasing traffic demand and has smaller vulnerability. The opposite is also true, which implies that a smaller capacity yields higher vulnerability.

To illustrate the effect of an arc's capacity degradation on vulnerability, assuming that the traffic demand of every arc is constant and capacity decreases by 10%, 20%, 30%, 40%, and 50%, respectively, the vulnerability of all arcs was studied. Traffic assignment was done for the above scenarios with the improved ant colony algorithm. The utility index and vulnerability, as shown in Table 3 and Figure 3, were calculated with the results of the traffic assignment. The results show that (1), with the degradation of capacity, arcs show increases in vulnerability at different levels and that

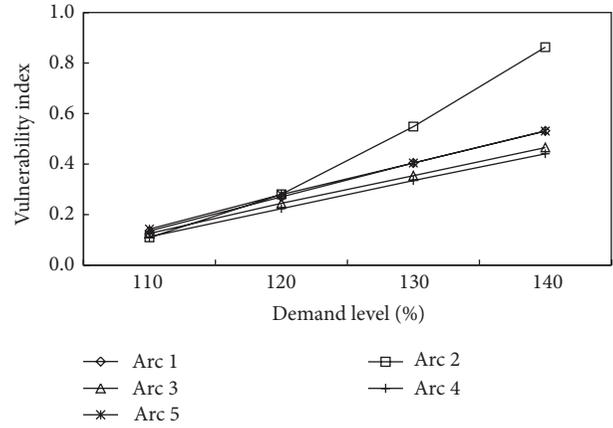


FIGURE 2: Curves of arcs vulnerability at different traffic demand.

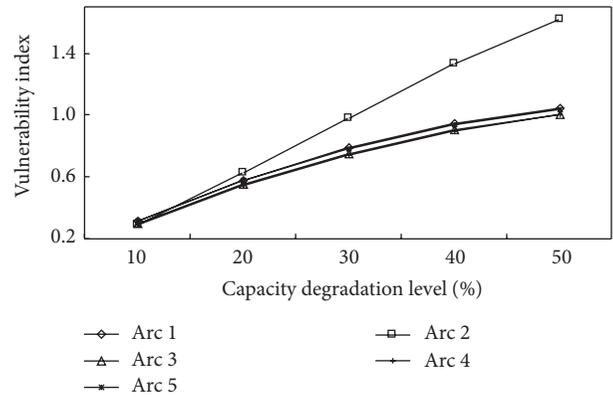


FIGURE 3: Vulnerability of arcs at different capacity degradation.

(2) the vulnerability between every arc should be similar to the result of former traffic capacity analyses.

To study the influence of some arcs' failures on the others, we assume that arcs 3 and 5 are disconnected and all other conditions remain unchanged. Then the traffic volume on arc 3 and arc 5 would be redistributed to arcs 1, 2, and 4; therefore, the traffic volume would increase. As a result, the utility decreases to -0.003 , -0.244 , and 0.090 , and the vulnerability increases to 1.011 , 4.843 , and 0.755 , respectively. From the results it can be seen that the vulnerability of arc 2 is the highest. This is consistent with all former analyses. Arc 2 should, therefore, be considered priority in road reconstruction and maintenance.

In terms of efficiency, the improved ant colony algorithm adopted in this paper reduces the calculation time to 4/5 of the time required by the traditional traffic assignment method.

5.2. *Medium-Sized Road Network.* To demonstrate the applicability of the proposed method, a medium-sized road network, as shown in Figure 4, was used as a test. The road network consists of 106 directed arcs, 37 nodes, and 921 OD movements. The arc travel time model is set to $T_a = t_a [1 + 0.15(x_a/C_a)^4]$.

TABLE 2: Utility and vulnerability of arcs at different traffic demands.

Demand level		Arc 1	Arc 2	Arc 3	Arc 4	Arc 5
100%	x_a	7.152	12.072	4.776	7.152	4.776
	c_a	1.004	2.208	1.103	1.203	1.104
	e_a	0.577	0.065	0.631	0.622	0.600
110%	x_a	8.448	12.170	5.782	8.448	5.782
	c_a	1.009	2.215	1.106	1.207	1.108
	e_a	0.499	0.058	0.552	0.552	0.514
	V_a	0.135	0.109	0.125	0.112	0.143
120%	x_a	9.734	12.326	6.739	9.734	6.739
	c_a	1.015	2.226	1.111	1.212	1.115
	e_a	0.421	0.047	0.477	0.483	0.432
	V_a	0.270	0.280	0.244	0.223	0.279
130%	x_a	11.014	12.574	7.613	11.014	7.613
	c_a	1.025	2.245	1.118	1.219	1.125
	e_a	0.344	0.029	0.408	0.414	0.358
	V_a	0.404	0.548	0.354	0.335	0.404
140%	x_a	12.230	12.869	8.501	12.230	8.501
	c_a	1.038	2.269	1.128	1.229	1.139
	e_a	0.270	0.009	0.337	0.348	0.282
	V_a	0.531	0.862	0.465	0.440	0.530

TABLE 3: Utility and vulnerability of arcs at different capacity degradation.

Capacity degradation level		Arc 1	Arc 2	Arc 3	Arc 4	Arc 5
Normal	x_a	7.152	12.072	4.776	7.152	4.776
	c_a	1.004	2.208	1.103	1.203	1.104
	e_a	0.577	0.065	0.631	0.622	0.600
10%	x_a	7.752	10.968	5.280	7.752	5.280
	c_a	1.009	2.216	1.106	1.207	1.109
	e_a	0.396	0.046	0.442	0.440	0.411
	V_a	0.313	0.293	0.300	0.292	0.315
20%	x_a	8.328	9.960	5.712	8.328	5.712
	c_a	1.020	2.236	1.114	1.215	1.119
	e_a	0.243	0.024	0.285	0.286	0.255
	V_a	0.578	0.625	0.549	0.540	0.575
30%	x_a	8.832	9.072	6.096	8.832	6.096
	c_a	1.043	2.277	1.131	1.233	1.143
	e_a	0.121	0.001	0.157	0.160	0.129
	V_a	0.790	0.980	0.751	0.742	0.784
40%	x_a	9.264	8.352	6.384	9.264	6.384
	c_a	1.095	2.368	1.169	1.273	1.195
	e_a	0.030	-0.022	0.061	0.064	0.038
	V_a	0.948	1.333	0.903	0.898	0.937
50%	x_a	9.552	7.848	6.600	9.552	6.600
	c_a	1.223	2.595	1.264	1.372	1.326
	e_a	-0.025	-0.040	-0.003	-0.001	-0.021
	V_a	1.044	1.618	1.005	1.002	1.035

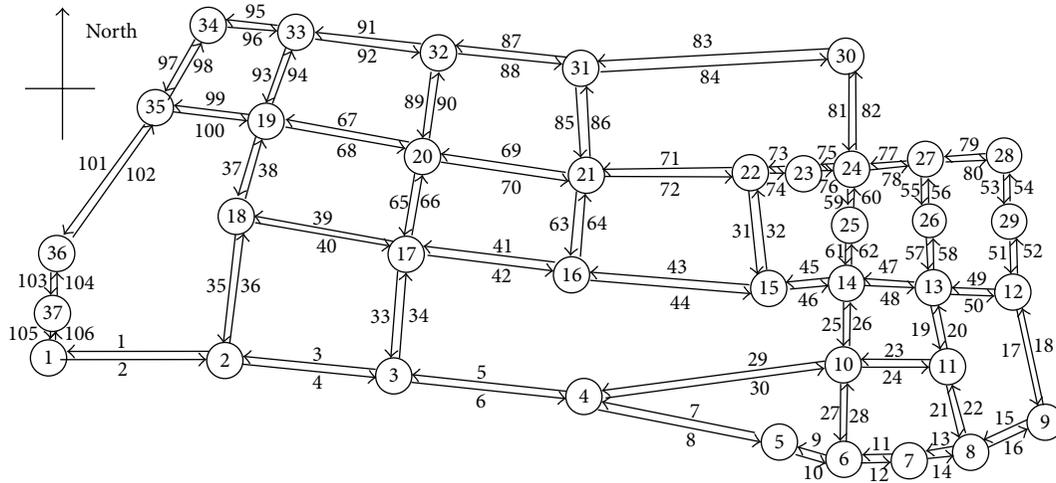


FIGURE 4: Partial road network.

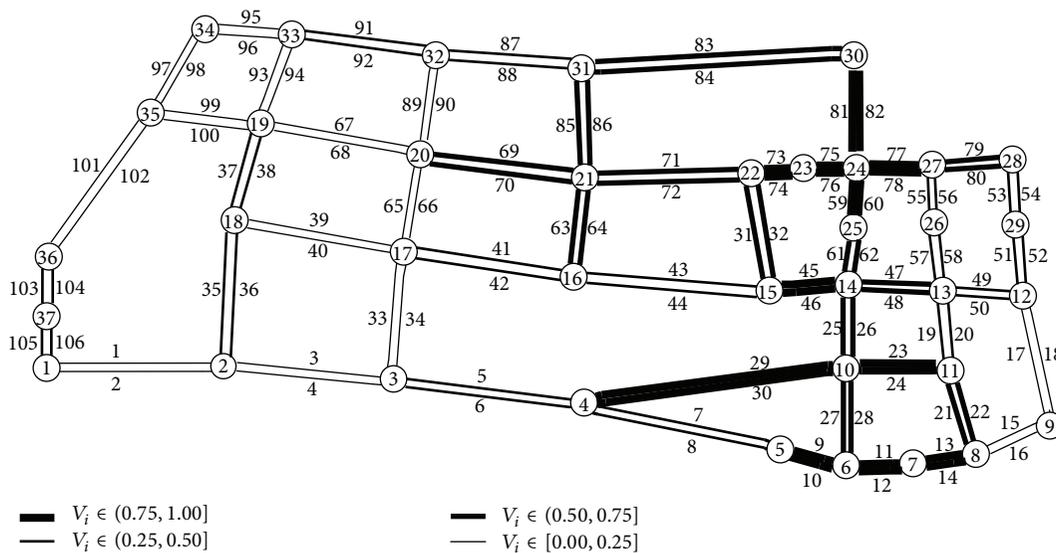


FIGURE 5: Vulnerability of arcs.

To illustrate the effect of different levels of capacity degradation and traffic demand increases on vulnerability, traffic assignment is conducted for the different scenarios with the improved ant colony algorithm. The utility index and vulnerability are calculated with the traffic assignment results. The vulnerability under capacity degradation and traffic demand increases has similar results; for the case with 20% capacity degradation the arcs' vulnerabilities are shown in Figure 5. It demonstrates that the analyzed road network has obvious vulnerable units. For instance, the vulnerability of arcs 9, 10, 11, 12, 13, 14, 23, 24, 29, 30, 45, 46, 59, 60, 73, 74, 75, 76, 81, and 82 is more than 0.75. It is consistent with the fact that the above arcs are bottlenecks in the road network, and congestion often occurs in the above arcs especially in the morning peak hours. Even with small fluctuations, the vulnerability will be magnified and the traffic conditions will get worse.

6. Conclusions

- (1) With the increase of traffic demand or capacity degradation, the vulnerability would be magnified. When the traffic demand is especially close to capacity, the road segment will be very sensitive to an increase of traffic demand, and the vulnerability will be higher.
- (2) There are two reasons why the vulnerability is magnified. One is the contradiction between traffic demand and the capacity supply or the imbalance of the traffic demand and capacity. There are two situations of this imbalance where either the traffic demand has increased too fast to be endured by normal capacity or the capacity has decreased so that it cannot meet the requirement of normal traffic demand. These two situations are both common in traffic congestion. The second reason vulnerability is magnified results from

traffic accidents, road maintenance, traffic control, or other reasons. Some units among the whole transportation network may even be suspended; therefore, the traffic demand would be redistributed to other routes and the traffic volume of these alternative routes would increase and possibly cause a vulnerability increase to emerge.

- (3) The vulnerability of a road network unit is principally relevant to its ability to deal with disturbances. In general, a larger capacity is less sensitive to an increase of traffic demand and decrease of capacity, while the ability to deal with disturbances is greater and the vulnerability is lower. On the contrary, smaller traffic capacities are weaker dealing with disturbances.
- (4) The proposed method based on the improved ant colony algorithm can be used as an effective tool to solve traffic assignments in vulnerability analysis. The vulnerability evaluation may provide the theoretical foundation for road network reforming, planning, and designing.
- (5) This paper focused on the vulnerability of individual road network units, although knowing the vulnerability of the whole road network is important for the traffic manager. Future efforts will be put on determining the road network's vulnerability as a whole.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The work described in this paper was jointly supported by the China Postdoctoral Science Foundation (2011M500676), the National Education Ministry Humanities and Social Sciences Foundation (12YJCZH097), and the Technology Research and Development Program of Shandong (2012G0020129).

References

- [1] D. Watling and N. C. Balijepalli, "A method to assess demand growth vulnerability of travel times on road network links," *Transportation Research Part A: Policy and Practice*, vol. 46, no. 5, pp. 772–789, 2012.
- [2] P. Luathep, A. Sumalee, H. W. Ho, and F. Kurauchi, "Large-scale road network vulnerability analysis: a sensitivity analysis based approach," *Transportation*, vol. 38, no. 5, pp. 799–817, 2011.
- [3] B. F. Santos, A. P. Antunes, and E. J. Miller, "Interurban road network planning model with accessibility and robustness objectives," *Transportation Planning and Technology*, vol. 33, no. 3, pp. 297–313, 2010.
- [4] K. Berdica, "An introduction to road vulnerability: what has been done, is done and should be done," *Transport Policy*, vol. 9, no. 2, pp. 117–127, 2002.
- [5] P. Timmerman, "Vulnerability, resilience and the collapse of society," *Environmental Monograph—Institute for Environmental Studies, University of Toronto*, vol. 1, 1981.
- [6] S. D. Clark and D. P. Watling, "Sensitivity analysis of the probit-based stochastic user equilibrium assignment model," *Transportation Research Part B: Methodological*, vol. 36, no. 7, pp. 617–635, 2002.
- [7] R. D. Connors, A. Sumalee, and D. P. Watling, "Sensitivity analysis of the variable demand probit stochastic user equilibrium with multiple user-classes," *Transportation Research Part B: Methodological*, vol. 41, no. 6, pp. 593–615, 2007.
- [8] E. S. Barber, *Calculation of Maximum Pavement Temperature from Weather Reports*, National Academy of Sciences National Research Council, Washington, DC, USA, 1995.
- [9] M. A. P. Taylor and G. M. D'Este, *Transport Network Vulnerability: A Method for Diagnosis of Critical Locations in Transport Infrastructure Systems, Critical Infrastructure: Reliability and Vulnerability*, edited by: A. T. Murray and T. H. Grubestic, Springer, New York, NY, USA, 2007.
- [10] P. M. Murray-Tuite and H. S. Mahmassani, "Methodology for determining vulnerable links in a transportation network," *Transportation Research Record*, no. 1882, pp. 88–96, 2004.
- [11] J. Husdal, "Reliability and vulnerability versus cost and benefits," in *Proceedings of the 2nd International Symposium on Transportation Network Reliability*, Queenstown, New Zealand, 2004.
- [12] A. Chen, C. Yang, S. Kongsomsaksakul, and M. Lee, "Network-based accessibility measures for vulnerability analysis of degradable transportation networks," *Networks and Spatial Economics*, vol. 7, no. 3, pp. 241–256, 2007.
- [13] M. G. H. Bell, U. Kanturska, J.-D. Schmöcker, and A. Fonzone, "Attacker-defender models and road network vulnerability," *Philosophical Transactions of the Royal Society of London. Series A. Mathematical, Physical and Engineering Sciences*, vol. 366, no. 1872, pp. 1893–1906, 2008.
- [14] A. Erath, J. Birdsall, and K. W. Axhausen, "Vulnerability assessment of the Swiss road network," in *Proceedings of the 88th Transportation Research Board Annual Meeting*, Washington, DC, USA, 2009.
- [15] K. Berdica and L. Mattsson, "Vulnerability: a model-based case study of the road network in Stockholm," in *Critical Infrastructure: Reliability and Vulnerability*, A. T. Murray and T. H. Grubestic, Eds., pp. 81–106, Springer, Heidelberg, Germany, 2007.
- [16] E. Jenelius, "Network structure and travel patterns: explaining the geographical disparities of road network vulnerability," *Journal of Transport Geography*, vol. 17, no. 3, pp. 234–244, 2009.
- [17] M. A. P. Taylor, "Critical transport infrastructure in urban areas: impacts of traffic incidents assessed using accessibility-based network vulnerability analysis," *Growth and Change*, vol. 39, no. 4, pp. 593–616, 2008.
- [18] E. Jenelius and L.-G. Mattsson, "Developing a methodology for road network vulnerability analysis," in *Nectar Cluster Seminar*, Molde University College, Molde, Norway, May 2006.
- [19] Z. Su, P. Wen, and R. Zhai, "Research on traffic assignment model based on multi-objective and ant colony algorithm," in *Proceedings of the 8th International Conference of Chinese Logistics and Transportation Professionals—Logistics: The Emerging Frontiers of Transportation and Development in China*, pp. 2926–2931, August 2008.
- [20] R. Qiao, X. Zhu, and Y. Liu, "An improved ant colony algorithm for UE traffic assignment problem," in *Proceedings of the 7th*

International Conference of Chinese Transportation Professionals Congress 2007: Plan, Build, and Manage Transportation Infrastructures in China, pp. 542–548, May 2007.

- [21] L. D’Acierno, B. Montella, and F. De Lucia, “A stochastic traffic assignment algorithm based on ant colony optimization,” in *Proceedings of the 5th International Workshop Ant Colony Optimization and Swarm Intelligence (ANTS ’06)*, pp. 25–36, 2006.
- [22] L. D’Acierno, M. Gallo, and B. Montella, “Ant Colony Optimisation approaches for the transportation assignment problem,” in *Proceedings of the 16th International Conference on Urban Transport and the Environment, Urban Transport*, pp. 37–48, May 2010.
- [23] Z. Zhou, G. Ren, F. Yi, and C. Xu, “Optimal routing search method based on improved ant colony algorithm,” in *Proceedings of the 2nd International Conference on Transportation Engineering (ICTE ’09)*, pp. 1530–1535, July 2009.

Research Article

Pattern Synthesis of Planar Nonuniform Circular Antenna Arrays Using a Chaotic Adaptive Invasive Weed Optimization Algorithm

Huaning Wu, Chao Liu, and Xu Xie

Department of Electronic Engineering, Naval University of Engineering, Wuhan 430033, China

Correspondence should be addressed to Huaning Wu; wuhuaning007@163.com

Received 17 February 2014; Revised 7 May 2014; Accepted 21 May 2014; Published 19 June 2014

Academic Editor: Fang Zong

Copyright © 2014 Huaning Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A novel invasive weed optimization (IWO) variant called chaotic adaptive invasive weed optimization (CAIWO) is proposed and applied for the optimization of nonuniform circular antenna arrays. A chaotic search method has been combined into the modified IWO with adaptive dispersion, where the seeds produced by a weed are dispersed in the search space with standard deviation specified by the fitness value of the weed. To evaluate the performance of CAIWO, several representative benchmark functions are minimized using various optimization algorithms. Numerical results demonstrate that the proposed approach improves the performance of the algorithm significantly, in terms of both the convergence speed and exploration ability. Moreover, the scheme of CAIWO is employed to find out an optimal set of weights and antenna element separation to obtain a radiation pattern with maximum side-lobe level (SLL) reduction with different numbers of antenna element under two cases with different purposes. The design results obtained by CAIWO have comfortably outperformed the published results obtained by other state-of-the-art metaheuristics in a statistically meaningful way.

1. Introduction

In several applications, such as mobile communication and spatial detection, techniques require antennas that have high directive radiation pattern, which cannot be achieved by a single element antenna. Antenna arrays are formed to circumvent such problems by combining many individual antenna elements in particular electrical and geometrical configurations. The primary design objective of antenna array geometry is to determine the locations of array elements that jointly produce a radiation pattern to resemble the desired pattern as nearly as possible. Poor design may result in a polluted electromagnetic environment. This will also result in wastage of power, which is a vital aspect in wireless devices that run on batteries. The classical derivative-based optimizations of designing antenna arrays are not effective as they are prone to getting local optima and strongly sensitive to initialization. Due to these inherent shortcomings of the

classical technique, many modern metaheuristics approaches were tried to achieve optimized side-lobe level (SLL) and null control from the designed arrays [1–6].

Circular arrays have become popular in recent years over array over other array geometries because they have the capability to perform the scan in all directions without a considerable change in the beam pattern and provide 360° azimuth coverage [7]. Moreover, circular arrays are less sensitive to mutual coupling as compared to linear and rectangular arrays since these do not have edge elements [8]. Circular antenna arrays are used in various applications in radar, sonar, mobile, and commercial satellite communication systems. The first metaheuristic approach towards the design circular arrays is real-coded genetic algorithm (GA), which can be traced in the work of Panduro et al. [9]. Later, particle swarm optimization was applied by Shihab et al. in [10] for the design of circular array. Panduro et al. [11] compared three powerful population-based optimization

algorithms PSO, GA, and differential evolution (DE) on the design problem of scanned circular arrays. Gürel and Ergül applied GA in [12] to design a circular array where each element was log-periodic antenna. In literature [13], Roy et al. applied a modified IWO algorithm to design a nonuniform circular antenna arrays. Some other applications of metaheuristics for the design of circular antenna array are found in [7, 14–17] that involves the use of biogeography based optimization (BBO), simulated algorithm (SA), and so on.

Amongst all evolutionary algorithms (EAs) described in various articles, invasive weed optimization has emerged as one of the most powerful tools for solving the real world optimization problems [18]. Invasive weed optimization has been successfully applied to solve problems in electromagnetic as found in literature [19–26]. As IWO is a stochastic search process hence it is not free from false or untimely convergence, in particular over multimodal fitness landscapes. To eradicate this problem, IWO needs to be modified. In this paper, we present here a new powerful variant of IWO denoted by CAIWO, for designing nonuniform circular arrays with optimized performance with respect to SLL, dynamic range ratios, beam width, directivity, and null control in a scanning range $[0^\circ, 360^\circ]$. Comparisons with the other well-known optimizers like GA [9], PSO [10], DE [15], BBO [7], original IWO and MIWO [13] have been made to verify the search ability of the proposed algorithm.

The rest of the paper is organized as follows: Section 2 gives a brief overview of the classical IWO algorithm and a comprehensive overview of the proposed CAIWO algorithm. A formulation of the array pattern synthesis as an optimization task has been discussed in Section 3. Section 4 presents the simulation results and in Section 5 conclusions are presented.

2. The Iwo Algorithm and Its Modification

2.1. Description of Traditional IWO. The original IWO algorithm was initially proposed by Mehrabian and Lucas in [18] for solving continuous optimization problems. IWO mimics the process of colonizing and distributing behavior of weeds. The IWO algorithm may be summarized as four steps, more details can be found in [18, 20].

(I) *Initialization.* Solutions are initialized and dispersed in the given n -dimensional search space uniformly and randomly.

(II) *Reproduction.* Each member of the population is allowed to produce seeds depending on its own, as well as the colony's lowest and highest fitness, so that the number of seeds produced by a weed increases linearly from lowest possible seed for a weed with worst fitness to the maximum number of seeds for a plant with best fitness.

(III) *Spatial Distribution.* The generated seeds are randomly scattered over the d -dimensional search space by perturbing them with normally distributed random numbers with zero

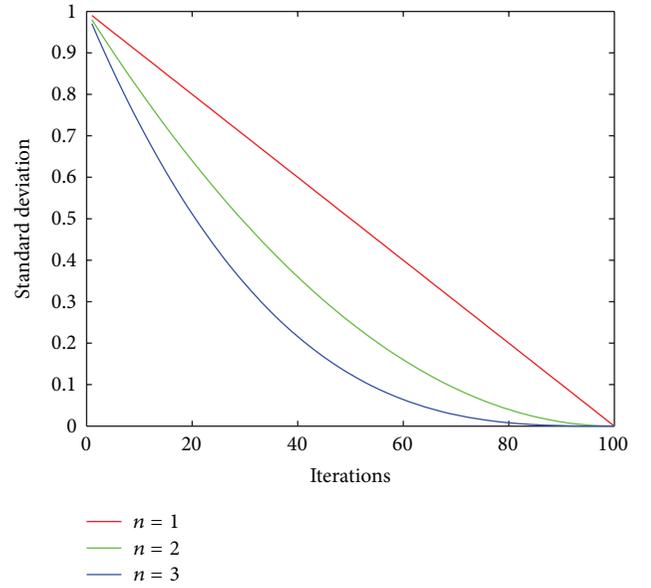


FIGURE 1: Standard deviation over the course of the run.

mean and a variable variance. The standard deviation for a particular iteration can be given as follows:

$$\delta_{\text{cur}} = \frac{(\text{iter}_{\text{max}} - \text{iter})^n}{\text{iter}_{\text{max}}^n} (\delta_{\text{initial}} - \delta_{\text{final}}) + \delta_{\text{final}}, \quad (1)$$

where δ_{initial} and δ_{final} represent the initial and final standard deviations (SD) and iter_{max} and iter are the maximum number of iteration and the current iteration number respectively. n is called the nonlinear modulation index. This is a relatively critical parameter which can influence the convergence performance of the IWO. Through a set of simulations, it has been shown that the best choice for n is 3. Figure 1 shows the standard deviation over the course of a run with 100 iterations and different modulation indexes. Having selected this suitable value for nonlinear modulation index, the algorithm starts with a relatively high SD that the optimizer can scan of the solution space completely. As the iteration number increases and SD value is decreased, the search would be restricted to the neighborhoods around the local minima or maxima to find the global optimal solution.

Then, the position of the new seed can be given as follows:

$$x_{\text{son}} = x_{\text{parent}} + sd = x_{\text{parent}} + \text{rand}n(0, 1) * \delta_{\text{cur}}. \quad (2)$$

(IV) *Competitive Exclusion.* Some kinds of competition between plants are needed for limiting maximum number of plants in a colony. Initially, the plants in a colony will reproduce fast and all the produced plants will be included in the existing colony, until the number of plants in the colony reaches a maximum value p_{max} . The steps (I) to (IV) are repeated until the maximum number of iterations has reached; that is, the colony size is fixed from thereon to p_{max} .

2.2. The Features and Shortcomings of IWO. One important advantage of the IWO is that it allows all of possible candidates to participate in the reproduction process. From step

TABLE 1: Test functions.

Function name	Search range	Dimension
Sphere	$[-100, 100]$	30
Rastrigin	$[-5.12, 5.12]$	30
Ackley	$[-32, 32]$	30
Griewank	$[-600, 600]$	30

(II) in Section 2.1, we can know that the fitter plants produce more seeds than less fit plants, which tends to improve the convergence of the algorithm. In contrast, most metaheuristic algorithms would not allow the less fit plants to produce offspring such as the GA. Another important property of IWO is straightforward and it includes less deal computational burden unlike other methods such as PSO. PSO needs to update both the position and velocity of individuals in each iteration which cost some extra computations to search the best position in the neighborhood of each particle as well as the whole population. Finally, the weeds in IWO can produce seeds without mating so that each agent may have different number of variables during the optimization process. Thus, the number of variables can be chosen as one of the optimization parameters in this algorithm [20].

But there are also some shortcomings of IWO, the most prominent one is concerning the way the seeds produced by a plant are dispersed in the search spaces. As mentioned in Section 2.1, we can know that the function of SD defines the exploration ability and exploitation ability of algorithm and acts as both diversification and intensification components of IWO; it has a great effect on final solutions. In early iterations, the bigger SD will help the algorithm to explore the solution space as much as it can. A good diversification will make the final solution near to global optimum. The algorithm will use this component to identify most the potential spaces where the global optimum may lie in. A good intensification will help the algorithm to exploit the potential areas to find the global optimum. It will increase the convergence speed of the algorithm and search the better final solution. Hence, it is very important to keep an efficient balance between diversification and intensification of the algorithm. But, IWO algorithm uses a fix SD to produce seeds related to each weed, and suffers from the lack of fine balance between exploration and exploitation.

2.3. Chaotic Adaptive Invasive Weed Optimization. In order to overcome the drawbacks of IWO, a chaotic search and adaptive dispersion mechanism are integrated into the IWO algorithm. This variant IWO is named CAIWO.

2.3.1. Chaotic Search. Chaotic search methods have a greater ability to escape from the local minima. Therefore, the CAIWO algorithm has a less chance of premature convergence compared to original IWO. There are many chaotic maps such as logistic map, sinusoidal map, and tent map. We chose the sinusoidal map to improve the performance of IWO based on the discussion in the literature [27].

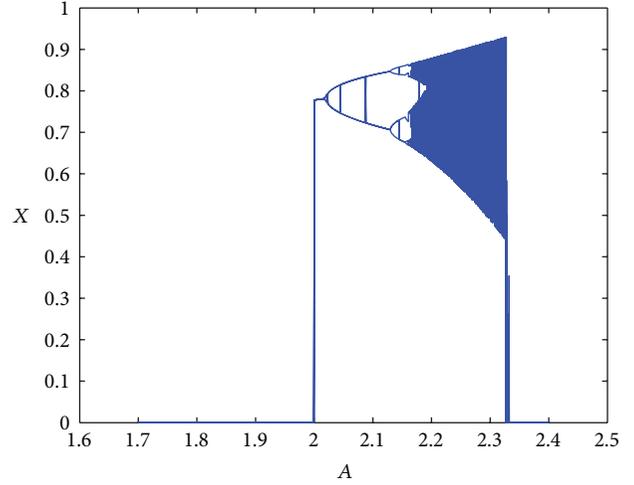


FIGURE 2: The bifurcation diagram for sinusoidal map.

The sinusoidal map or the sinusoid sequence is defined by

$$X_{k+1} = AX_k^2 \sin(\pi X_k) \quad (3)$$

which ensures chaotic behavior in the span of $(0, 1)$. Figure 2 shows the variations of the chaotic variable versus changes in the control parameter. As it can be seen, the performance of the system becomes chaotic when $A = 2.3$.

2.3.2. Adaptive Dispersion. The adaptive dispersion mechanism is that the SD of the current generation distribute linearly among the weeds as weed with the highest fitness achieves the lowest SD and the lowest fitness achieves highest SD, which can be represented by (4). j is the index of weeds in the colony which sorted according to their fitness, σ_{current} can be calculated by (1), p_{sum} is the sum number of weeds in the current generation, and σ_j is the SD of j th weeds to produce seeds. Hence, the plant with lower fitness will have the chance to produce good seeds in current generation. In addition, this concept will increase the diversification of algorithm so the algorithm will explore the search space effectively. Consider

$$\sigma_j = (\sigma_{\text{current}}) \times \left(\frac{j}{p_{\text{sum}}} \right). \quad (4)$$

2.3.3. CAIWO. The goal of the optimization algorithm is to minimize $F(X_1, X_2, \dots, X_D)$ subject to $X_{\min}^i < X_i < X_{\max}^i$, where $i = 1, 2, \dots, D$ and D is the dimension of the problem. A plant in the CAIWO represents a solution $X = (X_1, X_2, \dots, X_D)$ in the real problem. The steps of the proposed CAIWO algorithm proceed as follows.

- (1) Initially, pioneering solutions are initialized in the search space range of (X_{\min}, X_{\max}) . Chaotically distribute the pioneering solutions using the sinusoidal map described in Section 2.3.1. It is worth noting that the variables should be normalized to the range of $(0, 1)$ before applying the chaotic map. The normalization procedure is described as follows.

TABLE 2: Simulation results on benchmark functions.

Benchmark function	Algorithm	Fitness of best run	Fitness of worst run	Average value	Standard deviation
Sphere function	IWO	$2.6E - 05$	$7.11E - 05$	$3.97E - 05$	$5E - 6$
	MIWO	$7.93E - 06$	$2.44E - 05$	$1.40E - 05$	$2.00E - 06$
	CAIWO	$1.65E - 10$	$1.93E - 09$	$7.14E - 10$	$1.62E - 10$
Rastrigin function	IWO	0.044136	0.994962	0.797145	0.154781
	MIWO	0.095132	1.241246	0.872624	0.139673
	CAIWO	$1.78E - 06$	$1.34E - 05$	$8.12E - 5$	$1.87E - 6$
Ackley function	IWO	0.083654	0.099234	0.087016	0.001684
	MIWO	0.085449	0.256521	0.854496	0.256521
	CAIWO	$1.88E - 05$	$9.51E - 05$	$6.5E - 05$	$3.12E - 6$
Griewank function	IWO	0.049237	0.150065	$1.02E - 01$	$1.01E - 02$
	MIWO	0.024603	0.113173	$8.93E - 02$	$2.22E - 02$
	CAIWO	$1.18E - 10$	$9.74E - 10$	$6.57E - 10$	$4.74E - 12$

TABLE 3: Problem description.

Problem number	Numbers of array elements	FNBW
1	8	70.27
2	10	55.85
3	12	46.26

(I) Transform variable X to \bar{X} confine in the data range $(0, 1)$:

$$\bar{X} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}. \quad (5)$$

(II) Apply the sinusoidal sequence to transform \bar{X} to a new value \bar{X}' .

(III) Transform \bar{X}' into the range (X_{\min}, X_{\max}) by

$$X = X_{\min} + \bar{X}'(X_{\max} - X_{\min}). \quad (6)$$

- (2) Evaluate each weed, sort, and rank them according to their fitness in the colony.
- (3) Sum the number of current generation and calculate the SD of each weed with respect to their ranking in the colony using (4). And then new seeds are dispersed randomly on the field with the adaptive SD.
- (4) Distribute the newly generated seeds using sinusoidal map in the neighborhood of the parent weed. If the chaotically distributed seed has a better fitness than the previous seed, keep the better one. Otherwise, the chaotic search is continued. The algorithm is guaranteed to converge much faster by taking advantage of the local search superiorities of chaotic search.
- (5) Rank the seeds again and exclude those with lower fitness to reach the maximum number of seeds p_{\max} .
- (6) Continue from step 3 until maximum number of iterations is reached or a criterion is satisfied.

2.4. Simulation of CAIWO to Benchmark Functions. To verify its effectiveness, CAIWO has been applied to classical benchmark functions. All simulations are conducted in a Windows

7 Professional OS using 12-core processors with Intel Xeon (R), 3.33 GHz, 72 GB RAM, and the codes were implemented in MATLAB 7.10. In this section, four benchmark functions including unimodal and multimodal functions in from the test suite of the Congress on Evolutionary Computation (CEC) 2005 Special Session and Competition on Real Parameter Optimization [28] are employed. Test functions, search range (X_{\min}, X_{\max}) , and dimension for functions are listed in Table 1. All the benchmarks functions have been tested for 30 dimensions. Each experiment was carried out with 1000 iterations for the maximum population size of 50. It is a well-known fact that a stochastic optimization algorithm does not get same results over repeated runs on the same problem. So, we report the mean and the standard deviation of the best-of-run values for 30 independent runs of each of the three algorithms.

$$\text{Sphere function: } f_1 = \sum_{i=1}^n x_i^2.$$

$$\text{Rastrigin function: } f_2 = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10).$$

$$\text{Ackley function: } f_3 = -20 * \exp(-0.2 * \sqrt{(1/n) \sum_{i=1}^n x_i^2}) - \exp((1/n) \sum_{i=1}^n \cos(2\pi x_i)) + 20 + \exp(1).$$

$$\text{Griewank function: } f_4 = \sum_{i=1}^n (x_i^2/4000) - \prod_{i=1}^n \cos(x_i/\sqrt{i}) + 1.$$

The performance of CAIWO is compared with MIWO and classical IWO. The simulation results such as fitness of best run, fitness of worst run, and the standard deviation of fitness are shown in Table 2. The convergence of mean fitness values of benchmark functions is shown in Figure 3. From Table 2 and Figure 3, we can see that the CAIWO algorithm obviously performs better than the other two algorithms. It converges successfully and quickly in all functions.

3. Formulation of the Design Problem

The N -element circular array is shown in Figure 4. The elements are nonuniformly spaced on a circle of radius r in the x - y plane. The elements are assumed to be isotropic sources so that the radiation pattern of this array can be

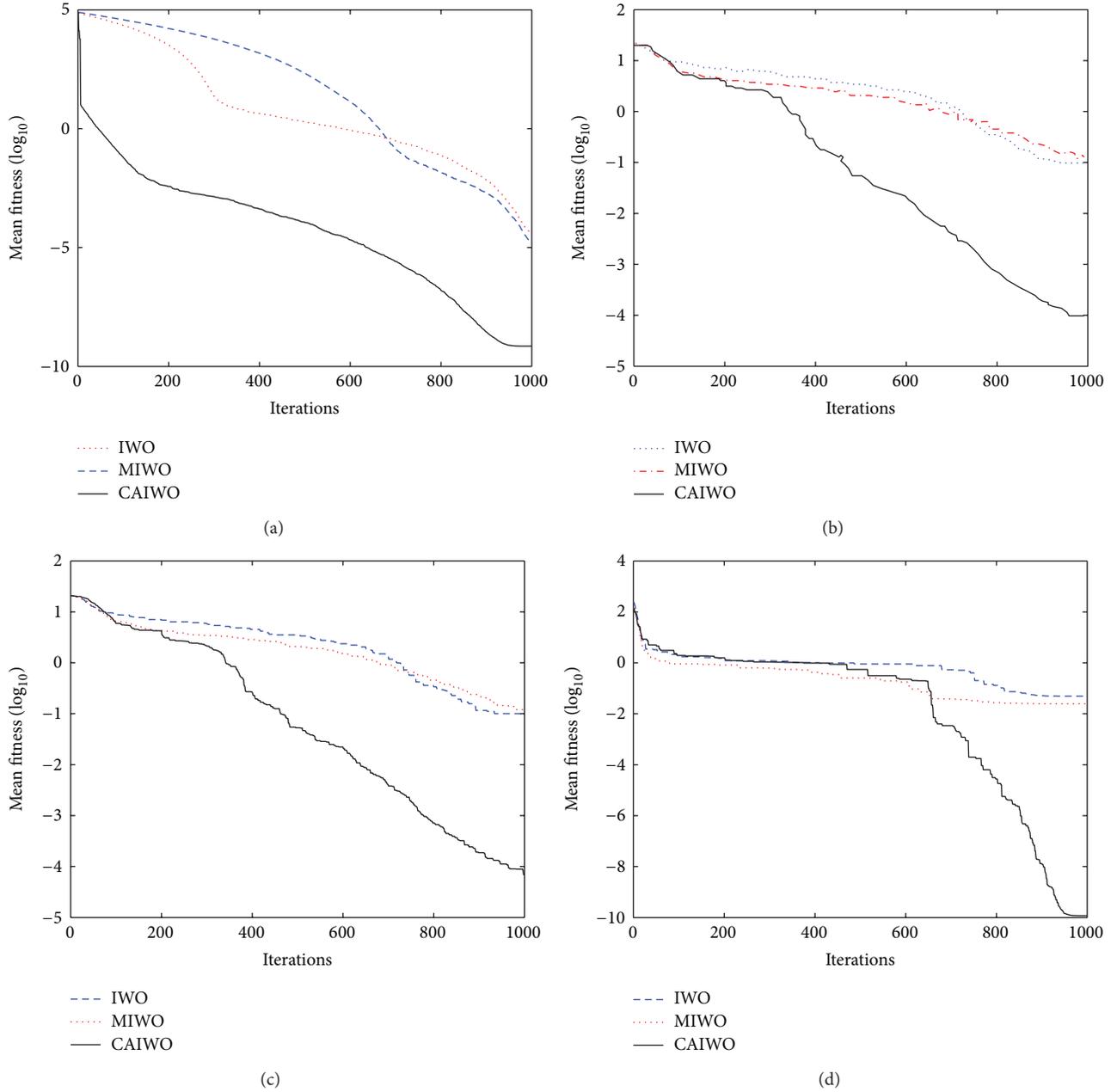


FIGURE 3: Convergence characteristic of IWO, MIWO, and CAIWO over four benchmarks. (a) Sphere. (b) Rastrigin. (c) Ackley. (d) Griewank.

described by its array factor. The formulation of the array factor requires the following: a_n is the normalized amplitude excitation; ψ_n is the phase excitation of the n th element; ϕ_n is the angular position of the n th element; the circular arc separation between any two adjacent elements d_n .

The array factor in the x - y plane can be written by [13]

$$AF(\phi) = \sum_{n=1}^N a_n \cdot e^{j(kr \cos(\phi - \phi_n) + \psi_n)}. \quad (7)$$

kr and ϕ_n can be given by

$$kr = \frac{2\pi r}{\lambda} = \sum_{i=1}^N d_i, \quad (8)$$

$$\phi_n = \frac{2\pi}{kr} \sum_{i=1}^n d_i. \quad (9)$$

When the peak of the array is in ϕ_0 direction, the excitation phase of the n th element can be written as [13]

$$\psi_n = -kr \cdot \cos(\phi_0 - \phi_n). \quad (10)$$

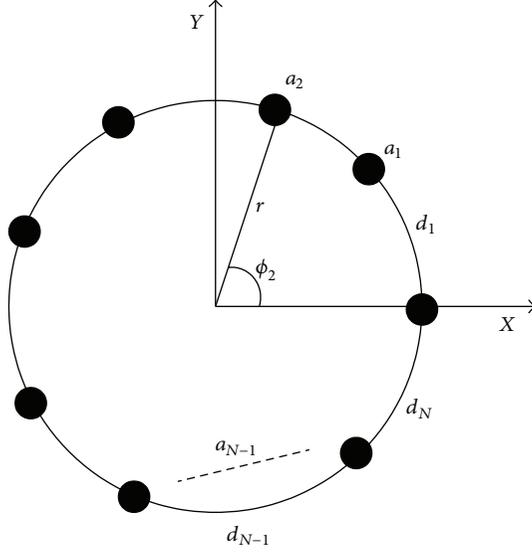


FIGURE 4: Geometry of a nonuniform circular antenna array with N elements.

TABLE 4: Design variables obtained with CAIWO algorithm.

Number of elements	FNBW	d_i in terms of wavelength	Normalized I_n
8	70.27	0.5600, 1.0000	0.4423, 0.3059
		1.0000, 0.5734	0.6405, 0.7325
		0.9509, 0.6363	0.2203, 0.1938
		0.5503, 0.5000	0.3925, 0.8590
10	55.85	0.5001, 1.0000	0.5823, 0.3462
		0.9253, 1.0000	0.41608, 0.7275
		0.5332, 0.5313	0.7032, 0.0737
		0.9433, 0.8756	0.3758, 0.3450
		0.5567, 0.5078	0.4751, 0.9033
12	46.26	0.6448, 1.0000	0.4958, 0.4722
		1.0000, 1.0000	0.0010, 0.4733
		0.6294, 0.5000	0.6587, 0.8100
		0.5000, 0.9564	0.4700, 0.4122
		0.5532, 0.7578	0.2060, 0.4956
		1.0000, 0.5967	0.5455, 0.9976

Then, the array factor can be simplified as [13]

$$AF(\phi) = \sum_{n=1}^N a_n \cdot e^{jkr(\cos(\phi-\phi_n) - \cos(\phi_0-\phi_n))}. \quad (11)$$

4. Design Examples

The first and most important parameter in antenna pattern synthesis is the normalized side-lobe level that is desired to be as low as possible. In this section, the CAIWO algorithm will be applied to determine the electrical and geometrical structure of circular antenna array for obtaining the radiation pattern with minimum SLL in two different cases. Three

circular array antennas with $N = 8, 10,$ and 12 elements are optimized in both cases.

4.1. Case One. The main goal of synthesis of antenna array in case one is to generate the radiation pattern with minimum side-lobe level for a specific first null beam width (FNBW). We have incorporated the maximum side-lobe level in addition to the average side lobe in the fitness function to ensure maximum directivity of the antenna arrays.

4.1.1. Fitness Function of Case One. The following objective functions represent these above requirements in a mathematical form [13]:

$$\begin{aligned} F_{\text{NULL}} &= |AF(\phi_{\text{NULL1}})| + |AF(\phi_{\text{NULL2}})|, \\ F_{\text{SLA}} &= \frac{1}{\pi + \phi_{\text{NULL1}}} \int_{-\pi}^{\phi_{\text{NULL1}}} |AF(\phi)| d\phi \\ &\quad + \frac{1}{\pi - \phi_{\text{NULL2}}} \int_{\phi_{\text{NULL2}}}^{\pi} |AF(\phi)| d\phi, \\ F_{\text{MSL}} &= |AF(\phi_{\text{MSLL1}})| + |AF(\phi_{\text{MSLL2}})|, \end{aligned} \quad (12)$$

where ϕ_{NULL1} and ϕ_{NULL2} are the two angle at the null and ϕ_{MSLL1} and ϕ_{MSLL2} are the angles where the maximum side-lobe level is obtained in the lower band $[-\pi, \phi_{\text{NULL1}}]$ and higher band $[\phi_{\text{NULL2}}, \pi]$, respectively. Combining all of the above objectives, a final cost function can be formulated as follows [13]:

$$F = \omega_1 * F_{\text{NULL}} + \omega_2 * F_{\text{SLA}} + \omega_3 * F_{\text{MSL}}, \quad (13)$$

where ω_i ($i = 1, 2, 3$) represents the weights assigned to the functions. In order to emphasize the three objectives equally, we set ω_i that is equal to 1. Since we are considering nonuniformly spaced circular array, the coupling will be different from one element to another [13]. This is especially critical as we are considering the radiation in the plane of antenna array. In order to get rid of the mutual coupling effect, we set the minimum value of arc separation between any two adjacent elements equal to 0.5λ . If d_i in a solution $(d_1, d_2, \dots, d_n, \dots, d_N)$ is smaller than 0.5λ , this solution will be penalized by adding the following function to the cost function (9):

$$F_p = \sum_i^M (0.5 - d_i), \quad d_i < 0.5. \quad (14)$$

M is the total number of d_i which is smaller than 0.5 in a solution.

4.1.2. Numerical Results. To illustrate the superiority of the CAIWO algorithm, three instantiations of the circular array antenna design problem are solved by using the CAIWO algorithm with five other state-of-the-art algorithms, namely MIWO, classical IWO, DE, PSO, and real coded GA. This DE variant is called DE/rand/1/bin and is the most widely used one in DE literature.

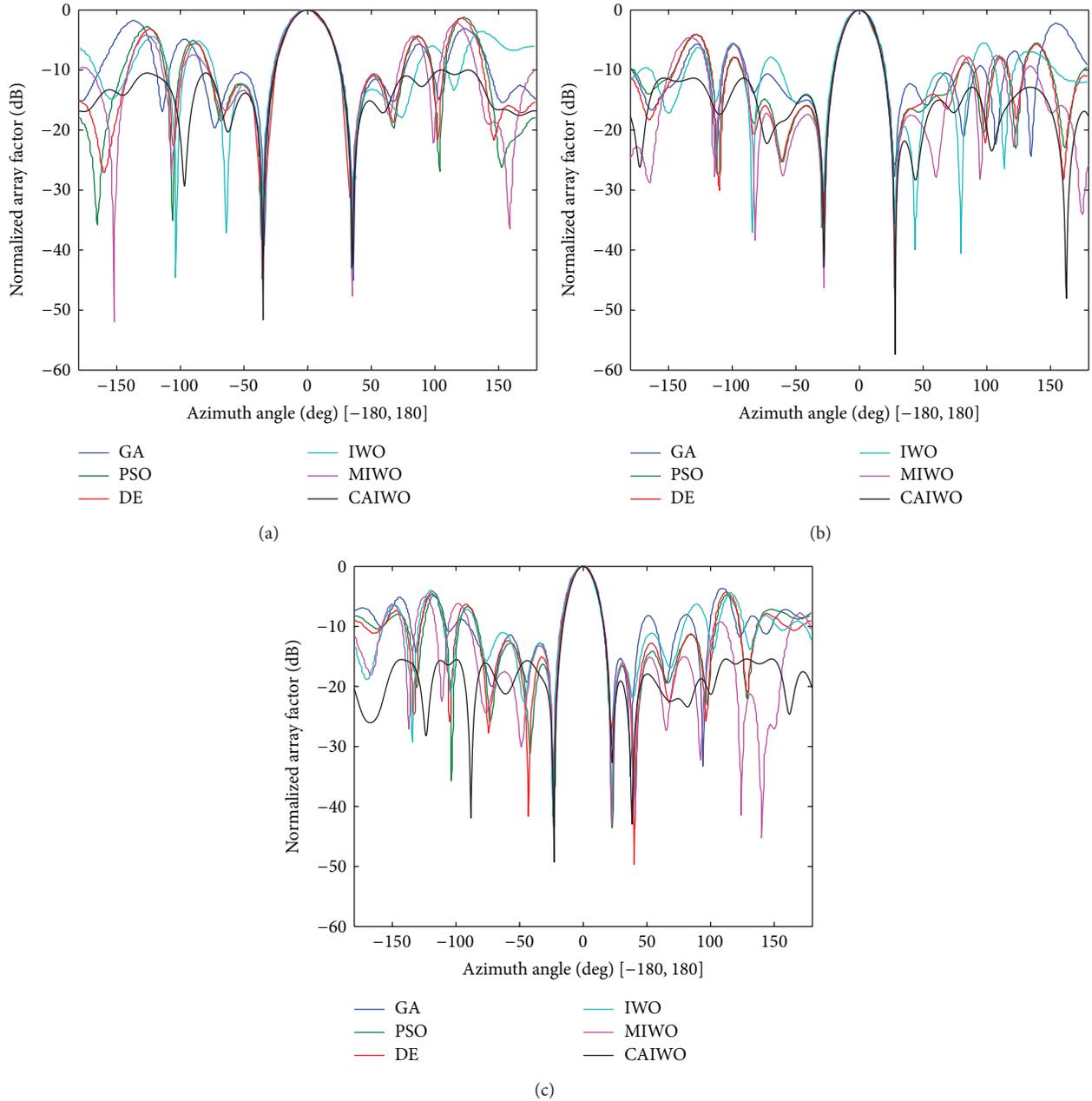


FIGURE 5: Normalized radiation patterns for circular arrays of different number of elements obtained using six different optimization algorithms. (a) For number of elements $N = 8$. (b) For number of element $N = 10$. (c) For number of elements $N = 12$.

The three instantiations are arrays with 8, 10, and 12 elements. The FNBW is assumed to be a constant, corresponding to a uniform circular array with a uniform 0.5λ spacing between the elements. The values of the element amplitudes are allowed to vary between $[0, 1]$ (Table 3).

The control parameters for CAIWO were set through a series of parameter tuning experiments. The parameters for IWO, MIWO, PSO, DE, and GA were set following the guidelines provided in [13, 20]. Once set, the values of parameters were used for each algorithm to solve all three simulation examples. The CAIWO, MIWO, and classical

IWO were let start from the same initial population over all runs on each problem.

Figure 5 presents the radiation patterns of the circular arrays optimized by CAIWO compared to other five algorithms in literature [13] (corresponding to the best of the 50 runs in each case) for 8, 10, and 12 element arrays. The normalized array geometry and current excitation weights are given in Table 4. Figure 6 shows the convergence characteristic of CAIWO over three instances of the circular array design problem (corresponding to the best of the 50 runs in each case) for 8, 10, and 12 element arrays. In Table 5, the

TABLE 5: Design figures of merit obtained in the best (out of 50) runs the six algorithms on three design instances.

Number of elements	Algorithms	Mean SLL (dB)	Max SLL (dB)	Average null depth (dB)	Directivity (dB)
8	CAIWO	-13.158	-10.01	-46.88	9.67
	MIWO	-10.75	-2.17	-46.25	9.14
	IWO	-9.488	-3.61	-34.6	8.27
	DE	-10.34	-1.35	-32.95	8.87
	PSO	-10.23	-1.17	-40.6	8.78
	GA	-9.16	-1.71	-42.2	8.08
10	CAIWO	-16.95	-11.42	-50.16	12.48
	MIWO	-13.78	-4.65	-46.35	11.75
	IWO	-11.35	-5.52	-32.8	9.93
	DE	-12.46	-4.07	-34.4	10.79
	PSO	-12.15	-4.07	-31.1	10.55
	GA	-10.65	-2.18	-30	9.40
12	CAIWO	-18.76	-15.43	-41.015	12.85
	MIWO	-14.23	-5.16	-43.05	12.25
	IWO	-10.99	-4.34	-41.7	9.86
	DE	-11.66	-4.17	-31.4	10.39
	PSO	-11.61	-4.75	-43.55	10.37
	GA	-10.12	-3.68	-32.8	9.21

TABLE 6: Comparisons of results obtained by CAIWO with other algorithms.

Number of elements	Algorithms	Max SLL (dB)	HPBW (°)	ADR	Directivity (dB)	Circumferences
8	CAIWO	-13.0225	17.19	1.4974	9.6365	9.1210
	IWO	-8.1667	14.9	4.6579	8.0011	8.3999
	BBO	-12.24	18.6	5.9595	9.43	9.0710
	SA	-12.00	26.60	3.2383	10.7201	5.8750
	PSO	-10.7996	32	2.7910	7.8269	4.4931
	GA	-9.811	32	3.9417	7.6245	4.4054
10	CAIWO	-14.96	16.04	1.9514	10.2658	9.2333
	IWO	-10.8961	14.9	4.8786	9.5974	9.2637
	BBO	-13.95	16.60	2.6185	10.2032	9.2318
	SA	-13	18.4	3.2179	10.8722	8.0214
	PSO	-12.307	24.34	1.9755	8.6048	5.9029
	GA	-9.811	32	2.8140	9.8037	6.0886
12	CAIWO	-16.50	13.75	2.3018	10.7494	10.5351
	IWO	-11.0042	11.46	7.6718	9.9301	12.0781
	BBO	-14.372	14.8	2.2676	10.8448	10.6453
	SA	-13.91	19.60	3.2639	10.7069	7.9523
	PSO	-13.670	21.2	2.5265	9.3667	7.1419
	GA	-11.83	20.8	4.1439	10.2209	7.7724

best results obtained (out of 50 independent runs) for the aforesaid three problem instances are judged in terms of the average SLL (in decibels), maximum SLL, the average null depth (in decibels), and the directivity (in decibels) for all the six algorithms based approaches.

From Figure 5, we can know that the radiation pattern synthesized by all these algorithms have same first null bandwidth (corresponding to the best of the 50 runs in each

case) for 8, 10, and 12 element arrays, but the average SLLs in lower band and higher band obtained by CAIWO are better than those obtained by the other five algorithms [13].

From Tables 4 and 5, we can clearly state that CAIWO is much better in a statistically significant way than the other five metaheuristics algorithms, namely MIWO, IWO, PSO, DE, and GA, on this specific case of circular array design. A scrutiny of Table 5 shows that the CAIWO obtains better

TABLE 7: Comparison of results obtained by CAIWO with other algorithms for $N = 8$ elements.

PSO [9]	I_n	0.7765	0.3928	0.6069	0.8446	1	0.7015	0.9321	0.3583
	d_n	0.359	0.5756	0.2494	0.7638	0.6025	0.8311	0.7809	0.3308
GA [9]	I_n	0.3289	0.2537	0.7849	1	0.9171	0.5183	0.6176	0.4612
	d_n	0.1739	0.3144	0.662	0.7425	0.6297	0.8929	0.4633	0.5267
SA [17]	I_n	0.3047	0.484	0.7751	0.9867	0.3371	0.4422	0.4067	0.6807
	d_n	0.9997	0.7743	0.9042	0.5652	0.8056	0.7818	0.5848	0.4594
BBO [9]	I_n	1	0.6736	0.1678	1	0.9088	0.6553	0.7571	1
	d_n	0.6341	1	1.8892	0.8456	0.5693	1.1639	1.3329	1.6367
IWO	I_n	0.6188	0.3016	0.3394	0.337	0.2093	0.9749	0.3675	0.7029
	d_n	1.6076	0.3132	1.1011	1.243	0.2924	1.9749	0.7915	1.0762
CAIWO	I_n	0.9406	0.6999	0.6677	0.9998	0.9294	0.6882	0.8101	0.9710
	d_n	0.5747	1.0263	1.7891	0.9507	0.5899	1.1968	1.3859	1.6077

TABLE 8: Comparison of results obtained by CAIWO with other algorithms for $N = 10$ elements.

PSO [9]	I_n	1	0.7529	0.7519	1	0.5062	1	0.7501	0.7524	1	0.5067
	d_n	0.317	0.9654	0.3859	0.9654	0.3185	0.3164	0.9657	0.3862	0.965	0.3174
GA [9]	I_n	0.9545	0.4283	0.3392	0.9074	0.8086	0.4533	0.5634	0.6015	0.7045	0.5948
	d_n	0.3641	0.4512	0.275	1.6373	0.6902	0.9415	0.4657	0.2898	0.6456	0.3282
SA [17]	I_n	0.692	0.5679	0.5937	0.6703	0.9693	0.6014	0.3575	0.302	0.5908	0.9718
	d_n	0.6221	0.988	0.7777	0.9934	0.6217	0.9514	0.7626	0.598	0.7655	0.941
BBO [9]	I_n	1	1	1	0.3819	0.897	1	0.7679	0.8899	0.7246	1
	d_n	0.5301	1.0603	1.3264	1	0.4307	0.4408	1.5276	1.3255	1	0.5904
IWO	I_n	0.2851	0.5493	0.7386	0.1722	0.8401	0.2874	0.3323	0.7330	0.4616	0.4811
	d_n	0.0100	0.5291	0.9957	1.2975	1.7320	0.5507	0.4642	0.7686	1.2974	1.6184
CAIWO	I_n	0.9941	0.9842	0.9518	0.5123	0.9233	0.9858	0.6593	0.9196	0.7336	0.9997
	d_n	0.5910	1.0471	1.2862	1.0304	0.4570	0.4422	1.4190	1.3552	1.0090	0.5962

TABLE 9: Comparison of results obtained by CAIWO with other algorithms for $N = 12$ elements.

PSO [9]	I_n	0.9554	0.6441	0.7109	0.7769	1	1	0.3958	0.7162	0.6746	0.7695	0.9398	0.6145
	d_n	0.2569	0.8509	0.6607	0.7057	0.854	0.3734	0.1609	0.8321	0.6464	0.7079	0.833	0.26
GA [9]	I_n	0.2064	0.5416	0.2246	0.6486	0.7212	0.7913	0.5277	0.3495	0.5125	0.4475	0.5233	0.8553
	d_n	0.4936	0.4184	1.4474	0.7577	0.4204	0.5784	0.452	0.8872	0.7514	0.4202	0.4223	0.7234
SA [17]	I_n	0.6231	0.399	0.3418	0.6054	0.9444	0.738	0.6741	0.3001	0.4311	0.5435	0.4195	0.9795
	d_n	0.8315	0.791	0.6699	0.8087	0.7347	0.5331	0.4777	0.896	0.4874	0.8657	0.3461	0.5105
BBO [9]	I_n	1	0.6501	0.6224	0.502	0.554	1	0.6683	0.7234	0.441	0.5123	0.4793	1
	d_n	0.6704	1	1.3046	0.8081	1	0.431	0.6183	1.1574	1.3465	0.6551	1	0.6539
IWO	I_n	0.8558	0.9912	0.4590	0.3337	0.7796	0.5693	0.1545	0.7573	0.3417	0.1292	0.7849	0.3379
	d_n	0.5683	1.6654	0.2651	1.4680	1.1137	0.5021	0.3406	0.8223	0.7608	0.1292	1.6655	1.7793
CAIWO	I_n	0.8996	0.7367	0.5923	0.4529	0.5664	0.9989	0.6111	0.7397	0.4340	0.5125	0.5364	0.9023
	d_n	0.5700	1.0866	1.2745	0.7590	1.0124	0.4897	0.5611	1.1737	1.3395	0.6553	1.0571	0.5562

result of four important merits: the average SLL, maximum SLL, null depth, and directivity in comparison to MIWO, IWO, PSO, DE, and GA for all instances.

4.2. Case Two. The goal of this case is to obtain the radiation pattern with minimum SLL and narrower beam width. This is done by manipulating the excitation currents and positions

of elements. This case is considered similar to that reported in literatures [7, 9, 10, 17]. The objective function to achieve the desired pattern using CAIWO is given as follows:

$$\text{Fitness} = a_1 |AF(\phi_{\text{MSLL}})| + a_2 BW, \quad (15)$$

where a_1 and a_2 are the weighting coefficients, BW is the half power beam width of the array pattern measured in degrees, and $AF(\phi_{\text{MSLL}})$ is the maximum SLL.

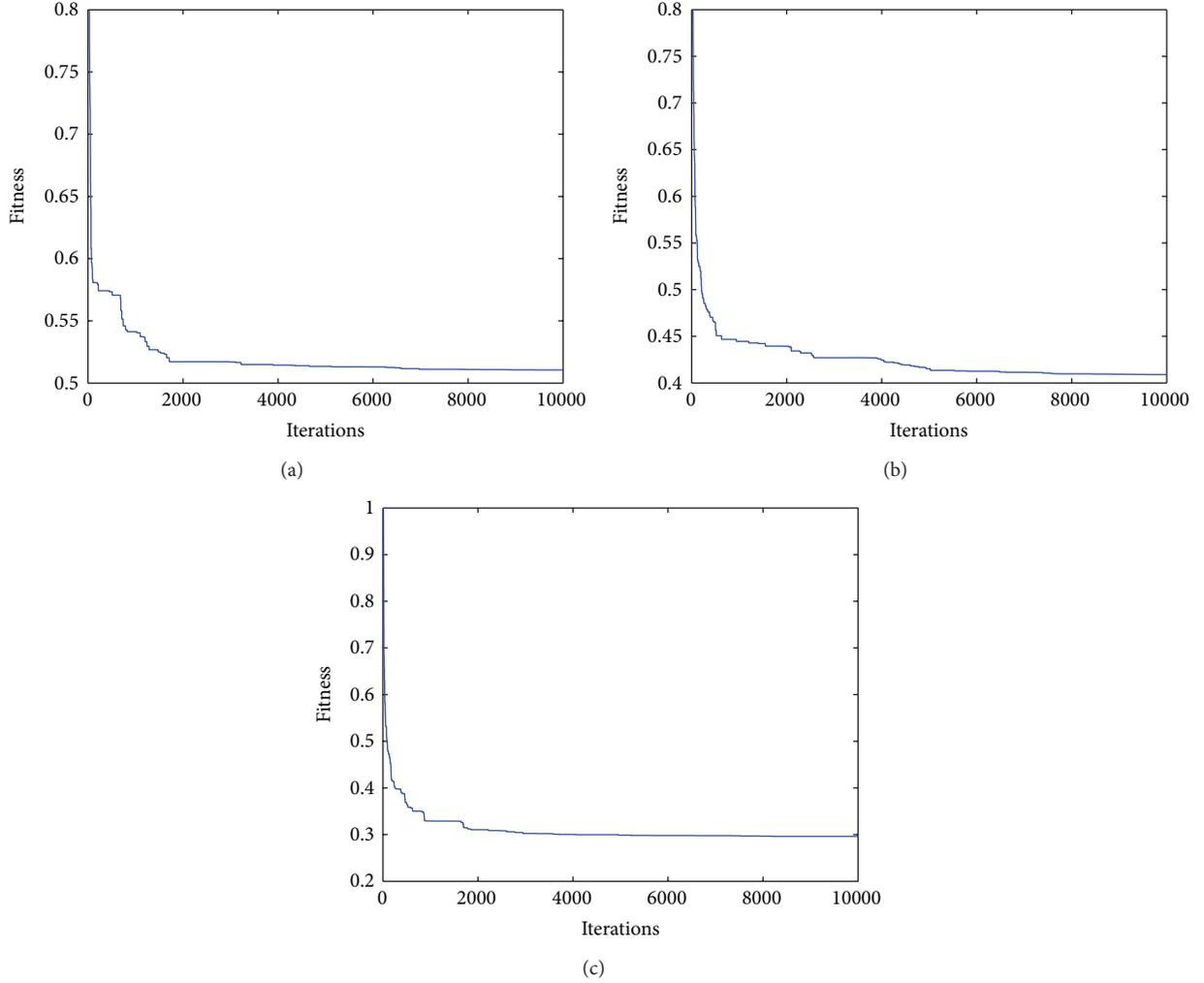


FIGURE 6: Convergence characteristic of six algorithms over three instances of the circular array design problem. (a) For number of elements $N = 8$. (b) For number of element $N = 10$. (c) For number of elements $N = 12$.

In order to reduce the mutual coupling effects between elements, an additional term is added in the objective function (15). The ratio between the maximum and minimum excitation amplitudes is used to minimize the coupling effect. The minimization of the amplitude-excitation dynamic range (ADR) can reduce the mutual coupling problem [29]. The objective function can be expressed as follows:

$$\text{Fitness}' = a_1 |AF(\phi_{\text{MSLL}})| + a_2 BW + \text{ADR}, \quad (16)$$

where ADR is the amplitude-dynamic ratio. The ADR is defined as the ratio between the maximum excitation amplitude and the minimum excitation amplitude. A small value of DRR not only can achieve a better control of the mutual coupling but also reduce the cost of the feeding network. The values of a_1 and a_2 are 70 and 1, respectively [7].

In this case, the values of the element amplitudes are allowed to vary between $[0, 1]$ and the separation between $[0, 2\lambda]$, where λ is the wavelength of the signal. The maximum number generation is 80. The values of the rest parameters

of CAIWO are same to those in Section 4.1. The stopping criterion for CAIWO is the maximum number of generations.

Along with the CAIWO results, the optimized array geometry and current excitation weights obtained using PSO [10], GA [9], SA [17], BBO [7], and IWO are also listed for comparisons in Tables 7, 8, and 9. In Table 6, the best results obtained for the aforesaid three problem instances are judged in terms of maximum SLL, HPBW, and ADR for all the six algorithms based approaches. Figure 7 presents the best radiation patterns of the circular arrays optimized by CAIWO compared to those of PSO, GA, SA, BBO, and IWO for 8, 10, and 12 element arrays.

In the 8 elements antenna array, the maximum SLL achieved by CAIWO is -13.0225 dB, the HPBW is 17.19° , and the ADR is 1.4974. Evidently, CAIWO provides better SLL, HPBW, and ADR than other techniques. The SLL obtained by CAIWO is lower by 0.7825, 1.0225, 3.2229, 3.2115 and 4.8558 dB than by the PSO, the GA, the SA, the BBO, and the IWO optimized arrays, respectively. The obtained HPBW is also narrower by 1.41, 9.42, 14.81, and 14.81° than the PSO, GA,

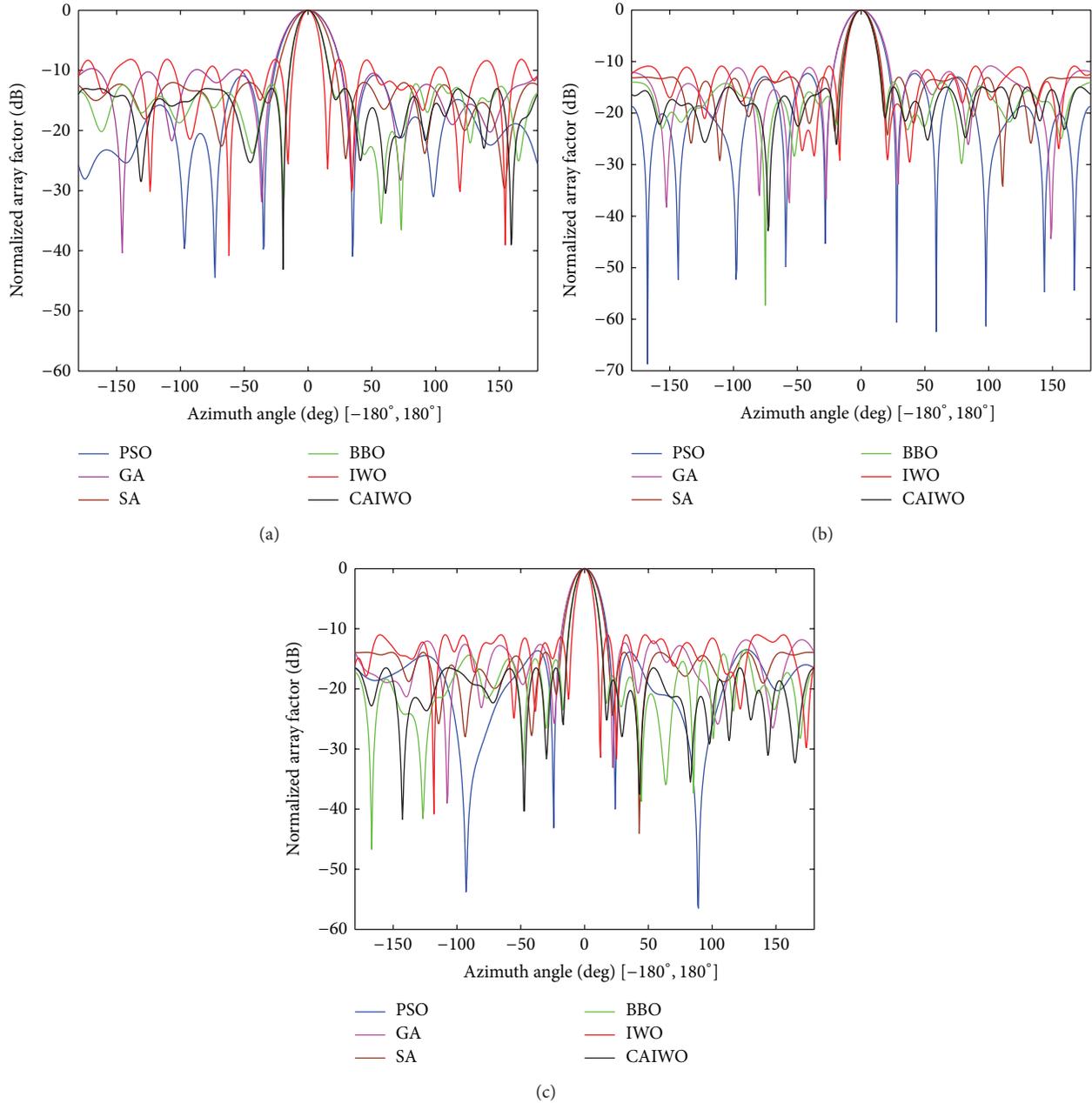


FIGURE 7: Normalized radiation patterns for circular arrays of different number of elements obtained using PSO GA SA BBO IWO and CAIWO. (a) For number of elements $N = 8$. (b) For number of element $N = 10$. (c) For number of elements $N = 12$.

SA, and BBO algorithms, respectively, except for the IWO algorithms. The ADR achieved by CAIWO is lower by 3.1605, 4.4621, 1.7409, 1.2936, and 2.4443 than by the IWO, BBO, SA, PSO, and GA algorithms. The best radiation pattern of the 8 elements array generated by CAIWO is plotted in Figure 7(a) along with that of the PSO, GA, SA, BBO, and IWO algorithms.

The maximum SLL of 10 elements array achieved by CAIWO is -14.96 dB, the HPBW is 17.19° , and the ADR is 1.9514. Evidently, CAIWO outperforms the other techniques. The SLL obtained by CAIWO is lower by 1.01, 1.96, 2.653, 5.149, and 4.0639 dB than by the PSO, the GA, the SA, the BBO, and

the IWO optimized arrays, respectively. Moreover, the HPBW obtained by CAIWO is also narrower by 0.56, 2.36, 8.3, and 15.96° than the PSO, GA, SA, and BBO algorithms, respectively, except for the IWO algorithms. The ADR achieved by CAIWO is lower by 2.9272, 0.6671, 1.2665, 0.0241, and 0.8626 than by the IWO, BBO, SA, PSO, and GA algorithms. The best radiation pattern of the 10 elements array generated by CAIWO is plotted in Figure 7(b). For comparisons, the radiation patterns of the antennas generated by the PSO, GA, SA, BBO, and IWO algorithms are also drawn in Figure 7(b). The radiation pattern shows that CAIWO obtains excellent results.

In the 12 elements antenna array, the maximum SLL achieved by CAIWO is -16.5 dB, the HPBW is 13.75° , and the ADR is 2.3018. Once again, the CAIWO algorithm yields results that are superior to the other algorithm. The maximum SLL is better than that achieved by the other algorithms. The reduction in SLL is significant and it is lower by 2.128, 2.59, 2.83, 4.67, and 5.4958 dB than by the PSO, the GA, the SA, the BBO, and the IWO optimized arrays, respectively. The obtained HPBW is also better than that attained by the other algorithms. It is narrower by 1.05, 5.85, 7.45, and 7.05° than by the PSO, GA, SA, and BBO algorithms, respectively, except for the IWO algorithms. The ADR achieved by CAIWO is lower by 5.3700, 0.9621, 0.2247, and 1.8421 than by the IWO, SA, PSO, and GA algorithms, except for BBO algorithm. The best radiation pattern of the 12 elements array generated by CAIWO PSO, GA, SA, BBO, and IWO algorithms is plotted in Figure 7(c). The directivity and circumferences of optimized array are also listed in Table 6. The obtained circumferences of CAIWO are bigger than those of other algorithms, but it achieves higher directivity. Certainly, CAIWO again outperforms the other algorithms in obtaining the required antennas.

5. Conclusions

This paper proposed a variant invasive weed optimization called CAIWO algorithm. The proposed algorithm takes advantages of chaotic search and modified IWO with adaptive dispersion, where the seeds produced by a weed are dispersed in the search space with standard deviation specified by the fitness value of the weed. The statistical results obtained from the four benchmark functions demonstrate the superiority of the proposed CAIWO algorithm to MIWO and classical IWO. In addition, numerical examples of circular antenna array synthesis problems have been presented. We formulated two design problems with different purposes. The first case is an optimization task on the basis of a cost fitness that takes care of the average side-lobe levels and the null control. The second case takes care of the maximum SLL reduction with the constraint on the beam width. The fitness functions of both cases are minimized under a constraint in order to avoid the mutual coupling between the array elements. The simulation results over different element number show that the CAIWO algorithm could comfortably outperform the above mentioned algorithms in circular antenna array synthesis.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] M. M. Khodier and C. G. Christodoulou, "Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 8, pp. 2674–2679, 2005.
- [2] D. Marcano and F. Durán, "Synthesis of antenna arrays using genetic algorithms," *IEEE Antennas and Propagation Magazine*, vol. 42, no. 3, pp. 12–20, 2000.
- [3] E. Rajo-Iglesias and O. Quevedo-Teruel, "Linear array synthesis using an ant-colony-optimization-based algorithm," *IEEE Antennas and Propagation Magazine*, vol. 49, no. 2, pp. 70–79, 2007.
- [4] K.-K. Yan and Y. Lu, "Sidelobe reduction in array-pattern synthesis using genetic algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 7, pp. 1117–1122, 1997.
- [5] A. Tennant, M. M. Dawoud, and A. P. Anderson, "Array pattern nulling by element position perturbations using a genetic algorithm," *Electronics Letters*, vol. 30, no. 3, pp. 174–176, 1994.
- [6] F. J. Ares-Pena, J. A. Rodriguez-Gonzalez, E. Villanueva-Lopez, and S. R. Rengarajan, "Genetic algorithms in the design and optimization of antenna array patterns," *IEEE Transactions on Antennas and Propagation*, vol. 47, no. 3, pp. 506–510, 1999.
- [7] U. Singh and T. S. Kamal, "Design of non-uniform circular antenna arrays using biogeography-based optimisation," *IET Microwaves, Antennas and Propagation*, vol. 5, no. 11, pp. 1365–1370, 2011.
- [8] C. A. Balanis, *Antenna Theory: Analysis and Design*, John Wiley & Sons, New York, NY, USA, 2012.
- [9] M. A. Panduro, A. L. Mendez, R. Dominguez, and G. Romero, "Design of non-uniform circular antenna arrays for side lobe reduction using the method of genetic algorithms," *AEU-International Journal of Electronics and Communications*, vol. 60, no. 10, pp. 713–717, 2006.
- [10] M. Shihab, Y. Najjar, N. Dib, and M. Khodier, "Design of non-uniform circular antenna arrays using particle swarm optimization," *Journal of Electrical Engineering*, vol. 59, no. 4, pp. 216–220, 2008.
- [11] M. A. Panduro, C. A. Brizuela, L. I. Balderas, and D. A. Acosta, "A comparison of genetic algorithms, particle swarm optimization and the differential evolution method for the design of scannable circular antenna arrays," *Progress in Electromagnetics Research B*, no. 13, pp. 171–186, 2009.
- [12] L. Gürel and Ö. Ergül, "Design and simulation of circular arrays of trapezoidal-tooth log-periodic antennas via genetic optimization," *Progress in Electromagnetics Research*, vol. 85, pp. 243–260, 2008.
- [13] G. G. Roy, S. Das, P. Chakraborty, and P. N. Suganthan, "Design of non-uniform circular antenna arrays using a modified invasive weed optimization algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 1, pp. 110–118, 2011.
- [14] A. Mandal, H. Zafar, S. Das, and A. V. Vasilakos, "Efficient circular array synthesis with a memetic differential evolution algorithm," *Progress in Electromagnetics Research B*, no. 38, pp. 367–385, 2012.
- [15] P. Ghosh, J. Banerjee, S. Das, and S. S. Chaudhury, "Design of non-uniform circular antenna arrays—an evolutionary algorithm based approach," *Progress in Electromagnetics Research B*, no. 43, pp. 333–354, 2012.
- [16] P. Civicioglu, "Circular antenna array design by using evolutionary search algorithms," *Progress in Electromagnetics Research B*, vol. 54, pp. 265–284, 2013.
- [17] M. Rattan, M. S. Patterh, and B. S. Sohi, "Optimization of circular antenna arrays of isotropic radiators using simulated annealing," *International Journal of Microwave and Wireless Technologies*, vol. 1, no. 5, pp. 441–446, 2009.

- [18] A. R. Mehrabian and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, vol. 1, no. 4, pp. 355–366, 2006.
- [19] A. Foudazi and A. R. Mallahzadeh, "Pattern synthesis for multi-feed reflector antennas using invasive weed optimisation," *IET Microwaves, Antennas and Propagation*, vol. 6, no. 14, pp. 1583–1589, 2012.
- [20] S. Karimkashi and A. A. Kishk, "Invasive weed optimization and its features in electromagnetics," *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 4, pp. 1269–1278, 2010.
- [21] S. H. Sedighy, A. R. Mallahzadeh, M. Soleimani, and J. Rashed-Mohassel, "Optimization of printed yagi antenna using invasive weed optimization (IWO)," *IEEE Antennas and Wireless Propagation Letters*, vol. 9, pp. 1275–1278, 2010.
- [22] Y.-Y. Bai, S. Xiao, C. Liu, and B.-Z. Wang, "A hybrid IWO/PSO algorithm for pattern synthesis of conformal phased arrays," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 4, pp. 2328–2332, 2013.
- [23] S. Karimkashi, A. A. Kishk, and D. Kajfez, "Antenna array optimization using dipole models for MIMO applications," *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 8, pp. 3112–3116, 2011.
- [24] A. R. Mallahzadeh, H. Oraizi, and Z. Davoodi-Rad, "Application of the invasive weed optimization technique for antenna configurations," *Progress in Electromagnetics Research*, vol. 79, pp. 137–150, 2008.
- [25] F. M. Monavar, N. Komjani, and P. Mousavi, "Application of invasive weed optimization to design a broadband patch antenna with symmetric radiation pattern," *IEEE Antennas and Wireless Propagation Letters*, vol. 10, pp. 1369–1372, 2011.
- [26] H. Wu and C. Li, "Planar array synthesis with sidelobe reduction and null control using invasive weed optimization," *Progress in Electromagnetics Research M*, vol. 33, pp. 83–94, 2013.
- [27] M. Ahmadi and H. Mojallali, "Chaotic invasive weed optimization algorithm with application to parameter estimation of chaotic systems," *Chaos, Solitons & Fractals*, vol. 45, no. 9-10, pp. 1108–1120, 2012.
- [28] P. N. Suganthan, N. Hansen, J. J. Liang et al., "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," KanGAL Report 2005005, 2005.
- [29] J. A. Rodriguez, F. Ares, and E. Moreno, "Linear array pattern synthesis optimizing array element excitations using the simulated annealing technique," *Microwave and Optical Technology Letters*, vol. 23, no. 4, pp. 224–226, 1999.

Research Article

Optimal Sensor Placement for Latticed Shell Structure Based on an Improved Particle Swarm Optimization Algorithm

Xun Zhang,¹ Juelong Li,² Jianchun Xing,¹ Ping Wang,¹ Qiliang Yang,¹
Ronghao Wang,¹ and Can He¹

¹ College of Defense Engineering, PLA University of Science and Technology, Nanjing 21007, China

² Technical Management Office of Naval Defense Engineering, Beijing 100841, China

Correspondence should be addressed to Xun Zhang; xunzhang893@163.com

Received 19 February 2014; Revised 22 May 2014; Accepted 22 May 2014; Published 16 June 2014

Academic Editor: Fang Zong

Copyright © 2014 Xun Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Optimal sensor placement is a key issue in the structural health monitoring of large-scale structures. However, some aspects in existing approaches require improvement, such as the empirical and unreliable selection of mode and sensor numbers and time-consuming computation. A novel improved particle swarm optimization (IPSO) algorithm is proposed to address these problems. The approach firstly employs the cumulative effective modal mass participation ratio to select mode number. Three strategies are then adopted to improve the PSO algorithm. Finally, the IPSO algorithm is utilized to determine the optimal sensors number and configurations. A case study of a latticed shell model is implemented to verify the feasibility of the proposed algorithm and four different PSO algorithms. The effective independence method is also taken as a contrast experiment. The comparison results show that the optimal placement schemes obtained by the PSO algorithms are valid, and the proposed IPSO algorithm has better enhancement in convergence speed and precision.

1. Introduction

Many large-scale civil infrastructures have been established with the rapid development of engineering technology. However, these structures are easily damaged in their long service lives because of inadequate designs and increasing traffic loads, which may cause sudden destruction and large economic losses. Therefore, developing a continuous structural health monitoring (SHM) system is essential. The sensor system is an important subsystem of a typical SHM system. More information can generally be obtained with the increasing sensor number. Unfortunately, an excessive sensor number increases the testing cost, which also increases the SHM system cost. Installing sensors are also difficult in many parts of the structure. Thus, designing an optimal sensor scheme is a key issue. The optimal placement of sensors not only lowers the cost but also acquires reliable and comprehensive structural health information [1]. Therefore, how to determine the optimal number of sensors and where to arrange the limited sensors have become key research issues.

Over the past two decades, large quantities of optimal sensor placement (OSP) techniques and criteria have been proposed for modal parameter identification and health monitoring of structures. These techniques can be divided into two categories, namely, traditional iterative approaches and global optimization techniques. The information-theory-based approaches are the important parts of the traditional iterative approaches. Kammer [2], Udawadia [3], Kammer and Tinker [4], and Meo and Zumpano [5] have all proposed information-theory-based approaches to select the optimal sensor locations. In these methods, the optimal sensor set is selected by maximizing the determinant of the Fisher information matrix. Papadimitriou [6, 7] used the information entry as a criterion to determine the optimal sensor configuration. Trendafilova et al. [8] addressed the concept of average mutual information to identify the optimal sensor locations. The matrix-decomposition-based approaches are also applied to OSP. Schedlinski and Link [9] introduced QR-decomposition of the modal matrix to determine the optimal sensor locations. Park and Kim [10], Cherg [11], and Reynier

and Abou-Kandil [12] have all adopted singular value decomposition methods to place sensors at optimal locations. Other iterative methods, such as the Guyan reduction [13], modal kinetic energy [14, 15], eigenvector component product [16], and drive point residue [17], have also been presented to solve the OSP problem. In recent years, with the rapid development of computational intelligence approaches, some global optimization techniques such as genetic algorithms (GAs) [18–24] and particle swarm optimization (PSO) algorithms [25–27] have been employed to determine the OSP. Given their advantages over traditional iterative methods such as global search, efficient parallel, and robustness, global optimization algorithms have played an important role in solving the OSP problem.

The above approaches could be applied to solve practical OSP problems and have already made great progress. However, some defects still exist in these methods. Firstly, the basic data selection for OSP is empirical. Generally, the mode shape matrix attained through modal analysis is always used for solving the OSP problem. Considering that the mode shape matrix is composed of modes, different mode shape matrices can be obtained with different mode numbers. This scenario leads to different sensor placement results. Therefore, how to select the modes is very important. In the previous methods, the lower modes were considered to have larger contributions to the dynamic response and always selected based on experience for solving the OSP problem. This mode selection method is unreasonable to a certain extent. Secondly, determining the optimal sensor number for specific structural forms is difficult. In general, the sensor number is predefined to a desired one or is determined based on the relationship between the fitness value and different sensor numbers. However, the predefined sensor number is subjective. The fitness value used for determining the sensor number is also computed only once. This condition will result in unreliable and imprecise computation results. Finally, the effective performance of the optimal placement techniques may not be assured. In traditional approaches, the computation of the desired sensor locations is an iterative process that usually obtains the suboptimal rather than the optimal value. Thus, this condition expands the errors between the real and estimated modal parameters identified by the placed sensors of these techniques. For the global optimization techniques, some features of the GA-based algorithms still require improvement. For example, it takes more computation time for the GAs to search for the best sensor locations because of the computational complexity. In view of this, Yao et al. [18] lowered the number of possible candidate sensor locations before applying GA to limit the complexity. Rao and Anandakumar [25] claimed that the GA performance for the OSP is difficult to assure if the problems have large candidate sensor configurations. Thus, they proposed a hybrid swarm intelligence technique to solve the OSP problem without limiting the number of possible candidate sensor configurations. Lian et al. [26] also used a hybrid swarm intelligence algorithm for OSP. The optimized results show that the proposed algorithm outperforms a GA in the capability of global optimization.

Concerning the existing problems and advantages of the swarm intelligence algorithm over GA, this paper proposes a type of improved particle swarm optimization (IPSO) algorithm to solve the OSP problem better. The cumulative effective modal mass participation ratio is firstly proposed to select the reasonable number of modes. Three strategies are then used to improve the PSO algorithm, including the dual-structure coding, novel inertia weight adjustment method, and mutation operator. The root mean square value of the off-diagonal elements in the modal assurance criterion (MAC) matrix is also taken as the fitness function. Finally, the IPSO algorithm is employed to solve the OSP problem of a latticed shell structure. The simulation experiments compared with the effective independence (EI) method and four different PSO algorithms variants are also conducted to verify the performance of the proposed algorithm.

The remaining part of this paper is organized as follows. Section 2 introduces the background of the OSP problem. Section 3 presents the OSP implementation steps based on the proposed IPSO algorithm. Section 4 shows the performance of the proposed IPSO algorithm for optimizing OSP in a latticed shell structure. Section 5 is some conclusions and the future work.

2. Background

Placing sensors at every degree of freedom (DOF) for structures is impossible because of the high cost of sensors, data acquisition system, and data processing system. The goal of OSP is to acquire the maximum structural health information by placing as few sensors as possible and make sure that the sensor locations are the best. Suppose the number of nodes in a structure is m , the numbered nodes compose a set N , where $N = \{n_1, n_2, \dots, n_m\}$, the theoretical DOF of n_i is x_i ($i = 1, 2, \dots, m$), and the DOFs of all nodes compose a set $X = \{x_1, x_2, \dots, x_m\}$. Given the constraints of the installation process, x_1, x_2, \dots, x_s ($s \leq m$) of the set X are selected as the candidate sensor locations. The goal is to search for the optimal ones from the candidates. Thus, OSP can be presented as an optimal problem that consists of the objective function and constraints as follows:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & x = (x_1, x_2, \dots, x_s) \subset X, \end{aligned} \quad (1)$$

where $f(x)$ is the objective function with the decision variables, x_1, x_2, \dots, x_s . This condition is a minimum optimization problem. The key is to find the minimum value of f with respect to the decision variables, where x_j ($x_j \in x$) is one of them. If $x_j = 1$, then a sensor is placed at the j -th DOF; if $x_j = 0$, no sensor exists. Once x_j is confirmed, the locations of the sensors can thus be determined.

The PSO algorithm is a swarm intelligent computation technique based on a simple simulation of bird flocking, fish schooling, and swarming theory. Unlike other evolutionary computation techniques, the PSO algorithm is simple, easy to implement, and computationally efficient. This algorithm has attracted much attention since its first introduction in 1995 [28, 29]. The PSO algorithm also has science and engineering applications. In PSO, each solution point in the search space is

known as a particle. The particle can find the optimal position through its own experience and social cooperation with its neighbors. All particles are assigned a random velocity. The particles fly through the problem space with a certain velocity to reach better search areas. Suppose each particle searches for the optimal value in the D -dimensional problem space, $X_i = (X_{i1}, X_{i2}, \dots, X_{iD})$ is the position of the i th particle, $V_i = (V_{i1}, V_{i2}, \dots, V_{iD})$ represents the velocity of the particle, which is clamped to a maximum magnitude " V_{\max} " specified by the problem to be solved, $P_i = (P_{i1}, P_{i2}, \dots, P_{iD})$ is the best previous position for the i th particle, and $P_g = (P_{g1}, P_{g2}, \dots, P_{gD})$ is the best previous position of the swarm. The velocity V_{id} and position X_{id} update equations are given as follows:

$$\begin{aligned} V_{id}^{t+1} &= V_{id}^t + c_1 r_{1id} (P_{id} - X_{id}^t) + c_2 r_{2id} (P_{gd} - X_{id}^t) \\ X_{id}^{t+1} &= X_{id}^t + V_{id}^{t+1}, \end{aligned} \quad (2)$$

where c_1 and c_2 are the acceleration coefficients, which are normally taken as 2, t is the current iteration number, d is the current dimensional problem space, and r_1 and r_2 are independently uniformly distributed random numbers in the range of $[0, 1]$.

3. Proposed IPSO Algorithm for OSP

In this section, a new approach called IPSO algorithm is proposed to solve the existing sensor placement problems in traditional methods. Three steps are included in this approach, which can be seen in Figure 1. The mode number is initially selected based on the cumulative effective modal mass participation ratio. A novel IPSO algorithm is also proposed based on three different strategies. Finally, this proposed IPSO algorithm is applied to determine the optimal sensor number and configurations.

3.1. Mode Selection. Modal analysis can provide the original data for OSP (i.e., mode shape matrix), which is composed of modes. How to select mode number is very important. If the number is different, the same optimization algorithm obtains different location schemes. The computation is also time-consuming given an excessive number of modes. Thus, selecting proper modes before using the placement method to determine the sensor locations is essential. In this section, the effective modal mass participation ratio is employed to determine the main modes, and the cumulative effective modal mass participation ratio is adopted to decide the proper mode number. The computational method on the selection of modes is studied using the following mathematical derivation.

The basic motion equation of a general n -DOF system subjected to an external force can be described as follows:

$$M\ddot{u} + C\dot{u} + Ku = -Me\ddot{u}_a(t), \quad (3)$$

where n is the number of DOFs; M is the mass matrix; C is the damping matrix; K is the stiffness matrix; e is the direction matrix of the external force; $\ddot{u}_a(t)$ is the acceleration impact

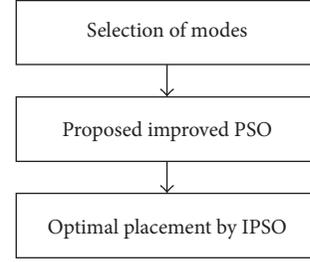


FIGURE 1: Steps of the IPSO method for OSP.

generated by the external force; and \ddot{u} , \dot{u} , and u are the acceleration, velocity, and displacement response, respectively.

Using the mode shape matrix Φ to express the structural response, the following values are obtained:

$$u = \Phi q, \quad \dot{u} = \Phi \dot{q}, \quad \ddot{u} = \Phi \ddot{q}. \quad (4)$$

Considering the orthogonal conditions of M , C , and K , as well as combining (3) and (4), we can obtain the following:

$$\ddot{q}_i + 2\zeta_i w_i \dot{q}_i + w_i^2 q_i = -\frac{\phi_i^T M e}{\phi_i^T M \phi_i} \ddot{u}_a(t), \quad (5)$$

where q_i is the modal coordinate, w_i is the natural frequency, ζ_i is the modal damping ratio associated with the i th mode, and ϕ_i is the i th column vector in the matrix Φ .

The modal participation factor can be defined as follows:

$$\gamma_i = \frac{\phi_i^T M e}{\phi_i^T M \phi_i}. \quad (6)$$

Normalizing the modal mass by the following equation:

$$\phi_i^T M \phi_i = 1, \quad (7)$$

the modal participation factor can be expressed as follows:

$$\gamma_i = \phi_i^T M e. \quad (8)$$

We define the effective modal mass of the i th mode as follows:

$$M_i = \frac{\gamma_i^2}{\phi_i^T M \phi_i}. \quad (9)$$

Considering (7), we can obtain $M_i = \gamma_i^2$, and the total mass can be derived from the following:

$$\sum_{i=1}^n M_i = \sum_{i=1}^n \gamma_i^2 = \gamma^T \gamma. \quad (10)$$

From (8), we can obtain $\gamma = \Phi^T M e$. Substituting this equation into (10), we can reach the following equation:

$$\gamma^T \gamma = (\Phi^T M e)^T (\Phi^T M e) = (e^T M \Phi) (\Phi^T M e). \quad (11)$$

The matrix form of (7) can be expressed as follows:

$$I = \Phi^T M \Phi = (\Phi^T M \Phi)^{-1}, \quad (12)$$

where I is a unit matrix. Substituting (12) into (11), we can obtain the following:

$$\begin{aligned}
 \gamma^T \gamma &= (e^T M \Phi) (\Phi^T M e) \\
 &= (e^T M \Phi) (\Phi^T M \Phi)^{-1} (\Phi^T M e) \\
 &= e^T M \Phi \Phi^{-1} (\Phi^T M)^{-1} (\Phi^T M) e \\
 &= e^T M e \\
 &= m_1 + m_2 + \cdots + m_n \\
 &= M_T.
 \end{aligned} \tag{13}$$

The above equation indicates that the total effective modal mass is equal to the total mass or the rotational inertia. Thus, the i th effective modal mass is one of the important parameters that can reflect the dynamic response of the given modes. It can be used to determine the mode number.

We define the ratio of effective modal mass to total effective modal mass as the effective modal mass participation ratio. The effective modal mass participation ratio with respect to the i th mode can be presented as follows:

$$r_i = \frac{M_i}{e^T M e}. \tag{14}$$

In practical calculation, if the number of modes is k ($k < n$), then the cumulative effective modal mass participation ratio can be expressed as follows:

$$R = \sum_{i=1}^k r_i. \tag{15}$$

The above equation reflects the effective modal mass participation ratio of the selected modes. The R value should be larger than 90% (i.e., $R \geq 0.9$) to obtain a sufficient number of main modes [30]. This criterion is applied to select the mode number.

3.2. OSP Using the IPSO Algorithm. The proposed novel IPSO algorithm is designed to solve the OSP problem. The flow chart of the designed algorithm is presented in Figure 2. Firstly, the population is initialized and the values of P_i and P_g are calculated. The parameters are updated, and the fitness values are evaluated. The mutation operator is employed if the fitness value does not improve over the iterations; P_i and P_g are updated otherwise. Finally, the second step is repeated until the termination condition is satisfied.

3.2.1. Initialization. Initializing position and velocity of the particle are two basic aspects. As the goal of the algorithm is to determine the optimal fitness value with respect to the particle's position, the position initialization directly determines the search result. Thus, the computation performance of the algorithm is greatly affected by the position. The initialization of the particle's position is thus mainly introduced as follows. However, the velocity is randomly initialized in the search space.

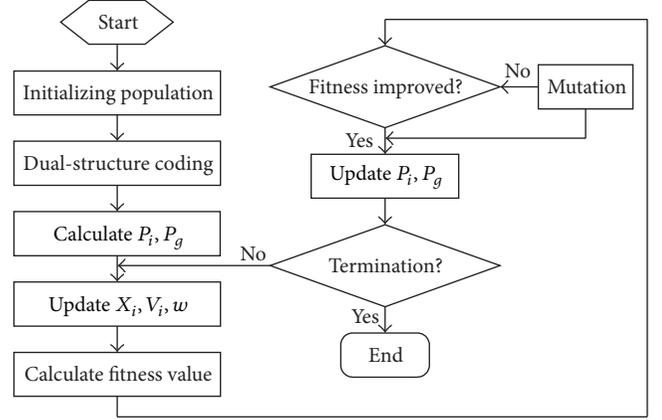


FIGURE 2: Flow chart of the proposed IPSO algorithm.

TABLE 1: A new particle position generated by dual-structure coding method.

Extra code	$P(1)$	$P(2)$	\cdots	$P(i)$	\cdots	$P(s)$
Variable code	$K_{P(1)}$	$K_{P(2)}$	\cdots	$K_{P(i)}$	\cdots	$K_{P(s)}$

TABLE 2: Example of a dual-structure coding method.

2	3	1	5	7	4	10	6	9	8
0	1	1	0	0	0	1	0	1	1

The position initialization is assumed to be a coding problem. The 1D binary coding method can be used to initialize the particle's position. The length of the binary string should be the same as the candidate sensor locations. If the value of the i th code is 1, a sensor is placed on the i th DOF. Otherwise, no sensor exists. This coding method is simple, intuitional, and easy to implement. However, the sensor number changes if 1 is changed in the particle's position code. Thus, this coding method is improper for solving the OSP problem. A dual-structure coding method is adopted in [31]. This method can overcome the shortcomings of binary coding. The dual-structure coding method is thus employed in the particles' position encoding of the PSO algorithm. The specific method can be described as follows.

The upper line is the extra code, which is generated randomly through the shuffle method firstly. The variable code is then produced randomly (with a value of 0 or 1) and placed on the lower line. Thus, the extra and variable codes compose the individual position. A particle's position encoded by the dual-structure method can be seen in Table 1. The upper line $P(i)$ is the extra code of K_j , where $P(i) = j$ and the length of extra code is equal to the length of the candidate sensor locations (i.e., s in this paper). The lower line denotes the variable code $K_{P(i)}$ corresponding to the extra code $P(i)$. When decoding the variable code, the constraints should be considered, and the penalty function is used. If a variable does not follow the constraints, its corresponding value is enforced to be 0 or 1 if otherwise.

For example, an OSP problem with 10 candidate sensor locations and the randomly generated sequence of extra code

is (2, 3, 1, 5, 7, 4, 10, 6, 9, and 8). Its dual-structure code is presented in Table 2. In this manner, the code corresponds to a feasible solution (i.e., the 1st, 3rd, 8th, 9th, and 10th DOFs are selected as the sensor locations).

3.2.2. Improved Inertia Weight Adjustment Strategy. The PSO algorithm shows significant performance in the initial iterations, but it might encounter local optimum in the latter iterations. Many researchers have worked on improving its performance by dynamically changing the PSO parameters during the iterations. Shi and Eberhart [32] introduced an inertia weight w into the original PSO algorithm to improve its performance. This parameter can provide a flexible influence of the previous velocity on the new velocity. This modified PSO algorithm is a standard PSO (SPSO). The velocity updating of the d -th dimension of the i th particle in the SPSO algorithm is presented as follows:

$$V_{id}^{t+1} = wV_{id}^t + c_1r_{1id}(P_{id} - X_{id}^t) + c_2r_{2id}(P_{gd} - X_{id}^t). \quad (16)$$

The inertia weight w reflects the particle's dynamic behavior. A higher value is good for globally exploring a better solution. However, a lower inertia weight could provide slower updating in exploring locally. A linear inertia weight adjustment strategy is proposed in [33] to balance the global and local search capabilities. Thus,

$$w(t) = w_{\max} - (w_{\max} - w_{\min}) \frac{t}{t_{\max}}, \quad (17)$$

where w_{\max} is the maximum value of the w , w_{\min} is the minimum value of the w , t_{\max} is the maximum number of allowable iterations, and t is the current number of iterations. Shi and Eberhart [33] conducted empirical studies and found that the optimal value can be improved when the value of w is linearly decreased in the range of [0.9, 0.4]. This PSO is thus referred to as linearly decreasing inertia weight PSO (LPSO).

Eberhart and Shi [34] applied PSO algorithm to track and optimize dynamic systems. However, they found that the improved LPSO algorithm is not very effective in tracking dynamic systems. Instead, they took the dynamic nature of real-world application into consideration and proposed a random inertia weight factor for tracking dynamic systems. The difference between this method and LPSO is that the inertia weight w changes randomly as follows:

$$w(t) = 0.5 + \frac{r_3}{2}, \quad (18)$$

where r_3 is the same as r_1 or r_2 (i.e., it is a random number in the range of [0, 1]). For the remainder of this paper, this PSO algorithm is referred to as random inertia weight PSO (RPSO).

A nonlinear time-varying inertia weight strategy was proposed in [35]. The inertia weight nonlinearly decreased from a high initial value w_{\max} to w_{\min} . The main difference between this method and LPSO is the following inertia weight decreasing strategy:

$$w(t) = w_{\min} + \left(\frac{t_{\max} - t}{t_{\max}} \right)^\alpha (w_{\max} - w_{\min}), \quad (19)$$

where w_{\max} is the maximum value of the w , w_{\min} is the minimum value of the w , t_{\max} is the maximum number of allowable iterations, t is the current number of iterations, and α is the nonlinear modulation exponent. The proper choice of exponent α is one of the important factors in successfully implementing this algorithm. $\alpha = 1.2$ has shown encouraging results from several benchmark simulations. This algorithm with $\alpha = 1.2$ is referred to as nonlinearly decreasing inertia weight PSO (NPSO).

Given the results in [35], we developed a PSO algorithm based on the nonlinearly decreasing inertia weight strategy. However, unlike the NPSO method in [35], the proposed inertia weight nonlinearly decreases from w_{\max} to w_{\min} based on the cosine function feature. The nonlinear inertia weight strategy can be mathematically presented as follows:

$$w(t) = w_{\min} + \left(\frac{1 + \cos(t\pi/t_{\max})}{2} \right)^k (w_{\max} - w_{\min}), \quad (20)$$

where w_{\max} is the maximum value of the w , w_{\min} is the minimum value of the w , t_{\max} is the maximum number of allowable iterations, t is the current number of iterations, and k is a positive constant, which can adjust the slope of the decreasing curve of the inertia weight. The decreasing curves with different k value can be seen in Figure 3.

The k value is important in determining the decreasing pattern of the inertia weight. When k is less than 1.0, the inertia weight decreases according to the convex function; when k is larger than 1.0, the inertia weight strategy is based on the convex and concave strategies. The inertia weight value is higher in the early iterations and quickly decreases during the latter iterations compared with the linearly decreasing strategy. The major consideration of this improvement is avoiding premature convergence in the early part of the search and enhancing convergence to the global optimum in the latter part. As mentioned above, a higher inertia weight in the early stages and a lower inertia weight during the latter stages can meet this demand. Through many simulation experiments, the algorithm shows better convergence performance with the increasing k value. We only considered the proposed nonlinearly decreasing strategy with $k = 10$ to take advantage of this proposed strategy and simplify this problem. In the remainder of this paper, this proposed PSO algorithm is referred to as the improved PSO (IPSO).

3.2.3. Mutation Operator. The fast decreasing inertia weight during the latter optimization stages can enhance the convergence rate. However, this condition may result in the lack of population diversity and may rush to a local optimum solution. Attempts have been made in the literature to devise the behavior of the particles in the latter range of the search space and improve the population diversity. Therefore, the concept of "mutation" has been observed to enhance the PSO algorithm performance in [36].

According to this new strategy, a particle is firstly picked randomly, and then a random perturbation is added to the velocity of that selected particle by a predefined mutation probability. In this study, the mutation operator is obtained using the strategy proposed in [36], which set the random

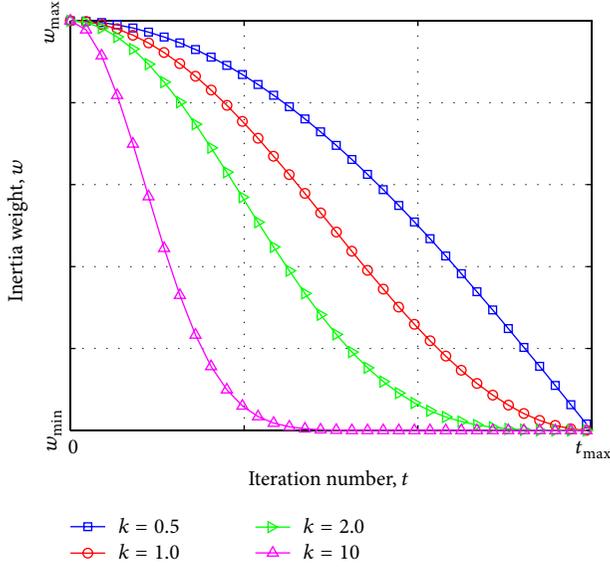


FIGURE 3: The cures of inertia weight with different value of k .

```

if ( $\Delta$ global  $\leq 0$ )
  if ( $\text{rand}_1(\cdot) < P_m$ )
    if ( $\text{rand}_2(\cdot) < 0.5$ )
       $v_{kl} = v_{kl} + \text{rand}_3(\cdot) * V_{\max}/m$ ;
    else
       $v_{kl} = v_{kl} - \text{rand}_4(\cdot) * V_{\max}/m$ ;
    end if
  end if
end if

```

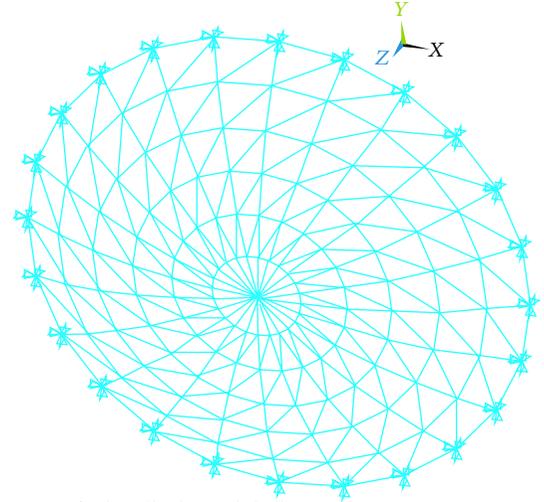
ALGORITHM 1

perturbation proportional to the maximum allowable velocity. The chief pseudocode for the mutation strategy is as in Algorithm 1, where Δ global is the divergence of the global solution over the adjacent two generations, $\text{rand}_i(\cdot)$, $i = 1, 2, \dots, 4$, are independently uniformly distributed random numbers in the range of $[0, 1]$, P_m is the mutation probability, k and l are random constants, and m is a constant, which can control the mutation step size.

Through numerical simulations, the parameters setting of the PSO algorithm with the mutation strategy are supplied in [36]. In this investigation, the same parameter setting is used. In particular, P_m is set to 0.4, and the mutation step size is set to linearly decrease from V_{\max} to $0.1V_{\max}$ during the search space.

3.2.4. Fitness Function. The quality of the new individuals is judged by the fitness function value in the optimization algorithm. For the PSO algorithm, the fitness value in the search space is used to evaluate the solution directly. Thus, the fitness function is very important for the PSO algorithm and should be selected properly.

The measured mode shape vectors should be linearly independent as much as possible to distinguish the measured modes. Carne and Dohrmann [37] pointed that the MAC



FEM of Schwedler latticed shell

FIGURE 4: Finite element model of the latticed shell.

is a good tool to evaluate this linear dependence. The MAC value compares the direction of the two vectors. When the MAC value is relatively small, the two vectors are easily distinguishable. Thus, the off-diagonal terms of the MAC matrix can be used to check the linear independence of the mode shapes. The MAC matrix elements can be expressed as follows:

$$\text{MAC}_{ij} = \frac{(\phi_i^T \phi_j)^2}{(\phi_i^T \phi_i)(\phi_j^T \phi_j)}, \quad (21)$$

where ϕ_i and ϕ_j are the mode shape vectors for the i th and j -th modes.

The root-mean-square (RMS) value reflects the magnitude and size of a data group. Therefore, the RMS value of the whole off-diagonal elements in the MAC matrix can be utilized to construct the fitness function. The equation of the optimization objective function [38] can be presented as follows:

$$f = \sqrt{\frac{1}{n(n-1)} \sum_{i=1}^n \sum_{j=1}^n \text{MAC}_{ij}^2}, \quad (i \neq j), \quad (22)$$

where n is the MAC matrix dimension. Thus, the function indicates the change in the off-diagonal elements in the MAC matrix, and the value should be as small as possible.

4. Sensor Placement Simulation

In this section, a numerical experiment is conducted to test and validate the proposed IPSO algorithm for the OSP problem. The EI method and five other PSO algorithms in Section 3.2.2 are also used for comparison.

4.1. Modeling and Modal Analysis. The considered numerical example is a single layer Schwedler latticed shell. The total span of the latticed shell is 100 m and the rise is 6.7 m. The

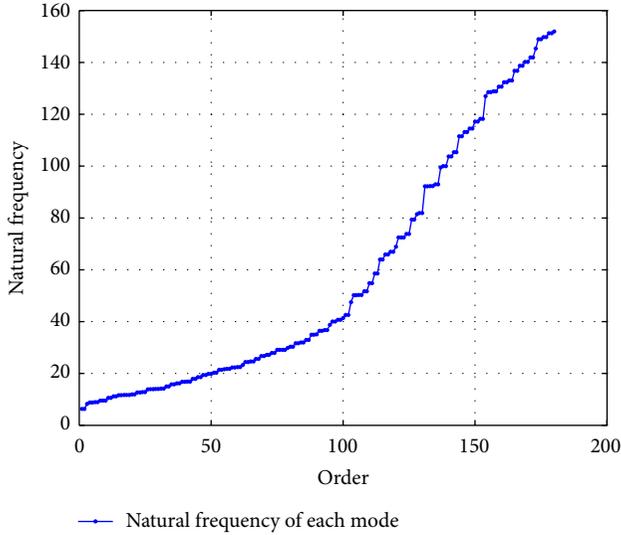


FIGURE 5: Natural frequency with respect to each mode.

grid numbers in the circumferential and radial directions are 24 and 6, respectively. All elements are tubular, and the specifications are $\emptyset 92.5 \text{ mm} \times 2.5 \text{ mm}$ and $\emptyset 83 \text{ mm} \times 3 \text{ mm}$. The material properties are obtained from Q235 steel, the elastic modulus of the material is 210 GPa, and the mass density is 7850 kg/m^3 . The shell is clamped on a fixed-point at the rim of the shell. The total element, node, and candidate DOF numbers are 408, 145, and 363, respectively. The finite element model (FEM) of the Schwedler latticed shell is created using the finite element analysis package (ANSYS [39]) to obtain the mode shape matrix. The obtained FEM of the latticed shell is shown in Figure 4. The structural dynamic characteristics, including the natural frequencies and normalized mode shapes, are also calculated by performing modal analysis based on the subspace iteration method. The natural frequency with respect to each order mode is shown in Figure 5. The first seven mode shapes and corresponding frequencies are shown in Figure 6.

4.2. Mode Number. In Figures 5 and 6, the distribution of the natural frequencies is very dense, and the mode shapes are complicated in the lower order modes. Some higher modes also have larger contributions to the vibration response of structure with different environmental incentives [40]. Therefore, selecting the main modes is essential. In this study, the cumulative effective modal mass participation ratio is used to determine the main contribution modes. As mentioned above, every effective modal mass participation ratio should be calculated until the cumulative effective modal mass participation ratio is larger than 90%. The main modes are then selected according to the larger ones. The selected main modes, effective modal mass, effective modal mass participation ratio, and cumulative effective modal mass participation ratio in the three directions (i.e., x , y , and z directions) are listed in Table 3.

As shown in Table 3, the 1st, 6th, 18th, and 80th modes are selected in the x direction; the 2nd, 7th, 17th, and 81st modes are selected in the y direction; and the 3rd, 8th, 19th, and 26th

modes are selected in the z direction. The cumulative effective modal mass participation ratios of the three directions are $R_x = 93.48\%$, $R_y = 93.48\%$, and $R_z = 91.38\%$, respectively, which are larger than the criterion at 90%. Thus, we considered that the selected mode number includes sufficient main modes.

4.3. Sensor Number. The sensor number is one of the key issues in OSP problems. In this section, the proposed IPSO algorithm is introduced to search for the best fitness value with respect to each sensor number. A circulating computation strategy is adopted to determine the optimal sensor number and enhance the precision and reliability of the computation result. With this strategy, the mean best fitness can be obtained. The sensor number is μ , and the mean best fitness value is f . Thus, the μ - f curve can be obtained, which indicates the relationship between the sensor number and the mean best fitness value. In this curve, the stable point demonstrates that the fitness does not change or only slightly changes with the increase in the sensor number. Therefore, the corresponding sensor number of this point is the economical one. This computation method's flow chart is shown in Figure 7.

For each computation, this method calculates the fitness value of each sensor. For each sensor, the IPSO algorithm repeats the maximum number of allowable iterations to search for the best fitness value. Therefore, this approach becomes complex and time-consuming if the maximum computation trial, sensor number, and allowable iterations are too large. We are also more concerned about the stable point of the μ - f curve than the calculating accuracy for the given problem. The maximum computation trial is set to 10, the maximum sensor number is set to 30, and the maximum number of allowable iterations is set to 100 so as to lower the computation complexity. The mutation probability $P_m = 0.4$, and mutation step size linearly decreases in the range of $[V_{\max}, 0.1V_{\max}]$. The acceleration coefficients $c_1 = c_2 = 2.0$, the population size $N = 40$, and the maximum velocity V_{\max} is equal to the candidate DOFs (i.e., $V_{\max} = 363$). The obtained μ - f curve for the 10 trials can be seen in Figure 8. The mean best fitness value clearly becomes smaller with the increase in the sensor number. This condition indicates that more information can be obtained with more sensors placed on the structures. The decreasing rate of the mean best fitness value becomes slow when the number of sensors increases to 20. Thus, we considered 20 as the stable point of the μ - f curve and set the economical sensor number to 20.

4.4. Results of the Contrast Experiments and Analysis. The comparison simulations of EI, SPSO, LPSO, NPSO, and RPSO algorithms are conducted to assess the proposed IPSO algorithm's performance. The inertia weight adjustment strategies of the five different PSO algorithms are set as follows. In SPSO, the inertia weight is fixed at 0.9. In LPSO, the inertia weight linearly decreases from 0.9 to 0.4. In NPSO, the inertia weight nonlinearly decreases from 0.2 to -0.3 with the slope $m = -2.5 \times 10^{-4}$ and exponent $\alpha = 1.2$. In RPSO, the inertia weight changes randomly according to (18). In the proposed IPSO, the inertia weight nonlinearly decreases from 0.9 to 0.4 with the exponent $k = 10$ as described

TABLE 3: The selected main modes, effective modal mass (EM), effective modal mass participation ratio (EMPR), and cumulative effective modal mass participation ratio (CEMR) in the three directions.

x direction			y direction			z direction					
Order	EM	EMPR	CEMR	Order	EM	EMPR	CEMR	Order	EM	EMPR	CEMR
1	1569.2	10.15%	10.15%	2	1569.2	10.15%	10.15%	3	1983.6	11.55%	11.55%
6	2043.4	13.22%	23.37%	7	2043.4	13.22%	23.37%	8	2800.4	16.31%	27.86%
18	781.19	5.05%	28.42%	17	781.21	5.05%	28.42%	19	10550	61.45%	89.31%
80	10056	65.06%	93.48%	81	10056	65.06%	93.48%	26	433.34	2.52%	91.83%

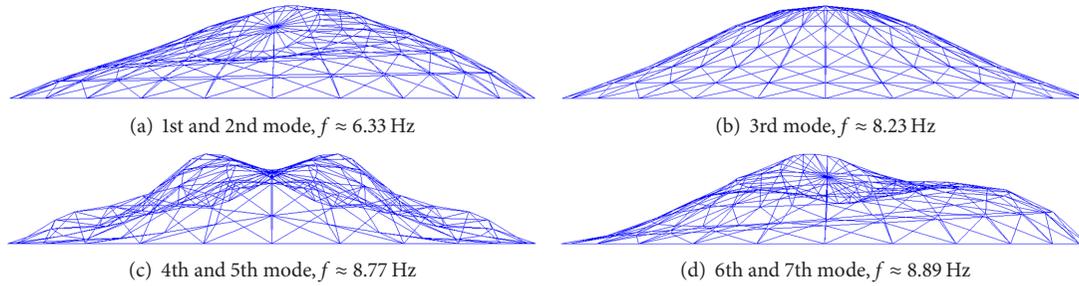


FIGURE 6: The mode shapes and corresponding frequencies of the Schwedler latticed shell.

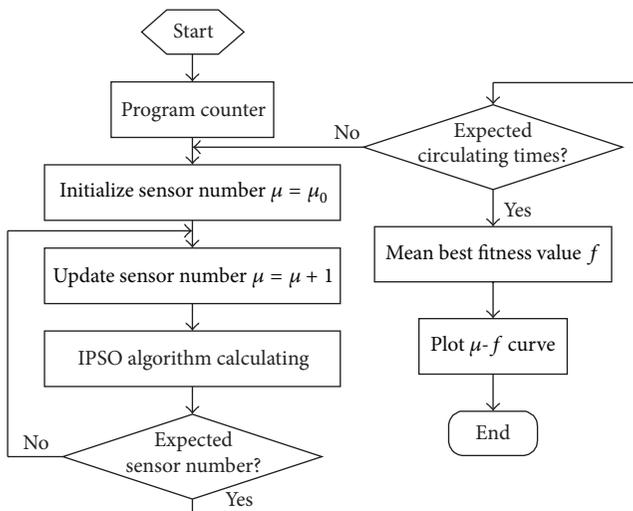


FIGURE 7: Flow chart of optimizing sensor number.

in (20). The maximum number of allowable iterations, t_{\max} , for the optimization algorithm is set to 4000 to ensure that the algorithms determine the optimal sensor placement and observe the variation of the fitness value with respect to the iterations. For a fair comparison, the other settings of the parameters are the same as mentioned above, including the mutation probability, acceleration coefficients, population size, and maximum velocity. The computation result of the EI method and the best fitness values of the five different PSO algorithms are listed in Table 4.

From the maximum off-diagonal elements (ODE_{\max}) of the MAC matrix presented in Table 4, the five PSO optimization algorithms can clearly minimize the ODE_{\max} . This result indicates that the PSO algorithms can be utilized to solve the

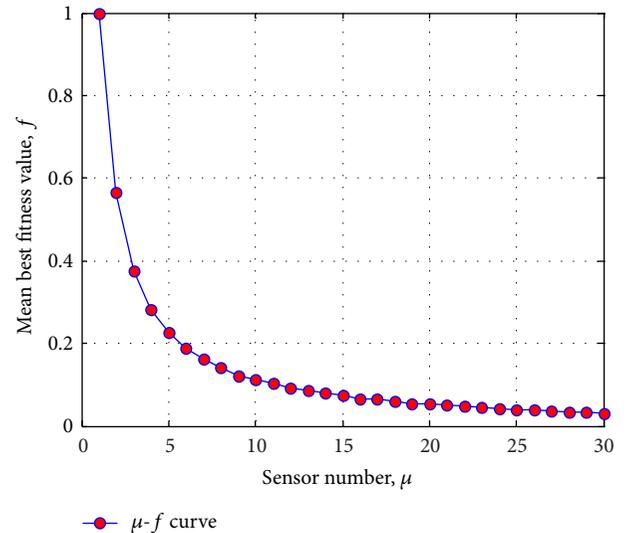
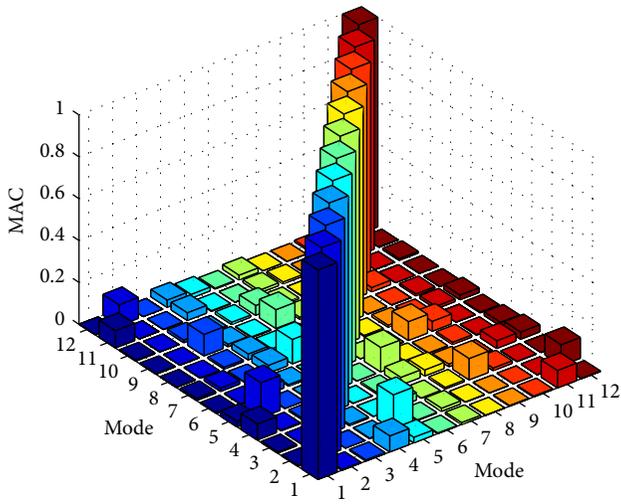


FIGURE 8: Relationship between the sensor number (μ) and the mean best fitness value (f).

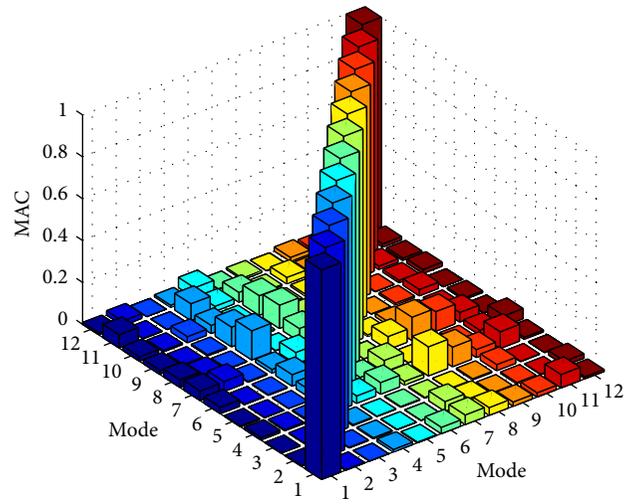
TABLE 4: Maximum values (ODE_{\max}) and root-mean-square values of off-diagonal elements (ODE_{rms}) of MAC matrix for 20 DOFs obtained by the six different methods.

Method	EI	SPSO	LPSO	RPSO	NPSO	IPSO
ODE_{\max}	0.1663	0.1382	0.1330	0.1281	0.1096	0.0963
ODE_{rms}	0.0377	0.0368	0.0415	0.0378	0.0328	0.0321

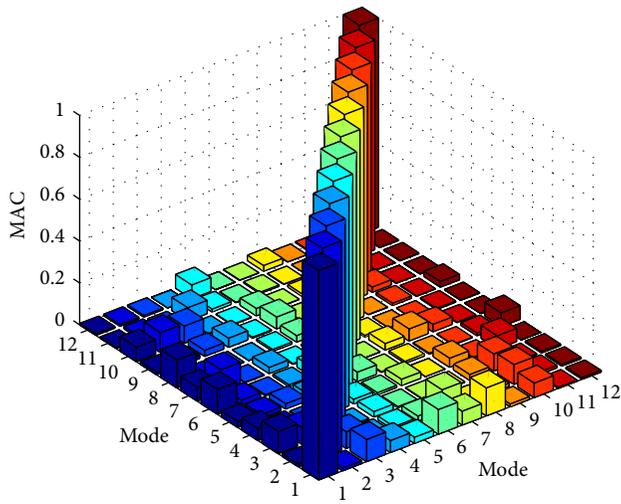
OSP problem. The RMS values of the off-diagonal elements (ODE_{rms}) are all close to each other, but the proposed IPSO algorithm can obtain the minimum value. This result verifies the proposed algorithm's superiority. The conclusions can



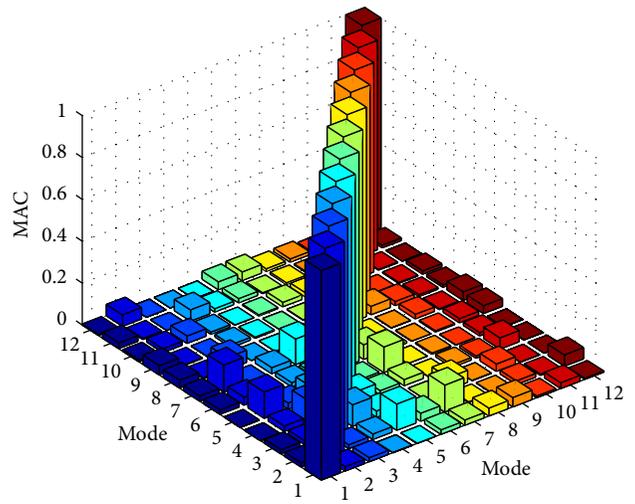
(a)



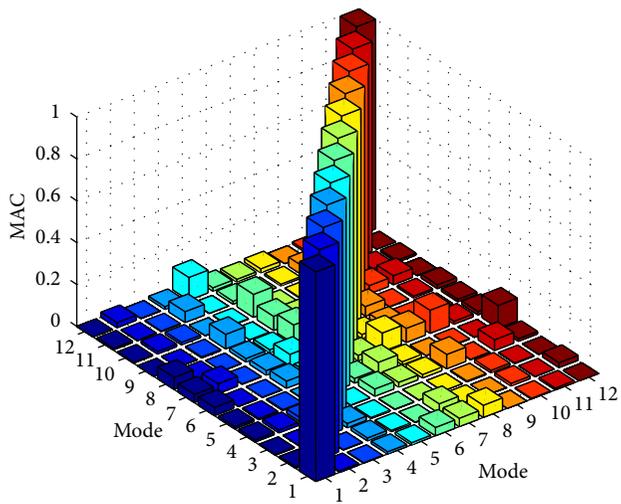
(b)



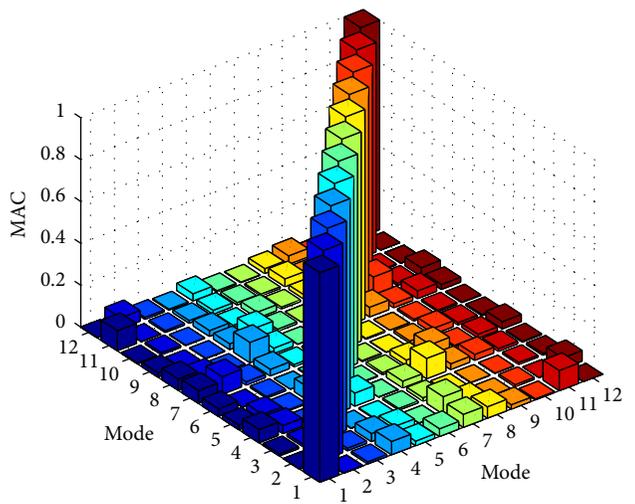
(c)



(d)



(e)



(f)

FIGURE 9: MAC values of 20 DOFs obtained by the six different methods. (a) EI method. (b) SPSO method. (c) LPSO method. (d) RPSO method. (e) NPSO method. (f) IPSO method.

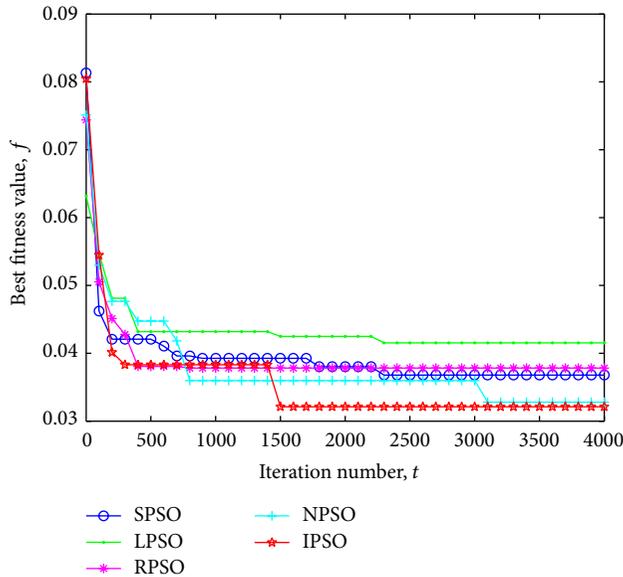


FIGURE 10: Performances of the five PSO algorithms.

also be reached from the distribution of the MAC values from the six different methods given in Figure 9. All of the off-diagonal elements are intuitively displayed in this figure. The proposed IPSO algorithm can obtain smaller values compared with the other five methods. In Figure 10, the best fitness values of the five PSO algorithms are compared. The IPSO algorithm can clearly determine the minimum fitness among these optimization algorithms. Finally, the placement results in three directions of the IPSO algorithm are given in Figure 11.

4.5. Discussion. The presentation case is based on the FEM of a latticed shell structure, which is created in strict accordance with the actual structure. Therefore, the validity of the proposed IPSO algorithm and the obtained optimal placement scheme can be verified by the comparison simulation results. However, this study still has some inevitable limitations. For example, the IPSO algorithm performance with different k values is not presented. Thus, the determination of k may be empirical to a certain extent. The FEM errors may also impact the placement results that are not considered in the simulation experiments. However, this problem could be solved if we consider a sensor placement experiment on a real latticed shell structure.

5. Conclusions and Future Work

In this paper, a novel IPSO approach is presented to address the existing defects of the traditional sensor placement methods. Firstly, a new method is proposed to select the mode number. Three strategies are then adopted to improve the PSO algorithm. Finally, the proposed IPSO approach is applied to determine the optimal sensor number and locations. With simulation experiments, some conclusions are summarized as follows.

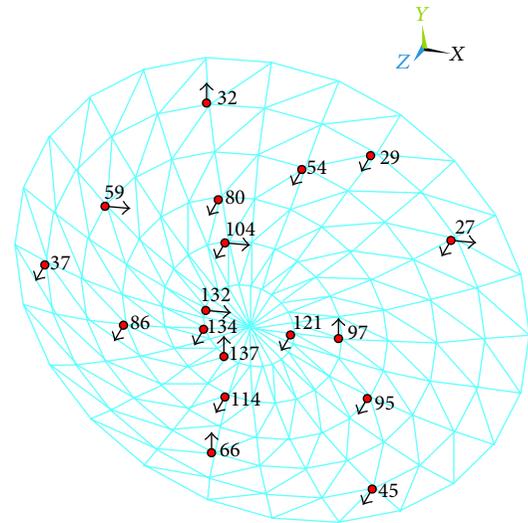


FIGURE 11: Optimal placement result of IPSO algorithm.

- (1) In selecting the proper modes, a new method called cumulative effective modal mass participation ratio is proposed, which can reflect the dynamic response of the given modes. Using this method, the sufficient main modes can be selected.
- (2) Strategies such as the dual-structure coding, novel nonlinear inertia weight adjustment method, and mutation operator can be utilized to improve the PSO algorithm's search ability.
- (3) The optimal sensor number is determined by a circulating computation strategy. This method calculates the mean best fitness value with the increase in sensor number. Therefore, the computation result is precise and reliable.
- (4) The contrast simulation results with the EI method and four different PSO algorithms for a latticed shell structure show that the proposed IPSO algorithm has better enhancement in convergence speed and precision.

The effectiveness of the IPSO approach has been proven in this study through simulation experiments. However, some aspects need to be further studied. The model errors should be investigated in future research.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] D. S. Shan, Z. H. Wan, and L. Qiao, "Optimal sensor placement for long-span railway steel truss cable-stayed bridge," in *Proceedings of the 3rd International Conference on Measuring Technology and Mechatronics Automation (ICMTMA '11)*, pp. 795–798, January 2011.

- [2] D. C. Kammer, "Sensor placement for on-orbit modal identification and correlation of large space structures," *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 2, pp. 251–259, 1991.
- [3] F. E. Udawadia, "Methodology for optimum sensor locations for parameter identification in dynamic systems," *Journal of Engineering Mechanics*, vol. 120, no. 2, pp. 368–390, 1994.
- [4] D. C. Kammer and M. L. Tinker, "Optimal placement of triaxial accelerometers for modal vibration tests," *Mechanical Systems and Signal Processing*, vol. 18, no. 1, pp. 29–41, 2004.
- [5] M. Meo and G. Zumpano, "On the optimal sensor placement techniques for a bridge structure," *Engineering Structures*, vol. 27, no. 10, pp. 1488–1497, 2005.
- [6] C. Papadimitriou, "Optimal sensor placement methodology for parametric identification of structural systems," *Journal of Sound and Vibration*, vol. 278, no. 4-5, pp. 923–947, 2004.
- [7] C. Papadimitriou, "Pareto optimal sensor locations for structural identification," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 12–16, pp. 1655–1673, 2005.
- [8] I. Trendafilova, W. Heylen, and H. Van Brussel, "Measurement point selection in damage detection using the mutual information concept," *Smart Materials and Structures*, vol. 10, no. 3, pp. 528–533, 2001.
- [9] C. Schedlinski and M. Link, "An approach to optimal pick-up and exciter placement," *Proceeding of SPIE International Society for Optical Engineering*, pp. 376–382, 1996.
- [10] Y. S. Park and H. B. Kim, "Sensor placement guide for model comparison and improvement," *Proceeding of SPIE International Society For Optical Engineering*, pp. 404–409, 1996.
- [11] A. P. Cherng, "Optimal sensor placement for modal parameter identification using signal subspace correlation techniques," *Mechanical Systems and Signal Processing*, vol. 17, no. 2, pp. 361–378, 2003.
- [12] M. Reynier and H. Abou-Kandil, "Sensors location for updating problems," *Mechanical Systems and Signal Processing*, vol. 13, no. 2, pp. 297–314, 1999.
- [13] J. E. T. Penny, M. I. Friswell, and S. D. Garvey, "Automatic choice of measurement locations for dynamic testing," *AIAA journal*, vol. 32, no. 2, pp. 407–414, 1994.
- [14] C. B. Larson, D. C. Zimmerman, and E. L. Marek, "A comparison of modal test planning techniques: excitation and sensor placement using the NASA 8-bay truss," *Proceeding of SPIE International Society for Optical Engineering*, pp. 205–205, 1994.
- [15] G. Heo, M. L. Wang, and D. Satpathi, "Optimal transducer placement for health monitoring of long span bridge," *Soil Dynamics and Earthquake Engineering*, vol. 16, no. 7-8, pp. 495–502, 1997.
- [16] W. Heylen and P. Sas, *Modal Analysis Theory and Testing*, Departement Werktuigkunde, Katholieke Universteit Leuven, 2006.
- [17] G. C. Marano, G. Monti, and G. Quaranta, "Comparison of different optimum criteria for sensor placement in lattice towers," *Structural Design of Tall and Special Buildings*, vol. 20, no. 8, pp. 1048–1056, 2011.
- [18] L. Yao, W. A. Sethares, and D. C. Kammer, "Sensor placement for on-orbit modal identification via a genetic algorithm," *AIAA journal*, vol. 31, no. 10, pp. 1922–1928, 1993.
- [19] R. R. Brooks, S. S. Iyengar, and J. Chen, "Automatic correlation and calibration of noisy sensor readings using elite genetic algorithms," *Artificial Intelligence*, vol. 84, no. 1-2, pp. 339–354, 1996.
- [20] W. Liu, W. C. Gao, Y. Sun, and M. J. Xu, "Optimal sensor placement for spatial lattice structure based on genetic algorithms," *Journal of Sound and Vibration*, vol. 317, no. 1-2, pp. 175–189, 2008.
- [21] G. Ma, F. L. Huang, and X. M. Wang, "Optimal placement of sensors in monitoring for bridge based on hybrid genetic algorithm," *Journal of Vibration Engineering*, vol. 21, no. 2, pp. 191–196, 2008.
- [22] K. D. Dhuri and P. Seshu, "Multi-objective optimization of piezo actuator placement and sizing using genetic algorithm," *Journal of Sound and Vibration*, vol. 323, no. 3–5, pp. 495–514, 2009.
- [23] Y. J. Cha, A. K. Agrawal, Y. Kim, and A. M. Raich, "Multi-objective genetic algorithms for cost-effective distributions of actuators and sensors in large structures," *Expert Systems with Applications*, vol. 39, no. 9, pp. 7822–7833, 2012.
- [24] C. He, J. C. Xing, J. L. Li et al., "A combined optimal sensor placement strategy for the structural health monitoring of bridge structures," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 820694, 9 pages, 2013.
- [25] A. R. M. Rao and G. Anandakumar, "Optimal placement of sensors for structural system identification and health monitoring using a hybrid swarm intelligence technique," *Smart Materials and Structures*, vol. 16, no. 6, pp. 2658–2672, 2007.
- [26] J. J. Lian, L. J. He, B. Ma et al., "Optimal sensor placement for large structures using the nearest neighbour index and a hybrid swarm intelligence algorithm," *Smart Materials and Structures*, vol. 22, no. 9, Article ID 095015, 2013.
- [27] L. J. He, J. J. Lian, B. Ma, and H. J. Wang, "Optimal multiaxial sensor placement for modal identification of large structures," *Structural Control and Health Monitoring*, vol. 21, no. 1, pp. 61–79, 2014.
- [28] R. C. Eberhart and J. Kennedy, "New optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, pp. 39–43, October 1995.
- [29] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [30] E. L. Wilson, *Three-Dimensional Static and Dynamic Analysis of Structures*. Computers and Structures, CSI, Berkeley, Calif, USA, 2002.
- [31] T. H. Yi, H. N. Li, and M. Gu, "Optimal sensor placement for health monitoring of high-rise structure based on genetic algorithm," *Mathematical Problems in Engineering*, vol. 2011, Article ID 395101, 12 pages, 2011.
- [32] Y. H. Shi and R. C. Eberhart, "A modified particle swarm optimizer," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 69–73, May 1998.
- [33] Y. H. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1945–1950, 1999.
- [34] R. C. Eberhart and Y. H. Shi, "Tracking and optimizing dynamic systems with particle swarms," in *Proceedings of the Congress on Evolutionary Computation*, pp. 94–100, May 2001.
- [35] A. Chatterjee and P. Siarry, "Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization," *Computers and Operations Research*, vol. 33, no. 3, pp. 859–871, 2006.
- [36] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240–255, 2004.

- [37] T. G. Carne and C. R. Dohrmann, "A modal test design strategy for modal correlation," *Proceeding of SPIE International Society for Optical Engineering*, pp. 927–927, 1995.
- [38] J. Teng and Y. H. Zhu, "Optimal sensor placement for modal parameters test of large span spatial steel structural," *Engineering Mechanics*, vol. 28, no. 3, pp. 150–156, 2011 (Chinese).
- [39] P. Kohnke, *ANSYS Theory Manual*, ANSYS Inc, 2001.
- [40] J. S. Zhang, Y. Wu, and S. Z. Shen, "Identification of natural modes with significant contributions to wind-induced vibration of single-layer reticulated shell," *Journal of Vibration Engineering*, vol. 19, no. 4, pp. 452–458, 2006 (Chinese).

Research Article

Forecasting Dry Bulk Freight Index with Improved SVM

Qianqian Han,¹ Bo Yan,² Guobao Ning,³ and B. Yu⁴

¹ School of Accountancy, Shandong University of Finance and Economics, Jinan, Shandong 250014, China

² Transportation Management College, Dalian Maritime University, Dalian 116026, China

³ School of Automotive Studies, Tongji University, Shanghai 201804, China

⁴ School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Guobao Ning; guobao.tj@163.com and B. Yu; yubjtt@126.com

Received 4 March 2014; Revised 5 May 2014; Accepted 6 May 2014; Published 11 June 2014

Academic Editor: Rui Mu

Copyright © 2014 Qianqian Han et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An improved SVM model is presented to forecast dry bulk freight index (BDI) in this paper, which is a powerful tool for operators and investors to manage the market trend and avoid price risking shipping industry. The BDI is influenced by many factors, especially the random incidents in dry bulk market, inducing the difficulty in forecasting of BDI. Therefore, to eliminate the impact of random incidents in dry bulk market, wavelet transform is adopted to denoise the BDI data series. Hence, the combined model of wavelet transform and support vector machine is developed to forecast BDI in this paper. Lastly, the BDI data in 2005 to 2012 are presented to test the proposed model. The 84 prior consecutive monthly BDI data are the inputs of the model, and the last 12 monthly BDI data are the outputs of model. The parameters of the model are optimized by genetic algorithm and the final model is conformed through SVM training. This paper compares the forecasting result of proposed method and three other forecasting methods. The result shows that the proposed method has higher accuracy and could be used to forecast the short-term trend of the BDI.

1. Introduction

The BDI published by the Baltic Exchange is used as an important evaluation factor for the dry bulk market in shipping industry. It is usually consulted by shipping operators and investors to forecast the trend of dry bulk market. However, as the price of dry bulk market changes almost every day and the affecting factors of the price are complicated, the prediction of the trend of dry bulk market becomes of difficulty. Since 2001, the BDI has experienced a huge fluctuation. The value of BDI was less than 1000 points at that time and increased to more than 11000 points in May, 2008. Five months later, it decreased to less than 800 points. Therefore, research on the law of shipping market freight fluctuation and the forecasting of the trend of BDI is of special significance for operators and investors to manage the market trend and avoid price risk in shipping industry.

Since the Baltic Freight Index (BFI) was established in 1985, many researchers and shipping scholars have made

in-depth research on the volatility and trend prediction of BFI and subsequent BDI. However, none of the forecasting methods is widely used in BDI prediction. The remarkable work in this field has been done by Kavussanos, who worked on the dry bulk market prices issues as early as the 1990s. Kavussanos and Visvikis [1] used VECM-GARCH model to investigate the lead-lag relationship in both return and volatilities between spot and future markets. Cullinake [2] is also a pioneer in developing BFI index forecasting method with ARIMA model. After that, some prediction techniques, such as statistical regression and neural network, are widely used in BDI prediction. However, the method of statistical regression is appropriate for stability, normality, and independence time series and not appropriate for complex time series. Neural network has a good nonlinear approximation capability, but the model structure is difficult to determine. It is prone to excessive training or insufficient training for the neural network, which induces some shortages such as trapping in local minimum, being sensitive to initial value,

and overreliance on design skills. Li and Parsons [3] used the neural network to forecast the tanker freight rate and then compared with ARMA model. They proved that the neural network model has higher prediction accuracy for a long time series. Cullinane et al. [4] forecasted the spot freight rate index with a simple single variable in ARMA model. A new model-fractional integrated autocorrelation moving average model (ARFIMA) is presented by Granger [5] in data forecasting. Based on the ARFIMA approach, Henry [6] showed that almost a half of the new-coming seventeen stock markets in the world in the 1990s have long memory, which is similar to their searches by Crato [7], Jonahan [8], and so on.

In recent years, econometric models are popular in freight rate index forecasting, for instance, ARIMA model, VAR model, and VECM model. Empirical analysis shows that the econometric models have higher accuracy than the traditional prediction methods for nonstationary time series. Veenstra and Franses [9] developed VAR model in forecasting BDI based on cointegrated process of the high time series and the method of unit root test. Kavussanos and Alizadeh-M [10] presented a single variable seasonal ARIMA-SARIMA model, multivariable seasonal cointegrating, and VAR to analyze the seasonal characteristics of dry bulk shipping market.

Tvedt [11] used the unit root tests to reject the random walk hypothesis of freight rate, confirming a state of stationary in freight rates forecasting for the classical shipping market models. Two years later, a rejection of applicability of the expectations theory in freight market was presented by Adland and Cullinane [12]. They also proved that the risk premium is time varying, depending on the freight market conditions and time charter duration. A fuzzy-DELPHI adjustment process for improvement of accuracy was proposed by Duru et al. [13]. They also illustrate its performance in the validation of adjustments of statistical forecasts in the BDI through an empirical study. Zhang et al. [14] employed R/S and GPH tests to model long memory of volatility of the indices based on the investigation of fluctuation features of dry bulk shipping market with the BDI.

A new machine learning method, namely, support vector machine (SVM), is widely employed in many fields, for instance, handwriting recognition, three-dimension objects recognition, faces recognition, text images recognition, voice recognition, and regression analysis [15–23]. The SVM based on statistical learning theory has good fitting ability for complex nonlinear function. At the same time, it can avoid trapping problem of overfitting learning. A support vector machine-based (SVM) model is developed to predict the baseline travel and dwell times of buses based on recent data by Yu et al. [24, 25]. The authors [24, 25] also presented a hybrid model based on support vector machine (SVM) and Kalman filtering technique to predict bus arrival times. Yu et al. [26] also adopted support vector machine (SVM), artificial neural network (ANN), k nearest neighbor's algorithm (k -NN), and linear regression (LR) for the bus arrival time prediction. van Gestel et al. [27] research the financial time series prediction problems based on the least squares SVM model. Cao and Tay [28] research the parameters of SVM selection and test their approach with empirical analysis of

financial time series. The SVM model is widely used in the stock market. Kim [29] compared the forecasting result of stock price index between SVM model with the BP neural network model and the CBR model. He proved that the SVM model has better prediction of financial time series. Huang et al. [21] applied SVM to forecast the movement direction of NIKKEI 225 index.

It is suitable for nonlinear time series prediction by the SVM since the full consideration with the randomness of the data sequence. From the application of SVM especially in financial index prediction, as well as freight data, it is no hard to be employed in another index prediction in terms of applicability. However, SVM is rarely studied and used in the field of BDI forecast. Therefore, this paper will analyze the applicability of SVM model in BDI forecasting and make an empirical test.

BDI data sequences can be regarded as signals changed over time. The signals usually have characteristics, such as periodic and seasonal, of dry bulk shipping market freight index fluctuation. The noise of the signal reflects the influence factors of freight index or random events. To grasp the rule of BDI data variation, the random disturbance factors should be eliminated. Therefore, denoising processing of the signal is necessary for an accuracy data forecasting. Some methods, such as a self-adaptive filtration method, the Kalman filter method, and average moving method, are often used to denoise signal of data. Yu et al. [30] adapt an adaptive filtration method in bus arrival time prediction model. However, the method of wavelet transform is proposed to denoise the raw signal in this paper.

Wavelet transform is the most widely used multiscale analysis method till now. The root of wavelet transform is scaling and translation in the signal analysis. It is a milestone in the history of the development of Fourier analysis. In 1982, a French oil exploration technician called J. Morfet tried to deal with irregular signals with wavelet method more effectively. Research of wavelet analysis was becoming popular after that. It is widely used in image processing, voice processing, and signal processing. In recent years, the wavelet transform method has been employed in areas of finance and economy, and the empirical result showed a good effect. Esteban et al. [31] predicted time series with wavelet method. Firstly, they decomposed the time series into two different cycles in sequence with wavelet. Then, ARMA model is present in regression of each data cycle. Lastly, the two forecasting values are added to get the final prediction results. They proved that their proposed approach had better predicted result than ARMA model.

Moreover, some researchers also tried to establish the combined model with the wavelet transform and SVM [32–39]. Wu [40] proposed novel robust wavelet support vector machine, which is based on wavelet theory and the modified support vector machine. They also designed swarm optimization algorithm to select the optimal parameters of their proposed model. Two years later, Wu and Law [41] made an in-depth research based on the wavelet support vector machine proposed by Wu [40]. Wavelet transform is also used by Guo to map the sample data into several time-frequency domains. He then developed the SVM model to forecast the

gross value of textile products in Japan precisely. A wavelet transform and SVM combined model is developed by Hsieh and Chen to predict the dissolved oxygen density in water-quality process. Their results showed a higher accuracy of the combined model than BP neural network model. Liu and Fan [42] stated that the performance of SVM can be improved with the introducing of discrete wavelet transform. Wang et al. [43] used the wavelet transform to decompose the dam deformation time series into different frequency components and then forecast the series with a SVM model. A wavelet kernel function for SVM is presented by Wei and Lin [17]; they also denoised the signal with multiscale interpolation and sparse attributes. The performance proved that their proposed model was accurate and convergent.

Although there are many studies of the combination of wavelet transform and SVM, few have been made in dry bulk freight index. Therefore, this paper constructs a wavelet transform and SVM combined forecast model. It removes the random factors in BDI series with wavelet and then establishes a SVM model. The numerical analysis shows that our method has better predicting results than the commonly used prediction methods. Of course as a prediction method, it should be tested with large numbers of tests while evaluating the accuracy of its prediction, which is, however, the shortage of this paper for the time limited.

This paper is organized as follows. Section 2 reviews the data on the shipping freight market and analyzes the future of the BDI data. Section 3 presents the decomposition and reconstruction of wavelet. The forecasting models and procedures are proposed in Section 4. A case study is shown in Section 5 and the performance of several prediction results is compared. Besides, the conclusion of this paper and the recommendations for future studies are provided in this part.

2. Characters and Influence Factors of BDI

The volatility of freight is directly reflected by the fluctuation of freight indexes, for example, BDI. In terms of market structure, freight price depends on the supply (ship owner) and the demand (cargo). Concerning market economies, freight price of general cargo is mainly influenced by three external factors: an act of war or natural disaster, the global economy, and the market speculation.

In retrospect, force majeure, for example, war factors, is the major power driving the fluctuation of the world shipping market, especially in the turbulent times. In 1956, the outbreak of Suez Crisis drastically increased the shipping market risks throughout the world. Shipping lines and area changed a lot, and supply in dry bulk shipping market rapidly went down, which led to the high volatility of freight price. In 1973, the third Middle East war broke out; Arab countries firstly used the "oil weapon," resulting in the sharp increasing of fuel price and freight price consequently. The First World War, the Second World War, the Middle East wars, hurricane, tsunamis, and other natural disasters brought high risks to maritime transport market. Firstly, wars and natural disasters such as force majeure occurrence or even expectation of those events can affect the confidence of both ship owners and

shippers; secondly, once sailing area is limited, such as the close of Suez Canal during the second Middle East war, the average travel distance will increase and the supply capacity will drop significantly; besides, the rise of oil prices due to wars will also increase shipping costs.

Shipping derivation has shown that the world economic situation and the development of international trade play a decisive role in the shipping market existence and changes. Therefore, the economic cycle and trade demand are the durable and fundamental influences in the shipping market. The most remarkable presentation of the impact of economic environment on maritime shipping was the terrible hit of the global economic crisis to the shipping industry. Economic crisis led to slower global economic growth and commodity prices falling sharply. In the first half of 2009, the fixed capital formation and manufacturing output of the world's major economies have double-digit decline. Steel mills and other enterprises, in order to cope with shrinking demand, take measures of limiting production or semiproduction, which led to the demand on iron ore and coal dropping significantly. Dry Freight Index experienced unprecedented volatility in the six months from the highest point in history falling to the lowest point. The demand of iron ore which was the largest dry bulk seaborne trade at that time decreased significantly. China, as the largest importer, unloaded 30 million tons of iron ore imports in Nov., 2008, down by 20.7%, which was the first negative growth. The impact of economic crisis on supply capacity is mainly reflected in the shipbuilding market. Global economic downturn led to sharp decline in shipbuilding demand, and some ship owners began to cancel the order because of the shortage of money. Because there are one to two years of construction time from ordering to delivery, the impact of the economic crisis on the shipbuilding industry has one to two years of lag extending to freight market. Therefore, new ships to be delivered two years after ordering will substantially decrease, resulting in shrinking supply capacity.

Freight derivatives were created in order to avoid the risk of emergencies in shipping market. Major functions of freight derivatives reside in hedging and price discovery. Freight derivatives include the Baltic Freight Index futures (BIFFEX), forward freight agreements (FFA), and the shipping options (freight option). Volatile freight rates since 2004 have given speculators profit opportunities. Investment banks as Goldman Sachs, Morgan Stanley, and other financial institutions and hedge funds have entered into speculative market; some shipping companies also use their information superiority to engage in the market.

According to dry bulk freight index trend, freight index, it disorderly changes in random variation, so it is difficult to grasp the change regulation. In order to better grasp the inherent regulation of fluctuations, it can be divided into two categories: the first category is the one in which there is a pattern existing, for example, the world economy with cyclical characteristics, coal, iron ore, grain production capacity, and shipping capacity with seasonal fluctuations; the second category is sudden and random factors, for example, natural climate political events, average travel distance, scientific and technological development, country's international

trade policy, sudden changes in trade structure, economic interest transferred, exchange rate fluctuations, ship archive, operational productivity, international shipping norms, and market rumors.

After the above analysis, both the two factors have effect on the dry bulk freight index. To grasp more accurate freight index fluctuation characteristics, what needs to be done is to dig out the historical dry bulk freight index data and then use data processing methods to eliminate the disorder characteristics caused by the second category of factors. Based on that, the most suitable methods are used for BDI prediction.

If the time series of BDI can be regarded as a kind of changeable signal with time elapsing, then there is rich information in the signal. The first category includes the information of cyclical fluctuations of BDI. As the cycle is long-term, the first category factors have lower frequencies and are located in low frequency range. The second category factors are stochastic, irregular, and unexpected. Though those factors occur not very often, the frequency can be still relatively high if aggregating the second category factors into monolithic. So the high frequency range includes the second category factors. The discussion on cyclical fluctuation of BDI is based on thought as follows: (1) signal reconstruction. Extracting BDI signal process attempts to remove stochastic, irregular and unexpected factors, and noise from the BDI signal by separating the low and the high frequency part. (2) BDI is an output of a complex function as there are so many factors impacting on the dry bulk freight market. In order to analyze the BDI signal accurately, this paper applies the SVM onto the prediction of the reconstructed signal based on the results of extracting BDI signal process.

3. Adopting the Wavelet Transform to Denoise the BDI

Useful signal is commonly presented as stationary signals or low frequency signals, while noise signal is usually unstable and has high frequency. Therefore, the characteristics of BDI ensure the application of wavelet analysis to eliminate noise signal. When using wavelet analysis to remove noise signal from shipping indexes, such noise signal is mainly included in high frequency wavelet coefficients, for which the threshold method can be used for decomposing wavelet coefficients. Each layer of decomposed wavelet coefficients should be reconstructed to eliminate the noise. The purpose of removing noise signals from BDI signal $S(t)$ is to obtain actual signal $f(t)$ from $S(t)$, by which the authenticity of data can be ensured.

The one-dimension model of BDI signal with noises can be presented as follows:

$$S(t) = f(t) + \sigma e(t), \quad t = 0, 1, \dots, n-1, \quad (1)$$

where $f(t)$ is the real signal; $e(t)$ is the noise; σ is the noise intensity; $S(t)$ is the signal with noises.

The process of wavelet noise reduction is the process of decomposition and reconstruction for signal. Original function or signal is split into several relevant pieces without

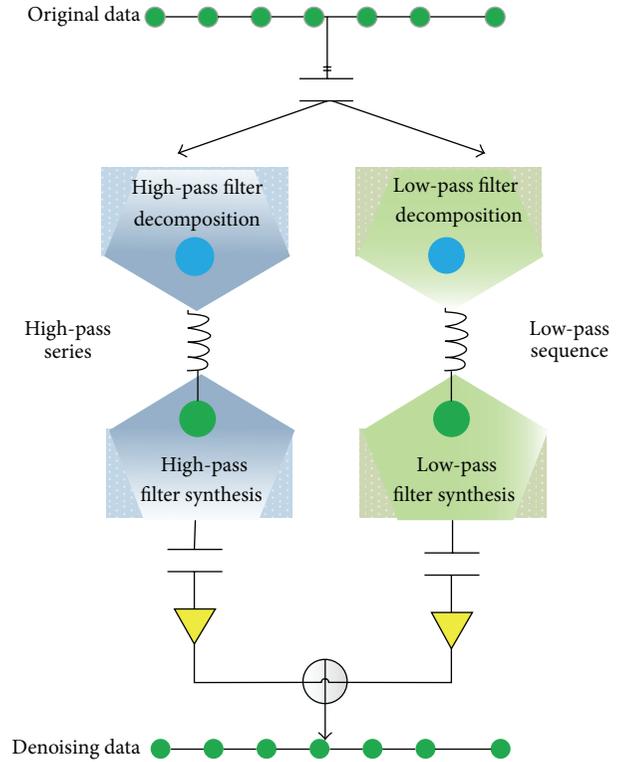


FIGURE 1: Decomposition and reconstruction of wavelet transform denoising process.

losing much information. Those pieces are such wavelet which changes in scale and decays in time. The wavelet reconstruction is the process where those pieces are combined to restore the real features.

The decomposition and reconstruction of wavelet are shown as in Figure 1.

BDI is one-dimension time series and the wavelet denoising process against such kind of signal is usually expressed as the procedure presented as follows.

Step 1. Preprocessing the data, which include noises, for using in next steps.

Step 2. Wavelet denoising process to the one-dimension signal. Selecting a suitable wavelet mother function and setting an appropriate decomposing layer N . Decomposing $S(t)$ into N layers.

Step 3. Quantizing the threshold of wavelet decomposition coefficients. Selecting a suitable threshold for the high frequency coefficient of each layer.

Step 4. Inverse transform of one-dimension wavelet. Based on the coefficient of N th layer and the quantized high frequency coefficients from 1st to N th layer, reconstructing the one-dimension signal. The reconstructed signal is the denoised signal.

Theoretical base of wavelet denoising is presented as follows.

$\psi(t)$ is a function where Fourier transform exists. If its Fourier transform $\widehat{\psi}(t)$ meets the condition $\int_{-\infty}^{\infty} (\widehat{\psi}(t)^2 / \omega) d\omega < \infty$, the function can be a wavelet function. Supposing $j \in Z$ and $\psi_{2^j}(t)$ is the dyadic stretching transformation of $\psi(t)$ against factor 2^j , then $\psi_{2^j}(t)$ can be expressed as

$$\psi_{2^j}(t) = \frac{1}{2^j} \psi\left(\frac{t}{2^j}\right). \quad (2)$$

Wavelet transform of function $f(t)$ with scale 2^j at position t can be defined as the convolution of $f(t)$ and $\psi_{2^j}(t)$, presented as

$$W_{2^j} f(t) = f \times \psi_{2^j}(t). \quad (3)$$

For wavelet function $\psi(t)$, supposing there exist constants A and B and $0 < A \leq B < \infty$, then we can get

$$\forall \omega \in R, \quad A \leq \sum_{j=-\infty}^{\infty} \widehat{\psi}(2^j \omega) \leq B(2), \quad (4)$$

where $\widehat{\psi}(t)$ is the Fourier transform of $\psi(t)$. Then $\psi(t)$ can be called dyadic wavelet function and the corresponding wavelet transform can be called dyadic wavelet transform.

For any function $\chi(t)$ with Fourier transform, if its Fourier transform meets Subject (5)

$$\sum_{j=-\infty}^{\infty} \widehat{\psi}(2^j \omega) \widehat{\chi}(2^j \omega) = 1, \quad (5)$$

then it can be called reconstruction wavelet. It can be easily found that there are countless functions $\chi(t)$ meeting Subject (5).

The dyadic wavelet transform is complete and stable. The ‘‘complete’’ means that the function can be restored by its dyadic wavelet transform. In terms of energy, ‘‘stable’’ means that the total ability of dyadic wavelet transform has limitation which is close to the energy of the function. Function $f(t) \in L^2(R)$ can be restored by its dyadic wavelet transform and the corresponding reconstruction wavelet on the basis of

$$f(t) = \sum_{-\infty}^{\infty} W_{2^j} f \times \chi_{2^j}(t), \quad (6)$$

$$A \|f\|^2 \leq \sum_{j=-\infty}^{\infty} \|W_{2^j} f(t)\|^2 \leq B \|f\|^2. \quad (7)$$

In practical application, the measurable resolution of signal is limited, so it is impossible to conduct wavelet transform on all scales 2^j ($-\infty < i < \infty$). Therefore, 2^j should be set as a limited value. The wavelet transform is confined between a limited maximum scale $j = J$ and a limited minimum scale $j = 1$. 2^1 is the highest resolution and 2^J is the lowest resolution. With respect to resolution, it is relevant to frequency. That is to say, the higher the frequency

is, the higher the resolution is, and vice versa. To express the signal resolution decomposition of wavelet transform, a real function $\varphi(t)$ is introduced hereafter, whose Fourier transform should meet Subject (8). Consider

$$\widehat{\varphi}(t)^2 = \sum_{j=1}^{+\infty} \widehat{\psi}(2^j \omega) \widehat{\chi}(2^j \omega). \quad (8)$$

According to (3) and (6), it can be easily obtained that

$$\begin{aligned} \widehat{\varphi}(0)^2 &= \lim_{k \rightarrow 0} \widehat{\psi}(2^k \omega)^2 = \lim_{k \rightarrow 0} \widehat{\psi}(2^j \omega) \widehat{\chi}(2^j \omega) = 1 \\ \widehat{\varphi}(\infty)^2 &= \lim_{k \rightarrow +\infty} \widehat{\psi}(2^k \omega)^2 = \lim_{k \rightarrow +\infty} \widehat{\psi}(2^j \omega) \widehat{\chi}(2^j \omega) = 1. \end{aligned} \quad (9)$$

Equation (9) indicates that the energy of $\widehat{\varphi}(\omega)$ gathers in the low frequency range, so $\varphi(t)$ is a smooth function with low-pass characteristics. A smooth operator S_{2^j} is defined as follows:

$$\begin{aligned} S_{2^j} f(t) &= f * \varphi_2(t), \\ \varphi_2(t) &= \frac{1}{2^j} \varphi\left(\frac{t}{2^j}\right), \end{aligned} \quad (10)$$

where $S_{2^j} f(t)$ denotes the low-pass filtering component of signal $f(t)$ when the resolution is 2^j . The high frequency component of $f(t)$ is not presented in $S_{2^j} f(t)$ but in the dyadic wavelet transform $\{W_{2^j} f(t)\}_{1 \leq j \leq J}$ between scales 2^1 and 2^J , so $W_{2^j} f(t)$ stands for the detailed component and $S_{2^j} f(t)$ means the low-pass smooth component of the signal. The signal details (the high frequency ingredient) contained in $S_{2^j} f(t)$ decrease with 2^j increasing, and the lost information can still be restored by the wavelet transform $W_{2^j} f(t)$.

The time series is defined as $S_2^{d_0} f$, and the low-pass smooth component at scale 2^j is defined as $S_2^{d_j} f$. According to (7), $S_2^{d_j} f$ can be split into the low and the high half frequency denoted by $S_2^{d_j} + 1f$ and $W_2^{d_j} + 1f$, respectively. The d is the concrete signal. The decomposition algorithm of $S_2^{d_j} f$ is shown as follows:

$$\begin{aligned} j &= 0, \\ \text{while } (j < J), \\ W_2^{d_j} + 1f &= (1/\lambda_j) S_2^{d_j} f * G_j, \\ S_2^{d_j} + 1f &= S_2^{d_j} * H_j, \\ j &= j + 1, \\ \text{the end.} \end{aligned}$$

The reconstruction algorithm of $S_2^{d_0} f$ is shown as follows:

$$\begin{aligned} j &= J, \\ \text{while } (j > 0), \\ S_2^{d_j} + 1f &= \lambda_j W_2^{d_j} f * K_{j-1} + S_2^{d_j} f * H_{j-1}, \\ j &= j - 1, \\ \text{the end,} \end{aligned}$$

where G_j , H_j , and K_j are a group of corresponding filters.

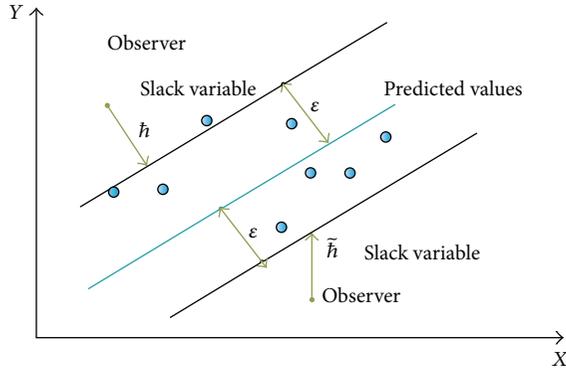


FIGURE 2: The ε -insensitivity tube of SVM.

4. Forecasting BDI with SVM

4.1. Support Vector Machine. The support vector machine is a kind of machine learning system, with the purpose of maximizing the margin distance between different categories of problems [44–46]. The model of SVM is as follows:

$$f(x) = \omega \times \varphi(x) + b, \quad (11)$$

where ω is the weight vector; b is error; $\varphi(x)$ is a kernel function to deal with the nonlinear problem, with mapping the nonlinear input to a high dimensional space by a nonlinear function, to make the input linear.

The least square method in conventional regression model takes the square error as the loss function in accordance with minimizing empirical risks. Vapnik et al. [44] took the ε -insensitivity as the loss function in SVM model, and the ε -insensitivity loss is shown as

$$L_\varepsilon(f(x) - y) = \begin{cases} |f(x) - y| - \varepsilon & |f(x) - y| \geq \varepsilon \\ 0 & \text{others,} \end{cases} \quad (12)$$

where parameter ε determines the area of ε -insensitivity (Figure 2). When the predicted value $f(x)$ is within the tube area, the loss is zero; otherwise, the loss is the difference between the prediction error and the tube area radius ε . \tilde{h} and \tilde{h} are slack variables, indicating the prediction errors in different directions:

$$L_\varepsilon(f(x) - y) = \begin{cases} |f(x) - y| - \varepsilon = \tilde{h} & |f(x) - y| \geq 0 \\ |f(x) - y| - \varepsilon = \tilde{h} & |f(x) - y| < 0 \\ 0 & \text{others,} \end{cases} \quad (13)$$

where \tilde{h} is the training error which is higher than the area boundary; \tilde{h} is the training error which is lower than the area boundary.

In the input space, SVM uses the minimize-adjustment-risk function to calculate the weight vector and the error. The function is shown as

$$R(C) = C \frac{1}{N} \sum_{i=1}^n L_\varepsilon(f(x_i), y_i) + \frac{1}{2} \|w\|^2, \quad (14)$$

where $L_\varepsilon(f(x_i), y_i)$ is the ε -insensitivity loss function; $C(1/N) \sum_{i=1}^n L_\varepsilon(f(x_i), y_i)$ is the empirical error; $(1/2)\|w\|^2$ is the adjustment item. Then the SVM model can be figured out with minimizing

$$\begin{aligned} \text{Min} : & \frac{1}{2} w^T w + C \sum_i (\tilde{h} + \tilde{h}) \\ \text{subject to} & \begin{cases} y_i - w^T x_i - b \leq \varepsilon + \tilde{h} \\ w^T x_i + b - y_i \leq \varepsilon + \tilde{h} \\ \tilde{h}, \tilde{h} \geq 0, \end{cases} \end{aligned} \quad (15)$$

where $i = 1, 2, \dots, n$ is the number of samples for training; $\tilde{h} + \tilde{h}$ is empirical risks; $(1/2)w^T w$ is structure risks which can avoid excessive learning; C is correction factor, indicating the balance between the experimental risk and the structure risk. Larger C means the model pays more attention to the experimental risk, otherwise, more attention to the structure risk. When C , ε , and the kernel function k which meets Mercer's condition are determined appropriately, the model can be solved with Lagrangian multiplier method.

Besides, in the process of artificial intelligent model construction, different data will lead to different combinations of best parameters. Therefore, the trial-and-error method is widely used to search the best parameter combination. With synthetically considering Cherkassky and Ma's suggestions [47] in parameter setting, this paper firstly applies Cherkassky and Ma's method [47] to estimate training data to calibrate several suggested parameter combinations (C and ε) of SVM model. Then the exponent search method is employed to select the best parameter combination, based on minimizing the mean square error. The method can prevent the risk of simple suggested parameter combination and also reduce the trial-and-error times.

4.2. Combined Model. In this paper, wavelet transform decomposes the original sequence of BDI layer by layer and then gets a low frequency signal layer and N high frequency detailed layers (N is a decomposition level). Fluctuation of international dry bulk shipping market is included in the low frequency part of the BDI. The impact of random factors such as incidents is included in the high frequency part. But the high frequency part is not an irregular mutational factor. Therefore, it needs to denoise each layer sequence of low and high frequencies, respectively. A denoised BDI sequence is retained by reconstructing. The process of sequence denoising not only filters random factors but also makes the predictive model robust.

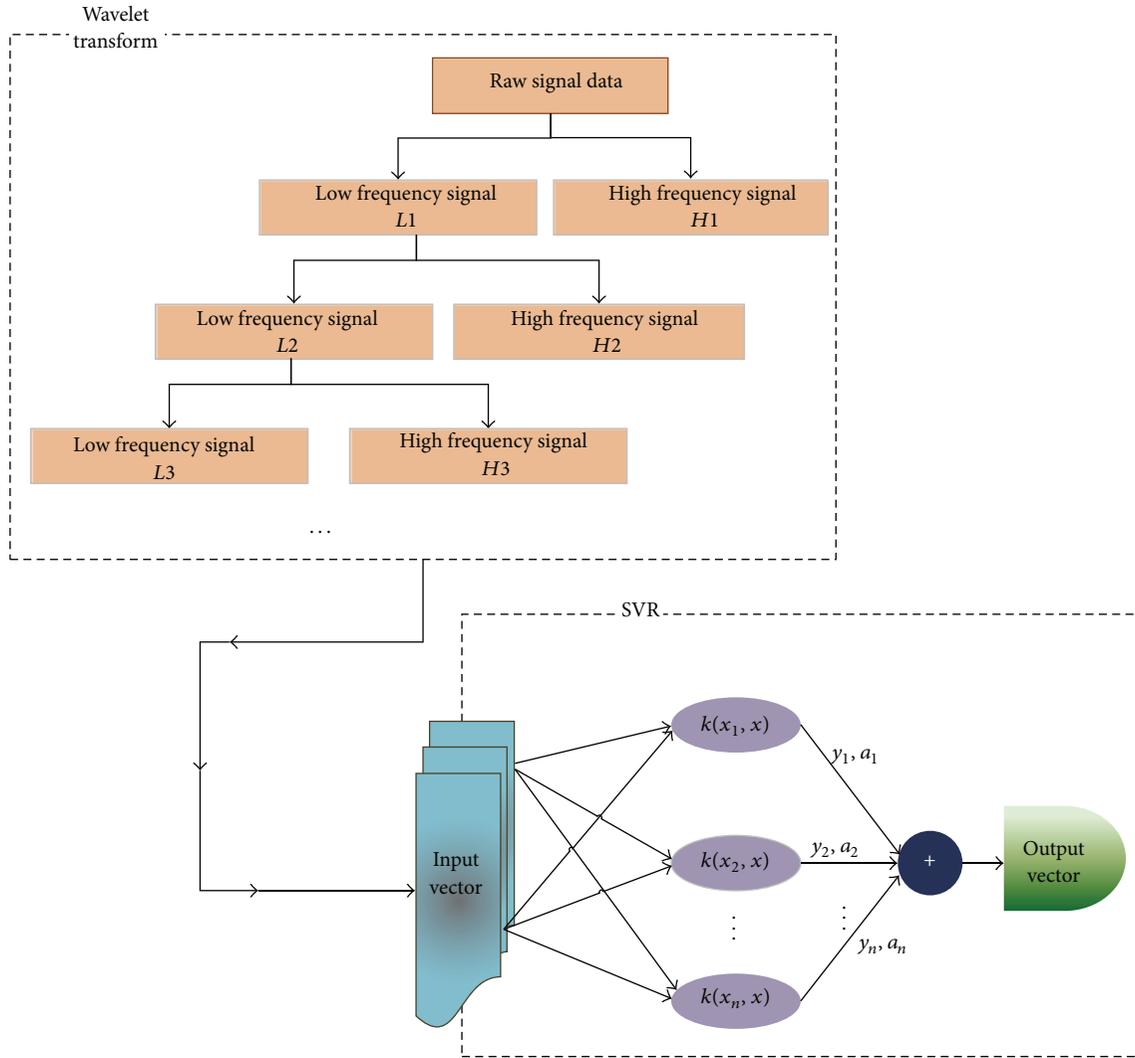


FIGURE 3: Structure of the wavelet transform-SVM combined model.

Wavelet transform has characteristics of time-frequency localization and zoom features, while support vector machine has nice tolerance of self-learning adaptive fault, generalization ability, and robustness. Through operation functions such as scaling and translation, wavelet transform is able to analyze functions or signals with multiscale refinement. Wavelet SVM is combined by the wavelet analysis and SVM can deal with nonlinear function approximation uniquely. This research uses wavelet transform to analyze BDI sequence and then trains the time series by SVM to get trained models and predictions. Figure 3 shows the structure of hybrid forecasting model.

5. Case Study

Since 2001, the BDI has experienced a huge fluctuation. The value of BDI was less than 1000 points at that time and increased to more than 11000 points in May, 2008. Five months later, it decreased to less than 800 points. This paper

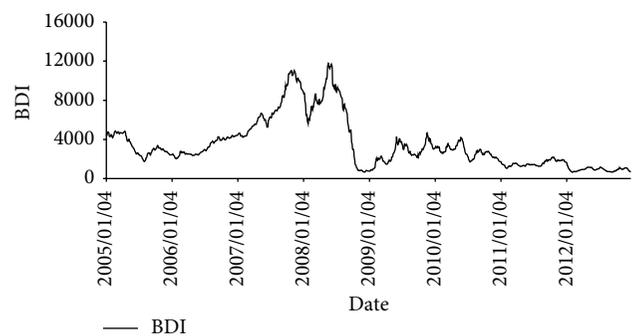


FIGURE 4: Historical data of monthly averaged BDI (2005/1-2012/12).

takes data of the BDI published by the Baltic Exchange from January 2005 to December 2012 as the empirical objective. Besides, the daily BDI data is replaced by month data; that is, the objective data is the average BDI for each month.

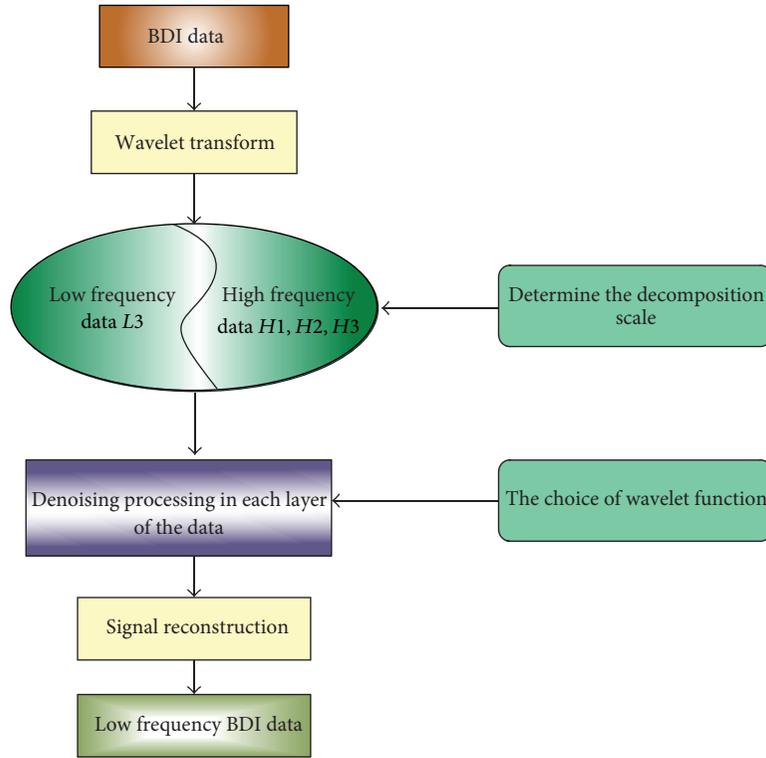


FIGURE 5: The wavelet transforming process of BDI series.

So there are 96 data of BDI. Among them, the 84 prior consecutive monthly BDI data are the inputs of the model, and the last 12 monthly BDI data are the outputs of model. The parameters of the model are selected and the final model is conformed through SVM training. Figure 4 shows the fluctuation phenomenon of monthly data.

5.1. Process Data. To avoid the training error resulting from dimension in sample data or a large dimension data value, the whole data should be normalized and processed before the SVM training. Consider

$$S'_i = 2 \cdot \frac{S_i - S_{\min}}{S_{\max} - S_{\min}} - 1, \quad (16)$$

where S'_i is normalized value. S_i is raw value. S_{\min} is the minimum value in a sequence of samples. S_{\max} is the maximum value in a sequence of samples.

5.2. Wavelet Analysis. The denoising process of original BDI sequence is presented by wavelet transform, which is shown in Figure 5. Figure 5 shows the wavelet transform process of BDI series. Firstly, the raw BDI data split into the high frequency data and the low frequency data decomposed with the wavelet transform. Then, by use of some tech-methods, such as threshold, each sequence will be processed with manic elimination. Go around and around until the final low frequency sequence is chosen.

Two problems, which wavelet function should be selected in denoising process and how to determine the decomposition scale, should be solved. Different wavelet function will get different wavelet transform analysis results, which is important for the effect of denoising. There is no acknowledged method about how to choose the optimal wavelet functions and decomposition scale for signal denoising. So this paper settles the above two problems with experiment.

The purpose of denoising is to remove the mutation factors and random effects in the sequence. So the denoised sequence should not be too smooth or existing obvious step phenomenon. Considering the orthogonality, symmetry, smoothness, and other characteristics of the wavelet function, the best wavelet function and the decomposition scale are determined. The paper used the wavelet toolbox of MATLAB to make the test.

The commonly used wavelet functions are *Haar wavelet*, *dbN wavelet*, *symN wavelet*, *biorN wavelet*, *coifN wavelet*, *dmey wavelet*, and so on. We make transformation analysis for the BDI sequence with the same scale and the same order number with different wavelet function. This paper will take three layers of decomposition. So the N is selected as 3. After the experiece, the *dbN wavelet* is selected as the one in denoising BDI sequence.

Then, different coefficients of dbN wavelet function are used to analyze wavelet transform. The coefficients of dbN wavelet function are usually selected from 1 to 6. Through effective comparison, the coefficient of dbN wavelet function is settled as 3.

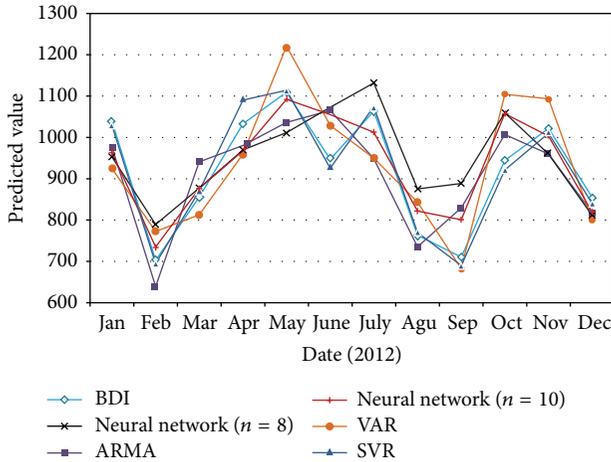


FIGURE 6: Forecasting results of four prediction models.

5.3. *The Wavelet-SVM to Forecast BDI Sequence.* The 84 prior consecutive monthly BDI data are the inputs of the model, and the last 12 monthly BDI data are the outputs of model. The SVM function with output close to the last 12 monthly BDI data will be selected. The parameters in SVM, which greatly influence the performance of SVM, need to be optimized and set by users. Heuristic algorithms have been successfully used in many complex problems [48–51]. Genetic algorithm (GA) is a common heuristic algorithm which has been widely used in lots of literatures [46, 52]. Therefore, GA is also used to optimize the three parameters C and ϵ for SVM. Due to lots of literatures about GA for references [46, 52], the process of GA has not been introduced in this paper. Before the implementation of GA, there are four GA parameters, namely, p_c , p_m , p_{size} , and T_{max} , which need to be predetermined. In general, p_c varies from 0.3 to 0.9. p_m varies from 0.01 to 0.1; p_{size} is the population size which is set according to the size of the samples. T_{max} is the maximum number of generation. At last, after the optimization of GA, the two parameters of SVM were optimized as (5.5 and 0.02) with the best optimization value.

Then, the trained model is presented for one-step prediction on the last 12 monthly data. To test the forecasting effect of mixed-model, three traditional econometric methods, ARIMA model, VAR model, and neural network model, are proposed for one-step prediction on the same sample data. Since the above three models use the raw BDI sequence as the input sample for index forecast, it has a strong comparability. Compare the results (Table 2) of one-step prediction with the actual value of BDI. For easy understanding and comparing, the actual and predicted values are antinormalized so that the data back to the real market freight index level. Figure 6 shows the compared results of the four predicted models.

As can be seen from Figure 6, the predicted results obtained from three models have the same trend with the actual value of BDI. However, among them, the deviation between the prediction results of neural network and the real value is the maximum. This is because that the international dry bulk market in 2007 and 2008 has always been in volatile

mood, causing the artificial neural network falling into the overlearning problem in the case of small samples. Therefore, it amplifies the up and down magnitude of BDI values for the BDI forecast after 2008. ARMA and VAR itself are suitable for short-term time series prediction, and results are better than the neural network model obviously. However, as can be seen in Figure 6, at some turning points, Wavelet-SVM model is more close to the true value than the ARMA model. Table 1 shows the forecasting value of each prediction model.

This paper uses root mean square error (RMSE) to test training effect and forecasting precision of the various forecasting methods:

$$RMSE = \left(\sum_{i=1, \dots, N} \frac{(S_{fi} - S_{ri})^2}{N} \right)^{1/2}, \quad (17)$$

where S_r is the actual value of BDI index and S_f is the prediction value.

By calculating the RMSE of the above four models with the forecasting result, we see that the wavelet-SVM hybrid prediction model has the best prediction accuracy. The large deviation among the four models is related with the fall of BDI under the influence of the economic crisis in 2008. BDI value fell more than 90% from more than 17000 points in May, 2008, to less than 700 points in end of 2008. Therefore, seeing from the predicted trend and the prediction accuracy of each forecasting model, wavelet SVM is the most suitable method in short-term prediction of BDI.

6. Conclusions

Research on the law of shipping market freight fluctuation and the forecasting of the trend of BDI is of special significance for operators and investors to manage the market trend and avoid price risk in shipping industry. Therefore, this paper constructs a wavelet transform and SVM combined forecast model. It removes the random factors in BDI series with wavelet and then establishes a SVM model. The BDI data in 2005 to 2012 are presented to test the proposed model. The 84 prior consecutive monthly BDI data are the inputs of the model, and the last 12 monthly BDI data are the outputs of model. The parameters of the model are selected and the final model is conformed through SVM training. This paper compares the forecasting result of proposed method with three other forecasting methods (VAR model, ARMA model, and neural network). The result shows that the proposed method has higher accuracy and could be used to forecast the short-term trend of the BDI. In further research, we will be devoted to improving the prediction accuracy and to forecasting the BDI with long-term period.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

TABLE 1: Forecasting results of five prediction models.

	BDI	ARMA			VAR			Neural network ($\tau = 8$)			Neural network ($\tau = 10$)			SVM	
		Predict value	Relative error	Relative error	Predict value	Relative error	Relative error	Predict value	Relative error	Relative error	Predict value	Relative error	Predict value	Relative error	Predict value
Jan. 12	1039.381	977.1347	0.059888	0.109869	925.1847	0.109869	0.082599	953.5296	0.082599	963.4226	0.07308	1029.505	0.009502	1029.505	0.009502
Feb. 12	702.619	639.3836	0.09	0.099965	772.8565	0.099965	0.123603	789.4652	0.123603	735.1234	0.046262	694.6194	0.011385	694.6194	0.011385
Mar. 12	855.381	941.0745	0.100182	0.049621	812.9358	0.049621	0.025919	877.5513	0.025919	876.2803	0.024433	870.5223	0.017701	870.5223	0.017701
Apr. 12	1032.905	980.6357	0.050604	0.070868	959.705	0.070868	0.061074	969.8216	0.061074	968.0003	0.062837	1090.248	0.055517	1090.248	0.055517
May 12	1109.762	1036.372	0.066132	0.100325	1221.098	0.100325	0.088066	1012.03	0.088066	1092.27	0.015762	1114.378	0.00416	1114.378	0.00416
June 12	947	1065.348	0.124972	0.086849	1029.246	0.086849	0.132989	1072.941	0.132989	1055.275	0.114335	925.786	0.022401	925.786	0.022401
July 12	1064.048	948.7609	0.108347	0.107174	950.0095	0.107174	0.064036	1132.184	0.064036	1012.775	0.048186	1075.023	0.010315	1075.023	0.010315
Aug. 12	763.5714	735.4512	0.036827	0.104717	843.5304	0.104717	0.146688	875.5779	0.146688	822.0825	0.076628	770.095	0.008543	770.095	0.008543
Sep. 12	710.381	829.6896	0.16795	0.039411	682.384	0.039411	0.251183	888.8169	0.251183	800.9311	0.127467	690.1364	0.028498	690.1364	0.028498
Oct. 12	944.619	1007.365	0.066424	0.169579	1104.807	0.169579	0.121653	1059.535	0.121653	1057.672	0.119681	921.4623	0.024514	921.4623	0.024514
Nov. 12	1021.714	960.0282	0.060375	0.070432	1093.676	0.070432	0.060438	959.9639	0.060438	1003.558	0.017771	1010.307	0.011165	1010.307	0.011165
Dec. 12	855.6875	817.9351	0.044119	0.065158	799.9326	0.065158	0.051794	811.3679	0.051794	818.812	0.043095	840.0157	0.018315	840.0157	0.018315

TABLE 2: RMSE of the three prediction models.

Model	RMSE
ARMA	78.96465
VAR	90.56454
Neural network ($n = 8$)	96.95657
Neural network ($n = 10$)	65.97173
SVM	21.67031

Acknowledgments

The research is sponsored by the National Natural Science Foundation of China 51108053, Shandong Natural Science Fund Project ZR2011GQ011, the Trans-Century Training Program Foundation for Talents from the Ministry of Education of China NCET-12-0752, and Liaoning Excellent Talents in University LJQ2012045.

References

- [1] M. G. Kavussanos and I. D. Visvikis, "Market interactions in returns and volatilities between spot and forward shipping freight markets," *Journal of Banking and Finance*, vol. 28, no. 8, pp. 2015–2049, 2004.
- [2] K. Cullinane, "A short adaptive forecasting modal for BIFFEX speculation: a Box-Jenkins approach," *Maritime Policy & Management*, vol. 2, pp. 91–114, 1992.
- [3] J. Li and M. G. Parsons, "Forecasting tanker freight rate using neural networks," *Maritime Policy & Management*, vol. 24, no. 1, pp. 9–30, 1997.
- [4] K. P. B. Cullinane, K. J. Mason, and M. Cape, "A comparison of models for forecasting the Baltic Freight Index: Box-Jenkins revisited," *International Journal of Maritime Economics*, vol. 1, no. 2, pp. 15–39, 1999.
- [5] C. W. J. Granger, "Long memory relationships and the aggregation of dynamic models," *Journal of Econometrics*, vol. 14, no. 2, pp. 227–238, 1980.
- [6] O. T. Henry, "Long memory in stock returns: Some international evidence," *Applied Financial Economics*, vol. 12, no. 10, pp. 725–729, 2002.
- [7] N. Crato, "Some international evidence regarding the stochastic behavior of stock returns," *Applied Financial Economics*, vol. 4, no. 1, pp. 33–39, 1994.
- [8] H. W. Jonahan, "Long memory in emerging stock market returns," Federal Reserve System Working Paper 650, 1999.
- [9] A. W. Veenstra and P. H. Franses, "A co-integration approach to forecasting freight rates in the dry Bulk shipping sector," *Transportation Research Part A*, vol. 31, no. 6, pp. 447–458, 1997.
- [10] M. G. Kavussanos and A. H. Alizadeh-M, "Seasonality patterns in dry bulk shipping spot and time charter freight rates," *Transportation Research Part E*, vol. 37, no. 6, pp. 443–467, 2001.
- [11] J. Tvedt, "A new perspective on price dynamics of the dry bulk market," *Maritime Policy and Management*, vol. 30, no. 3, pp. 221–230, 2003.
- [12] R. Adland and K. Cullinane, "A time-varying risk premium in the term structure of bulk shipping freight rates," *Journal of Transport Economics and Policy*, vol. 39, no. 2, pp. 191–208, 2005.
- [13] O. Duru, E. Bulut, and S. Yoshida, "A fuzzy extended DELPHI method for adjustment of statistical time series prediction: An empirical study on dry bulk freight market case," *Expert Systems with Applications*, vol. 39, no. 1, pp. 840–848, 2012.
- [14] H. Zhang, F. Wei, and Z. Zhang, "Modeling volatility of baltic dry bulk freight index," in *Proceedings of the IEEE International Conference on Automation and Logistics (ICAL '08)*, vol. 9, pp. 1089–1094, September 2008.
- [15] B. L. Koley and D. Dey, "Automatic detection of sleep apnea and hypopnea events from single channel measurement of respiration signal employing ensemble binary SVM classifiers," *Measurement*, vol. 46, no. 7, pp. 2082–2092, 2013.
- [16] M. G. Poddar, V. Kumar, and Y. P. Sharma, "Linear-nonlinear heart rate variability analysis and SVM based classification of normal and hypertensive subjects," *Journal of Electrocardiology*, vol. 46, no. 4, p. e25, 2013.
- [17] Y. C. Wei and C. H. Lin, "A robust video text detection approach using SVM," *Expert Systems with Applications*, vol. 39, no. 12, pp. 10832–10840, 2012.
- [18] X. M. Chen, H. B. Gong, and J. N. Wang, "BRT vehicle travel time prediction based on SVM and Kalman filter," *Journal of Transportation Systems Engineering and Information Technology*, vol. 12, no. 4, pp. 29–34, 2012.
- [19] O. Duru, "A fuzzy integrated logical forecasting model for dry bulk shipping index forecasting: an improved fuzzy time series approach," *Expert Systems with Applications*, vol. 37, no. 7, pp. 5372–5380, 2010.
- [20] B. Yu and Z. Z. Yang, "An ant colony optimization model: the period vehicle routing problem with time windows," *Transportation Research Part E*, vol. 47, no. 2, pp. 166–181, 2011.
- [21] W. Huang, Y. Nakamori, and S.-Y. Wang, "Forecasting stock market movement direction with support vector machine," *Computers and Operations Research*, vol. 32, no. 10, pp. 2513–2522, 2005.
- [22] K. K. Seo, "An application of one-class support vector machines in content-based image retrieval," *Expert Systems with Applications*, vol. 33, no. 2, pp. 491–498, 2007.
- [23] B. Wohlberg, D. M. Tartakovsky, and A. Guadagnini, "Sub-surface characterization with support vector machines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 1, pp. 47–57, 2006.
- [24] B. Yu, Z. Z. Yang, K. Chen, and B. Yu, "Hybrid model for prediction of bus arrival times at next station," *Journal of Advanced Transportation*, vol. 44, no. 3, pp. 193–204, 2010.
- [25] B. Yu, J. B. Yao, and Z. Z. Yang, "An improved headway-based holding strategy for bus transit," *Transportation Planning and Technology*, vol. 33, no. 3, pp. 329–341, 2010.
- [26] B. Yu, W. H. K. Lam, and M. L. Tam, "Bus arrival time prediction at bus stop with multiple routes," *Transportation Research Part C*, vol. 19, no. 6, pp. 1157–1170, 2011.
- [27] T. van Gestel, J. A. K. Suykens, D. E. Baestaens et al., "Financial time series prediction using least squares support vector machines within the evidence framework," *IEEE Transactions on Neural Networks*, vol. 12, no. 4, pp. 809–821, 2001.
- [28] L. J. Cao and F. E. Tay, "Support vector machine with adaptive parameters in financial time series forecasting," *IEEE Transactions on Neural Networks*, vol. 14, no. 6, pp. 1506–1525, 2003.
- [29] K. J. Kim, "Financial time series forecasting using support vector machines," *Neurocomputing*, vol. 55, no. 1-2, pp. 307–319, 2003.
- [30] B. Yu, B. Yu, J. Lu, and Z. Z. Yang, "An adaptive bus arrival time prediction model," *Proceedings of the Eastern Asia Society for Transportation Studies*, vol. 7, 2009.

- [31] M. Esteban, C. Ariño, and J. M. Díaz-Cruz, "Chemometrics for the analysis of voltammetric data," *TrAC Trends in Analytical Chemistry*, vol. 25, no. 1, pp. 86–92, 2006.
- [32] Z. Z. Yang, L. J. Jin, and M. H. Wang, "Forecasting Baltic Panamax index with Support Vector Machine," *Journal of Transportation Systems Engineering and Information Technology*, vol. 11, no. 3, pp. 50–57, 2011.
- [33] P. Du, K. Tan, and X. Xing, "Wavelet SVM in Reproducing Kernel Hilbert Space for hyperspectral remote sensing image classification," *Optics Communications*, vol. 283, no. 24, pp. 4978–4984, 2010.
- [34] I. Turkoglu and E. Avci, "Comparison of wavelet-SVM and wavelet-adaptive network based fuzzy inference system for texture classification," *Digital Signal Processing*, vol. 18, no. 1, pp. 15–24, 2008.
- [35] G. Y. Chen and W. F. Xie, "Pattern recognition with SVM and dual-tree complex wavelets," *Image and Vision Computing*, vol. 25, no. 6, pp. 960–966, 2007.
- [36] H. Keskes, A. Braham, and Z. Lachiri, "Broken rotor bar diagnosis in induction machines through stationary wavelet packet transform and multiclass wavelet SVM," *Electric Power Systems Research*, vol. 97, pp. 151–157, 2013.
- [37] Y. Zheng, L. Zhu, and X. Zou, "Short-term load forecasting based on Gaussian wavelet SVM," in *Proceedings of the 1st International Conference on Smart Grid and Clean Energy Technologies (ICSGCE '11)*, pp. 387–393, September 2011.
- [38] B. Yu, Z. Yang, and S. Li, "Real-time partway deadheading strategy based on transit service reliability assessment," *Transportation Research Part A*, vol. 46, no. 8, pp. 1265–1279, 2012.
- [39] V. Fernandez, "Wavelet- and SVM-based forecasts: an analysis of the U.S. metal and materials manufacturing industry," *Resources Policy*, vol. 32, no. 1-2, pp. 80–89, 2007.
- [40] Q. Wu, "The forecasting model based on wavelet ν -support vector machine," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7604–7610, 2009.
- [41] Q. Wu and R. Law, "An intelligent forecasting model based on robust wavelet ν -support vector machine," *Expert Systems with Applications*, vol. 38, no. 5, pp. 4851–4859, 2011.
- [42] F. Y. Liu and M. Fan, "A hybrid support vector machines and discrete wavelet transform model in futures price forecasting," in *Advances in Neural Networks*, vol. 3973 of *Lecture Notes in Computer Science*, pp. 485–490, 2006.
- [43] X. Wang, Q. Fan, C. Xu, and Z. Li, "Dam deformation prediction based on wavelet transform and support vector machine," *Geomatics and Information Science of Wuhan University*, vol. 33, no. 5, pp. 469–507, 2008.
- [44] V. Vapnik, M. R. Müller, A. J. Smola, G. Rätsch, B. Schölkopf, and J. Kohlmorgen, "Predicting time series with support vector machines," in *Artificial Neural Networks*, vol. 1327 of *Lecture Notes in Computer Science*, pp. 999–1004, Springer, Berlin, Germany, 1997.
- [45] B. Z. Yao, C. Y. Yang, J. B. Yao, and J. Sun, "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.
- [46] J. B. Yao, B. Z. Yao, L. Li, and Y. L. Jiang, "Hybrid model for displacement prediction of tunnel surrounding rock," *Neural Network World*, vol. 22, no. 3, pp. 263–275, 2012.
- [47] V. Cherkassky and Y. Ma, "Practical selection of SVM parameters and noise estimation for SVM regression," *Neural Networks*, vol. 17, no. 1, pp. 113–126, 2004.
- [48] B. Yao, C. Yang, J. Hu, J. Yao, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6-7, pp. 2127–2131, 2011.
- [49] B. Z. Yao, P. Hu, X. H. Lu, J. J. Gao, and M. H. Zhang, "Transit network design based on travel time reliability," *Transportation Research Part C*, 2014.
- [50] B. Z. Yao, P. Hu, M. H. Zhang, and S. Wang, "Artificial bee colony algorithm with scanning strategy for periodic vehicle routing problem," *SIMULATION: Transactions of the Society for Modeling and Simulation International*, vol. 89, no. 6, pp. 762–770, 2013.
- [51] B. Z. Yao, P. Hu, M. H. Zhang, and X. M. Tian, "Improved ant colony optimization for seafood product delivery routing problem," *Promet: Traffic & Transportation*, vol. 26, no. 1, pp. 1–10, 2014.
- [52] A. C. Lorena and A. C. P. L. F. de Carvalho, "Evolutionary tuning of SVM parameter values in multiclass problems," *Neurocomputing*, vol. 71, no. 16-18, pp. 3326–3334, 2008.

Research Article

Applications of PCA and SVM-PSO Based Real-Time Face Recognition System

Ming-Yuan Shieh, Juing-Shian Chiou, Yu-Chia Hu, and Kuo-Yang Wang

Department of Electrical Engineering, Southern Taiwan University of Science and Technology, Tainan City 710, Taiwan

Correspondence should be addressed to Juing-Shian Chiou; jschiou@mail.stust.edu.tw

Received 25 February 2014; Accepted 23 April 2014; Published 13 May 2014

Academic Editor: Rui Mu

Copyright © 2014 Ming-Yuan Shieh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper incorporates principal component analysis (PCA) with support vector machine-particle swarm optimization (SVM-PSO) for developing real-time face recognition systems. The integrated scheme aims to adopt the SVM-PSO method to improve the validity of PCA based image recognition systems on dynamically visual perception. The face recognition for most human-robot interaction applications is accomplished by PCA based method because of its dimensionality reduction. However, PCA based systems are only suitable for processing the faces with the same face expressions and/or under the same view directions. Since the facial feature selection process can be considered as a problem of global combinatorial optimization in machine learning, the SVM-PSO is usually used as an optimal classifier of the system. In this paper, the PSO is used to implement a feature selection, and the SVMs serve as fitness functions of the PSO for classification problems. Experimental results demonstrate that the proposed method simplifies features effectively and obtains higher classification accuracy.

1. Introduction

There are numerous approaches that develop useful schemes to detect and recognize the features of human faces in recent years. They are used to filter the background and detect the faces blocks from a digital image firstly, then to determine their features and generate characteristic vectors, to localize the faces continuously, and to recognize the main face finally. In general, the approaches of human face image processing consist of two fields as face detection and face recognition. For human-robot interaction, to recognize faces is much more difficult than to detect faces. It is because the facial expression and features are changeable and not easily recognized and predicted.

There are two standard linear subspace projections of the low-resolution facial images [1], the PCA and the linear discriminant analysis (LDA) [2], used to distinguish the different styles of images. The PCA is basically a compression procedure based on linear projection techniques on a subspace spanned by the principal eigenvectors (those corresponding to the largest eigenvalues) of the input covariance

matrix. The LDA approach was proposed by Fisher firstly which identifies directions in space along which separation of the projections is maximized. While the LDA is not always superior to the PCA in terms of recognition accuracy, the PCA + LDA approach has been successfully applied in some face recognition applications.

However, the PCA and/or LDA based facial recognition may fail if the facial samples are captured in different directions. While the face images are captured from different sight depths and directions, the accuracy of the PCA based recognition will be reduced heavily. There are many face recognition methods provided to overcome such problems, some focus on the feature extraction. How to use the smallest dimensions to replace the most representation's feature and classification is the most important issue in this paper.

Since the facial feature selection process can be considered as a problem of global combinatorial optimization in machine learning, the PSO-SVM is usually used as an optimal classifier of the system. In the part of the classification, the PSO is used to implement feature selections, and the SVMs [3, 4] serve as fitness functions of the PSO for the classification

problem. The advantage of the SVM in the multidimensional space is that it can quickly and correctly classify samples to find out the best support vector [5]. The PSO algorithm is a kind of imitation of birds clustering phenomenon algorithm [6–9], and in this paper, the PSO will be issued to correct the parameters of the SVM so that the image recognition process can be faster and more stable.

The rest of this paper is organized as follows: in Section 2, the image detection scheme is described which consists of smoothing filter, connected position, and ellipse detection. Section 3 describes the face recognition system which includes the adjustment of the dimensions, the PCA, and the PSO-SVM classifier. Section 4, experimental results are presented to demonstrate the feasibility of the proposed scheme. Finally, some conclusions are made in Section 5.

2. The Face Detection System

In order to detect the face from an image, there are several key steps necessarily processed. The first is to detect the area of skin color; the second is to reduce the noise; the others are to distinguish which one or ones are the face by ellipse detection and to separate the face block from the image background. Figure 1 illustrates the flowchart of the proposed face detection scheme. The images captured by the webcam in sequence will be sent to the face detection system firstly, and then the area of the human face is separated from the complex background by the skin-color detection. Next, the noise is going to be filtered by the noise reduction. Finally, the captured contours will aid in locating the position of the human face. The block of the human face will only remain in the image after the face detection.

2.1. Skin-Color Detection. The RGB image is always with chromatism because of changeable illumination in every capture. For skin-color detection, a reliable color range defined as the skin-color is principal. Since the color values in RGB are so sensitive to varying illumination, most approaches adopt the color model of YCbCr replacing the RGB because the value Y is relative to luminance and the values of Cb and Cr are relative to chroma.

YCbCr color space can be regarded as a modified YUV color model. However, YCbCr is not an absolute color space which is a way of encoding RGB information. In YCbCr color space, Y is the luma component; Cb and Cr represent the blueness and the redness chroma components, respectively. The transform defined in the ITU-R BT.601 standard for digital component video the YCbCr color space can be translated from the RGB color space by

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.0 \\ 112.0 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix}. \quad (1)$$

The resultant Y is just between 16 and 235 because the values from 0 to 15 are called footroom and the values from 236

to 255 are called headroom. Besides, in this paper, the skin-color region of Cb calculated is between 71 and 127; the skin-color region of Cr is between 130 and 170 from the 150 sample images. By the skin-color regions, the human face block will be easily distinguished from the image according to the values of Cb and Cr.

The low-pass filter (LPF) was used to eliminate the small noises and connect the part of the incomplete image. The LPF makes the image for filtering become more smooth and uniform. In general, the start of the mask is set to be at the top left pixel of the preprocess image, and it will scan the whole image from left to right. It is also referred to the neighboring 8 pixels for processing. Generally the sizes of the masks are 3×3 , 5×5 , and 7×7 . The larger the mask, the larger the filtering effect, but the calculation becomes relatively large. The 3×3 mask was used for low-pass filtering in this paper.

Besides, the opening operation of the morphology is usually used to eliminate noises in an image. The opening operation included two operands with erosion and dilation. First, it will use the erosion for the binary image and then use the dilation for its result. After this procedure, the noises will be removed from the image.

The opening operation of the morphology can not only remove a part of the noise pixels of the binary image but also make more complete region of the skin-color.

After finishing the procedure of the noise reduction, the connected component labeling (CCL) was used to find the location of the human face. This method was mainly used to find the connected pixels of the same object in the image. It marks each block by different labels and counts size, height, and width of each independent object in the image. The 4-connected was used to label pixels in this paper. It starts to scan the prelabeled binary image from top left. In the coordinates (x, y) of the pixel, determine the presence of 255 on a pixel before checking the right $(x + 1, y)$, left $(x - 1, y)$, top $(x, y + 1)$, and bottom $(x, y - 1)$, for any other 255 values. If yes, it will record its coordinates and the pixel is set to 0. Then there is the recursive process of checking all the pixels for presence of 255 valued pixels until none is present. Then the group with the most number of 255 valued pixels is searched and labeled.

While finishing the recursive scanning of the image, the group objects for image labeling were calculated. The biggest area of the region was found from all numbered regions, that is, the region of the human face. This region was scanned to find the black background of the facial boundary. Then the relative coordinates (X_{\min}, Y_{\min}) and (X_{\max}, Y_{\max}) were employed, to capture the facial region from the original image. The biggest region for labeling of the binary image is shown in Figure 2.

The Sobel edge detection [10] is used to detect the edge. For face detection, since there are obvious differences between the background and the region of the skin color in the facial binary image, the edge detection usually employed the first derivative of the image to estimate the regional edge and was used to calculate the size of the gradient for image processing. The Sobel edge detection employs the gray scale difference at the position of the edge and weighted the top and bottom, or left and right, to detect the edge of the object.

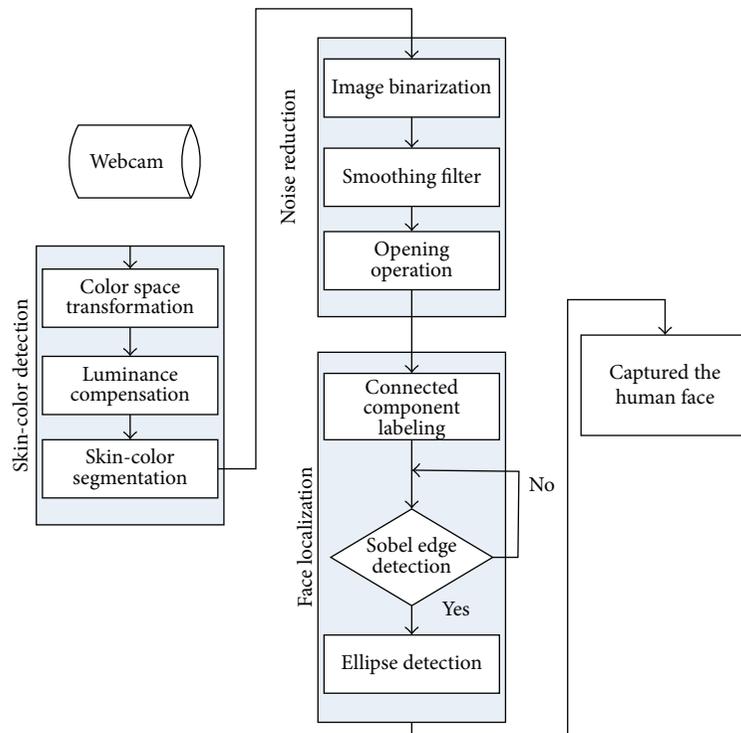


FIGURE 1: Face detection system flowchart.



FIGURE 2: The biggest region for labeling of the binary image.



FIGURE 3: The result of the Sobel edge detection.

Through the Sobel edge detection processing, the region of the face is shown in Figure 3.

Since the size of a human face is similar to the elliptical model with the ratio of the vertical axis and the horizontal axis as approximately 1.2:1, thus, an ellipse mask is used to locate the human face, which marks a boundary to extract the edge of the image. After the edge detection, the region that resembles more the shape of an ellipse will provide the position of the face. The center of the ellipse with the use of its circumference and the length of its axis can help to determine its position, shape, and size. In order to meet the size of the

facial change, the elliptical model must adjust the ratio of the length of the axis to scan the image in real time. Therefore, we designed an elliptical model that determines the coordinates of the center of the ellipse (x_0, y_0) with x as the radius of the horizontal axis and y for that of the vertical axis.

Face detection is then finished after the ellipse detection processing. The ellipse detection cooperates with the connected component labeling to find the facial region. The locations of the detected faces in the original image and the binary image are shown in Figures 3 and 4, respectively. Next, the unnecessary region outside of the elliptical range was

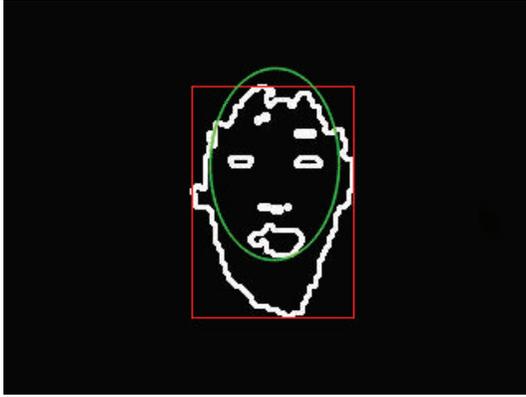


FIGURE 4: Face location of the binary image.



FIGURE 5: The result of the captured facial image.

removed. This procedure will help in eliminating the image of the neck and locate the facial region. The relative coordinate was used to capture the face for the original image. The result of the captured facial image is shown in Figure 5.

3. The Face Recognition System

3.1. Description of the Face Recognition System. The face recognition processing will be executed after finishing face detection. However, since the dimensions of the facial images are not the same, the normalization process for the facial image becomes important. The processed image to be analyzed contains the same information of the environment to make the dimensions of the facial regions the same for each face. After the image normalization processing, the captured image serves as the input data for the face recognition system. And then, principal component analysis is utilized to calculate the feature. This method could reduce the dimension of the image and save computation time. It could surmount the problems of the changed expression or

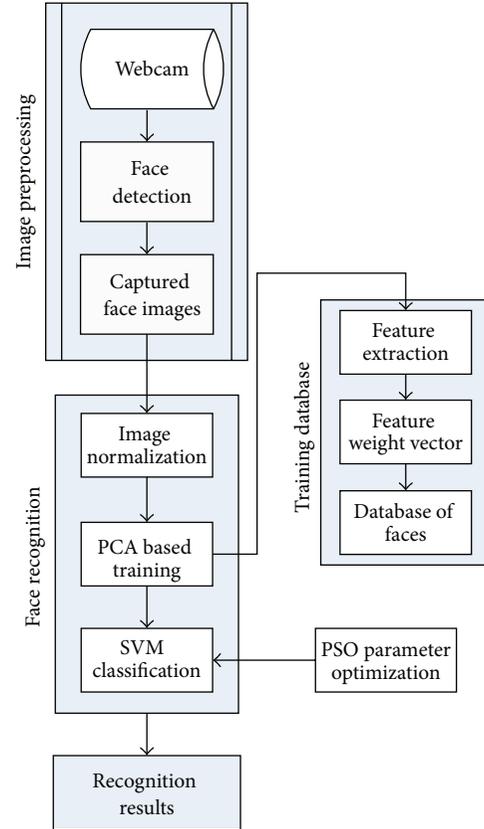


FIGURE 6: The scheme of the face recognition system.

presence of glasses on the face, because it treats the whole face as a feature.

The weighting vectors of the processed facial image will be calculated by using PCA. All the weighting vectors are collected to build the database for the face recognition system. The user identity could be determined by the support vector machine that compares the current image with the image in the database. This way is a fast and accurate classifier that can be applied to classification and comparison applications. It shows that the velocity and accuracy of the face recognition system can be increased through the support vector machine processed. At the same time, the particle swarm optimization (PSO) is used to design the parameter of the support vector machine. This method makes the face recognition system complete and quick. The face recognition system flowchart proposed in this paper is as shown in Figure 6.

3.2. Image Normalization. Achieving high recognition rate for the face recognition system does not only need a good recognition algorithm but also needs a robust face image for the preprocessing of the image. It could reduce the difference for each input image and changed each image to the same dimension for the database. Therefore the bilinear interpolation is used to amend the image that detects the face with the size of the pixel at 80×100 . Through the bilinear interpolation processing, feature extraction and recognition are executed for the facial image.

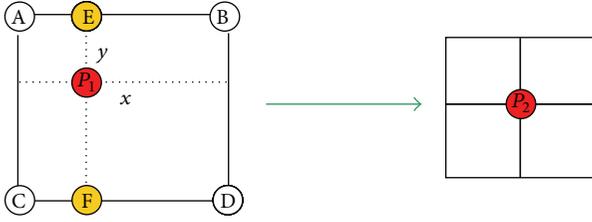


FIGURE 7: The diagram of the bilinear interpolation.

The relation of the pixel coordinate between the original image and final image is not the same as the general image processing when the image is zoomed. A pixel of the original image could project into many pixels for the final image, when the images are zoomed in. Similarly, many pixels of the original image could project onto a pixel for the final image, when the images are zoomed out. Therefore, the final image must search for a pixel to substitute the original image. And with this in mind, the interpolation process is needed to calculate the pixel of the final image; otherwise the final image would produce tremendous distortion. Actually the interpolation used discrete samplings to calculate the continuous function which passes through the coordinates of these samples and then employ this function to find the value of nonsamplings. The image zooming could define the signal resampling because the image was a signal that is composed of the two-dimensional discrete sampling. Then the bilinear interpolation adopted the four neighboring pixels to calculate the new pixel. The diagram of the bilinear interpolation is shown in Figure 7.

In order to obtain the pixel P_2 as a projection of the original image, P_2 is supposed to project onto P_1 ; thus the four neighboring pixels (A, B, C, D) were used to calculate the distances between P_1 and the coordinates of four pixels. If the four pixels are closer to P_1 then the contribution is large for P_2 . Conversely the influence is smaller if the distance is farther. Therefore the effect is inversely proportional to the distance. In fact, the bilinear interpolation calculated the linear interpolation continuously for three times. The first interpolation was calculated to be the influence between two points (A, B) and P_2 in order to obtain pixel E. The equation of the first interpolation is expressed in

$$E = (1 - x)A + xB. \quad (2)$$

And then the second interpolation was calculated using the influence between two points (C, D) and P_2 to obtain pixel F. The equation of the second interpolation is expressed in

$$F = (1 - x)C + xD. \quad (3)$$

Finally, the third interpolation was used to calculate the pixel of P_2 for two points (E, F). The equation of the third interpolation is expressed in

$$P_2 = (1 - y)E + yF, \quad (4)$$

where x represents the relative horizontal distance of P_1 in relation to the four neighboring pixels; y is the relative

vertical distance of P_1 corresponding to the four neighboring pixels. If the distance from pixel to pixel is one unit as assumed in this paper, it can be represented by $0 < x$ and $y < 1$, then the adjusted facial dimension is shown in Figure 8.

3.3. Principal Component Analysis. After normalization, if the face recognition is directly implemented it would cost a lot of computing time. This is due to the fact that all the information of the original image is spread in each pixel; hence, there is the need to reduce the dimensions of the image. And then, the suitable features are captured to express a lot of information in lower dimensions. It could reduce many variations for data with the use of PCA and applying some mutually independent linear combinations to substitute for the original data. Through the linear combination computing, the difference of the variation was the large influence of the data. This analysis made data to display the biggest individual differences. Below is the process for implementing PCA.

If there were N facial images as the training samples the original feature parameters were $\{X_1, X_2, \dots, X_N\}$. The objective of PCA was order to find the linear transformation matrix P with a size of $n \times m$. It extracts feature parameter X_k of the original n dimension to transform the more representative of the feature parameter Z_k that the dimension was m ($m \leq n$). The equation of the transformation expressed in

$$Z_k = P^T X_k, \quad k = 1, 2, \dots, N. \quad (5)$$

Before the transformation computing, the mean vector was \bar{X} . After the transformation, the mean vector was \bar{Z} expressed in

$$\bar{Z} = \frac{1}{N} \sum_{k=1}^N Z_k = \frac{1}{N} \sum_{k=1}^N P^T X_k = P^T \left(\frac{1}{N} \sum_{k=1}^N X_k \right) = P^T \bar{X}. \quad (6)$$

Then the total scatter matrix was used to indicate the dispersion of all feature parameters that were opposite to the mean vector \bar{X} . Before the transformation computation, the total scatter matrix was S_{t_x} which has a size of $n \times n$. The equation of the total scatter matrix S_{t_x} is expressed in

$$S_{t_x} = \sum_{k=1}^N (X_k - \bar{X})(X_k - \bar{X})^T. \quad (7)$$

Through (18) to (20), the total scatter matrix S_{t_z} can be obtained having a size of $m \times m$ after the transformation. The equation of the total scatter matrix S_{t_z} is expressed in

$$\begin{aligned} S_{t_z} &= \sum_{k=1}^N (Z_k - \bar{Z})(Z_k - \bar{Z})^T \\ &= \sum_{k=1}^N (P^T X_k - P^T \bar{X})(P^T X_k - P^T \bar{X})^T \\ &= P^T \left(\sum_{k=1}^N (X_k - \bar{X})(X_k - \bar{X})^T \right) P = P^T S_{t_x} P. \end{aligned} \quad (8)$$

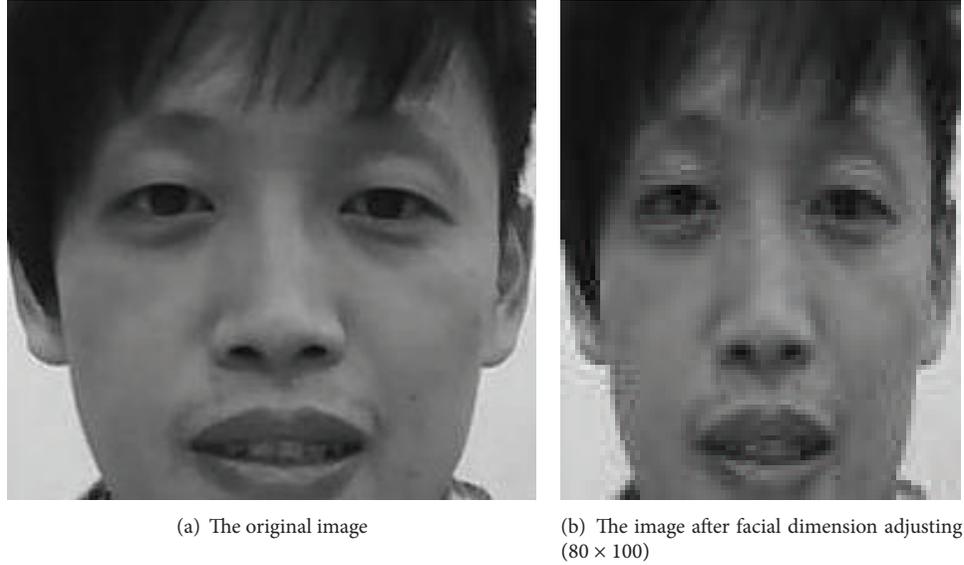


FIGURE 8: Facial dimension adjusting.

In order to increase the dispersion between the mean value and the feature parameters of the transformation, the transformation matrix P_{opt} should be calculated that could make the maximization of S_{t_z} . The equation of the transformation matrix P_{opt} is expressed in

$$P_{\text{opt}} = \arg \max_P (P^T S_{t_z} P). \quad (9)$$

According to the theory of linear algebra, the trace or the determinant can be used to express the element's distribution. Therefore, (9) could be rewritten as

$$P_{\text{opt}} = \arg \max_P \text{tr} (P^T S_{t_z} P). \quad (10)$$

In (10), in order to limit the value of $\text{tr} (P^T S_{t_z} P)$ and avoid the occurrence of infinity, a limit condition as $P^T P = I$ was added for the transformation matrix P that has a size of $n \times m$:

$$F(P) = P^T S_{t_z} P - \lambda (P^T P - I). \quad (11)$$

As follows, the value of $F(P)$ is maximum while the first derivative for P is zero:

$$\frac{\partial F(P)}{\partial P} = 2S_{t_z} P - 2\lambda P = 0. \quad (12)$$

Then, (12) becomes (13) through simplification using transposition:

$$(S_{t_z} - \lambda I) P = 0. \quad (13)$$

In (13), it needs to compute the eigenvectors of matrix S_{t_z} by composing the matrix for P . The eigenvalues and the eigenvectors are the special form in matrix algebra and its elements could be restructured in the matrix. In addition, the

important information would be concentrated in the larger eigenvalue that would correspond to the eigenvectors.

The advantages of PCA in face recognition can be divided into three advantages. First, it could quickly and easily calculate the result. Second, PCA could retain the largest information of the projection data in the linear projection. Third, PCA used the whole face to do feature extraction that could overcome the presence of glasses and the changes of the facial expression. Below is the operational procedure of PCA.

After normalization, M number of facial images was trained using PCA. The size of each sample was $N \times N$ matrix. Next, each sample is rearranged as the augmented vector Γ which has the size $N^2 \times 1$ as shown in (14). $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ represent the M facial images processed.

Each facial sample corresponded to Γ , and the mean vector Ψ was calculated by the M amount of Γ as expressed in

$$\Psi = \frac{1}{M} \sum_{k=1}^M \Gamma_k. \quad (14)$$

The mean vector Ψ is the mean face which indicates the mutual parts of all face. And then the mutual parts for M facial images were deleted to highlight the different parts between them. Therefore the different image vector of each image was obtained as shown in

$$\varphi_k = \Gamma_k - \Psi \quad k = 1, 2, 3, \dots, M, \quad (15)$$

wherein matrix A equals $[\varphi_1, \varphi_2, \varphi_3, \dots, \varphi_M]$ that had the size $N^2 \times M$ and the covariance matrix C of all faces was defined as

$$C = \frac{1}{M} \sum_{k=1}^M \varphi_k \varphi_k^T = AA^T. \quad (16)$$

The eigenvalue λ_k and the eigenvector u_k of the matrix C are expressed in

$$Cu_k = \lambda_k u_k \quad k = 1, 2, 3, \dots, N^2, \quad (17)$$

where $\lambda_k = (1/M) \sum_{j=1}^M (u_k^T \varphi_j)^2$ and $\varphi = \{\varphi_1, \varphi_2, \dots, \varphi_M\}$.

Since the size of matrix A was $N^2 \times M$, it makes the size of matrix C be $N^2 \times N^2$. For such a large matrix calculating the eigenvalues and eigenvectors is time consuming. Thus, if the dimensions of the matrix could be reduced, it could effectively save calculation time. Therefore, the matrix $A^T A$ must be calculated first and the dimensions of the matrix must be reduced as $M \times M$ to obtain the eigenvector v_k which expressed in

$$A^T A v_k = \mu_k v_k \quad k = 1, 2, 3, \dots, N^2. \quad (18)$$

Equation (18) multiplies by the matrix A to obtain

$$AA^T A v_k = \mu_k A v_k, \quad (19)$$

in which AA^T has the same eigenvalue and eigenvector with $A^T A$, because the matrix C equals AA^T . By comparing (17) and (19), (20) can be obtained as follows:

$$u_k = A v_k \quad \lambda_k = \mu_k. \quad (20)$$

By using (18), the matrix of $A^T A$ is used to calculate the eigenvector v_k which determines the eigenvalue u_k . It is considered as an eigenface as expressed in

$$u_k = \sum_{j=1}^M \varphi_j v_{kj}, \quad i = 1, 2, \dots, M. \quad (21)$$

The vector $\Gamma_1, \Gamma_2, \dots, \Gamma_M$ of the individual facial image combined with the corresponding eigenvector to build the feature space. And we calculate the weight vector V from the feature space as expressed in

$$\Omega_k = u_k^T (\Gamma - \Psi) = u_k^T \varphi \quad V = [\varphi_1, \varphi_2, \varphi_3, \dots, \varphi_M] \quad (22)$$

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

Finally, each training sample Γ_i of the face is inputted to substitute the Γ of (22) and calculate the eigenvector V_i in the feature space. Through the computation, the matrix V will be taken as the database of the facial images after the training.

3.4. SVM Based Classification for Face Recognition. Euclidean distance based methods [11] aim to calculate the difference value of the distance measurement, which are usually used in pattern recognition system. The resemblance computation directly calculates the difference between two vectors. The smaller value means that the two vectors are closer. It also indicates that the features of two images are also closer, and

there is the presence of similarity in the images. The equation of the Euclidean distance is expressed in

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_M \end{bmatrix} = \begin{bmatrix} \sqrt{\sum_{i=1}^k (a_{i1} - b_{i1})^2} \\ \sqrt{\sum_{i=1}^k (a_{i2} - b_{i2})^2} \\ \vdots \\ \sqrt{\sum_{i=1}^k (a_{iM} - b_{iM})^2} \end{bmatrix}, \quad (23)$$

in which $d_1, d_2, d_3, \dots, d_M$ are the Euclidean distances between the eigenvector of each image and the eigenvector of the target image; a_i is the i th element of the input eigenvector; b_i is the i th element of the eigenvector saved in the database; k is the dimension of the eigenvector; and M is the M th image saved in the database.

In general, if the Euclidean distance method is directly used in face recognition system, it would require a lot of computation time, because the Euclidean distance applies bubble sort for comparison. For example, if there are one thousand data in the database, it will require the process to be performed one thousand times and the larger the database the longer the comparison time. Therefore, support vector machine is used to assist such face recognition problems. The calculated Euclidean distances are inputted the feature space of the support vector machine to perform the comparison and classification.

The most important goal of face recognition system is how to raise the accuracy and shorten the computing time of the system. In fact, principal component analysis could indeed raise the accuracy as shown in previous experiments. However, the comparison of the Euclidean distance would require a lot of computing time. Therefore, the design of the support vector machine classification shortens the program's computing time for the face recognition system. The SVM is performed through the following process. First, the hyper-plane is designed to classify the Euclidean distance as shown in Figure 9.

In the figure, d_i denotes the Euclidean distance between the target image and i th image of the database. There are M images in the database with $i = 1, 2, 3, \dots, M$. Besides, d_{mean} indicates the mean value of the Euclidean distance between the target image and all images in the database. The equation of d_{mean} is expressed in

$$d_{\text{mean}} = \frac{\sum_{i=1}^M d_i}{M}. \quad (24)$$

In this method, the training data is shown:

$$(d_1, y_1), \dots, (d_M, y_M), \quad d_1 \in R^n \quad (25)$$

$$i = 1, 2, \dots, M \quad y_i \in d_{\text{mean}}.$$

Then, consider

$$(w \cdot d_i) + b \geq d_{\text{mean}} \longrightarrow y_i = d_{\text{mean}}, \quad i = 1, 2, \dots, M \quad (26)$$

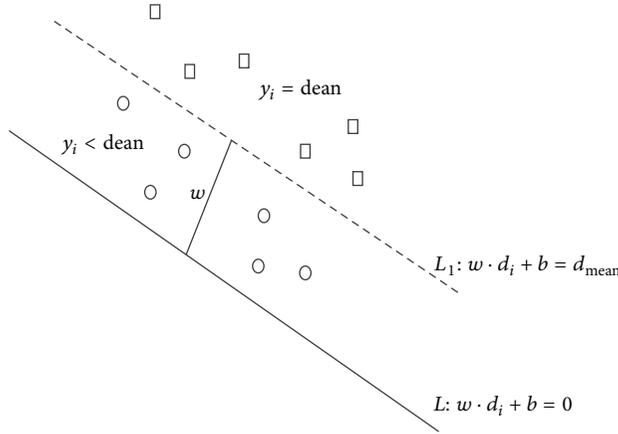


FIGURE 9: Hyperplane of the Euclidean distance.

in which w is the normal vector of the hyperplane and b is the deviation value. In order to find the division of the hyperplane the question of quadratic optimization needs to be resolved. The constraint is expressed in

$$y_i (w \cdot d_i - b) \geq d_{\text{mean}}, \quad i = 1, 2, \dots, M. \quad (27)$$

The minimum value of $d(w) = (1/2)\|w\|^2$ must be determined because the equation above is quadratic with a linear constraint. This is a typical quadratic optimization problem. So, the Lagrange multiplier is resolved to the question of quadratic optimization with linear constraint to obtain

$$L(w, b, \alpha) = \frac{1}{2}w^2 - \sum_{i=1}^M \alpha_i [y_i (w \cdot d_i + b) - d_{\text{mean}}] \quad \alpha_i \geq 0. \quad (28)$$

However, the support vector machine still does not produce the optimal solution. The method in which the problem is dealt with was to address the dual question. The dual question is expressed in

$$\begin{aligned} \frac{\partial L}{\partial w} = w - \sum_{i=1}^M \alpha_i y_i d_i = 0 &\longrightarrow w = \sum_{i=1}^M \alpha_i y_i d_i. \\ \frac{\partial L}{\partial b} = \sum_{i=1}^M \alpha_i y_i = 0. \end{aligned} \quad (29)$$

And a new equation was left after performing the substitution as expressed in

$$L_D = \sum \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j d_i d_j. \quad (30)$$

After having determined the optimal solution to the dual question, each Lagrange modulus α_i is mapped onto each trained data. If $\alpha_i \geq 0$, this means that the data is the support vector of this question and it is located in the margin separating the hyperplane. The final function is expressed in

$$f(d) = \text{sgn} \left[\sum_{i=1}^M \alpha_i y_i (d_i d) + b \right]. \quad (31)$$

By the method of support vector machine, d_i found to be located between zero and d_{mean} in the hyperplane are retained. These data are used to recalculate the d_{mean} and reexecute the classification by the same way. This procedure would be repeated until data are all plotted before terminating the program. This data represents the Euclidean distance closest to the target image. Finally, this data shows the image which is the recognition results needed.

3.5. PSO-SVM Classifier. The particle swarm optimization (PSO) was proposed by [12–15]. This method is a concept of swarm intelligence which belongs to a territory of the evolutionary search. This algorithm is an evolutionary optimization implementation similar to the genetic algorithm (GA). First, they could produce the initial solution and apply evolution to find the optimal solution. The difference is that PSO does not have the procedures of crossover and mutation. It belongs to the signal-channel messaging, and the process of searching update is changed according to the current optimal solution. Therefore, in the general optimal questions, the PSO converges to the optimal solution more quickly than the GA.

The origin of the PSO is from the concept of the predation on bird populations. Kennedy used this concept to research the solution of the optimal question, and this question is just like a bird which flies in space, called particle. There is a fitness function of the objective function mapping for all particles that moved in the space. In addition, each particle has the velocity to determine the direction and the distance of the movement. The particles fly in the solution space by the individual successful experience and the trajectory of the best particle in the current population. In addition, each particle could search independently in the PSO space. When the individual particle found the optimization of the function, the best search variable will be recorded in the individual memory. Thus, each particle owns the best memory of the search variable for itself. It would change the next search direction by the individual best memory of the search variable, and this procedure is called the cognition-only model. Every search would compare the optimization extent between the individual best search variable and the best search variable of the population. This procedure would adjust the variable memory of the best function for the population. At the same time, each particle could change the search velocity of the particle for next time according to the best variable memory of the population, and this process was called the social-only model. Through the evolutionary computation, the PSO would calculate the optimal solution according to the best fitness value of the particles [16]. The flowchart of the PSO is shown in Figure 10.

In the space of the SVM, it requires the design of an important parameter w . Therefore, PSO is applied to optimize this parameter. The particle's position in the PSO space was used to substitute the parameter w of the SVM space. Through the evolutionary computation, each particle would update the position and the parameter w would be updated continually too. By this procedure, we could find the optimization value of the parameter w .

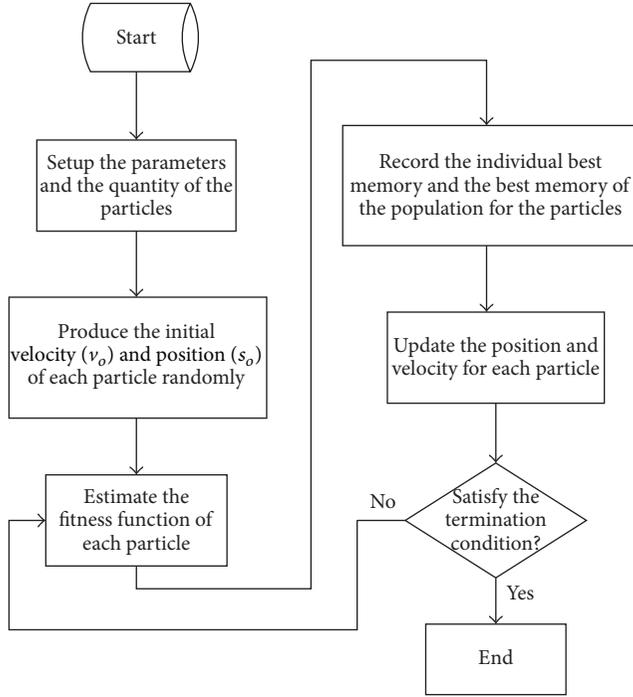


FIGURE 10: The flowchart of the PSO.

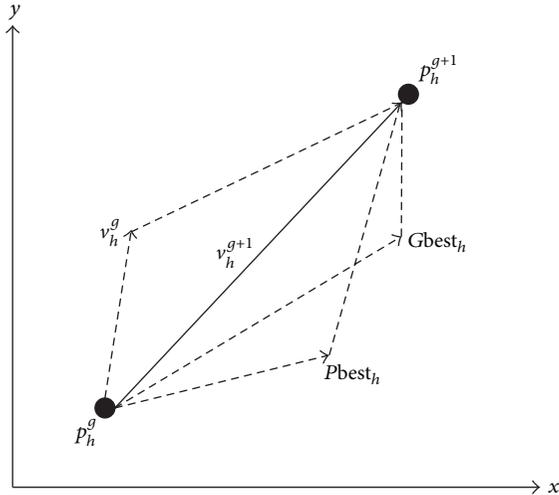


FIGURE 11: The PSO search in the space.

The PSO produces the particles of the initial population randomly and through the evolutionary computation to find the optimal solution for the function. In each evolution, the particle would change the individual search direction by two search memories. The first search is the optimal individual variable memory $Pbest$ and the other is the optimal variable memory of the population $Gbest$. After the computation, the PSO would calculate the optimal solution according to the optimal variable memory. Figure 11 shows the PSO search in a particular space.

Having a range of $x \in [0, 20]$ and $y \in [0, 20]$, it is supposed that the coordinate of particle's position was

(p_{hx}^g, p_{hy}^g) ; therefore the parameter w_h^g of the SVM could be calculated by

$$w_h^g = \frac{\sqrt{(p_{hx}^g)^2 + (p_{hy}^g)^2}}{28.29}. \quad (32)$$

If the set p^g with N particles is called the population in the g th generation, it can be expressed in

$$p^g = (p_1^g, p_2^g, \dots, p_h^g, \dots, p_N^g). \quad (33)$$

The velocity vector and position vector of the h th particle ($h \in [1, 2, 3, \dots, N]$) in the g th generation ($g \in [1, 2, 3, \dots, G]$) are expressed in (34) and (35), respectively, as follows:

$$v_h^g = (v_1^g, v_2^g, \dots, v_h^g, \dots, v_N^g), \quad (34)$$

$$p_h^g = (p_1^g, p_2^g, \dots, p_h^g, \dots, p_N^g), \quad (35)$$

in which the position of the h th particle in the g th generation is p_h^g . Also, the processes of the PSO could be explained as follows.

Step 1. The initialization of the PSO was set to $g = 1$, $F_1^{pbest} = F_2^{pbest} = \dots = F_N^{pbest} = 0$. The number of the particles (N), the number of the generation (G), and four parameters of C_1 , C_2 , γ_{max} , and γ_{min} are given.

Step 2. The initial velocity $v_h^1 = (v_1^1, v_2^1, \dots, v_h^1, \dots, v_N^1)$ and the initial position $p_h^1 = (p_1^1, p_2^1, \dots, p_h^1, \dots, p_N^1)$ of N particles are created.

Step 3. The fitness value of each particle in the g th generation is calculated by using (36). $fit(\cdot)$ is the fitness function which is expressed by the reciprocal computing time of the recognition system:

$$F(p_h^g) = fit(p_h^g), \quad h = 1, 2, \dots, N. \quad (36)$$

Step 4. The F_h^{Pbest} and p_h^{Pbest} for each particle were determined and the equation of F_h^{Pbest} is expressed in (37), and the equation of p_h^{Pbest} is expressed in (38) as follows:

$$F_h^{Pbest} = \begin{cases} F_h^g, & \text{if } F_h^{Pbest} \leq F_h^g \\ F_h^{Pbest}, & \text{otherwise,} \end{cases} \quad h \in \{1, 2, \dots, N\}, \quad (37)$$

$$p_h^{Pbest} = \begin{cases} p_h^g, & \text{if } F_h^{Pbest} \leq F_h^g \\ p_h^{Pbest}, & \text{otherwise,} \end{cases} \quad h \in \{1, 2, \dots, N\}, \quad (38)$$

where, p_h^{Pbest} was the individual optimal fitness value F_h^{Pbest} form the starting to the current generation.

Step 5. The index q of the particle with the highest fitness function is designed by

$$q = \arg \max_{h \in \{1, 2, \dots, N\}} F_h^{Pbest}. \quad (39)$$

And then, $F^{G_{\text{best}}}$ and $p^{G_{\text{best}}}$ are determined by

$$F^{G_{\text{best}}} = F_q^{P_{\text{best}}} = \max_{h \in \{1, 2, \dots, N\}} F_h^{P_{\text{best}}}, \quad p^{G_{\text{best}}} = p_q^{P_{\text{best}}}, \quad (40)$$

in which $p^{G_{\text{best}}}$ is the position vector of the particle with the global optimal fitness value $F^{G_{\text{best}}}$ from the starting to the current generation.

Step 6. If $g = G$, and then go to Step 10, otherwise go to Step 7.

Step 7. The velocity vector is updated for each particle by

$$v_h^{g+1} = \gamma \cdot v_h^g + c_1 \cdot \text{rand1}() \cdot (p_h^{P_{\text{best}}} - p_h^g) + c_2 \cdot \text{rand2}() \cdot (p^{G_{\text{best}}} - p_h^g), \quad (41)$$

where v_h^g is the current velocity vector of the h th particle in the g th generation. v_h^{g+1} is the next velocity vector of the h th particle in the $(g + 1)$ th generation. $\text{rand1}()$ and $\text{rand2}()$ are two uniformly distributed random numbers in $[0, 1]$. C_1 and C_2 are the constant values that are set to 2. γ was the weight value which is defined by

$$\gamma = \gamma_{\max} - \frac{\gamma_{\max} - \gamma_{\min}}{G} \cdot g, \quad (42)$$

where γ_{\min} and γ_{\max} are, respectively, the minimum value and the maximum value of γ , and the γ_{\min} is set to 0.4; the γ_{\max} is set to 0.9.

Step 8. Then the position vector is updated for each particle by

$$p_h^{g+1} = p_h^g + v_h^{g+1}, \quad (43)$$

where p_h^g is the current position vector of the h th particle in the g th generation. p_h^{g+1} is the next position vector of the h th particle in the $(g + 1)$ th generation.

Step 9. Let $g = g + 1$ and go to Step 3.

Step 10. The optimal position vector of the particle $p^{G_{\text{best}}}$ with the optimal fitness value $F^{G_{\text{best}}}$ is determined.

After the above procedures, the particle moves in the generation that would create the new parameter w for the SVM. The computing time of the face recognition system is compared for each parameter w , and the best w in current generation is searched for. It projects the space of the PSO to be the optimal global solution for the next generation. Through this method, the best parameter w of the SVM could be found to reduce the computing time of the face recognition system.

4. Experimental Results

4.1. Actual Experiments. In the experiments of the face recognition system, the facial database used contained ten

TABLE 1: The comparison with the experiments with fifty samples.

	PCA + ED	PCA + SVM	PCA + SVM + PSO
Sample numbers	50	50	50
Test times	80	80	80
Successful numbers	74	76	76
Success rate	93%	95%	95%
Average training time	0.179 s	0.177 s	0.177 s
Average computing time	0.149 s	0.137 s	0.129 s

TABLE 2: The comparison with the experiments with one hundred samples.

	PCA + ED	PCA + SVM	PCA + SVM + PSO
Sample numbers	100	100	100
Test times	80	80	80
Successful numbers	71	73	74
Success rate	89%	91%	93%
Average training time	0.344 s	0.345 s	0.344 s
Average computing time	0.296 s	0.203 s	0.180 s

people as training samples. Each person has ten facial images used as the input samples, and then there are one hundred test samples in the real-time face recognition system. Through the real-time detection, the current facial image is captured to be the test face. The PCA-SVM-PSO algorithm is used to execute the face recognition system. The Euclidean distance between the test face and the samples of the facial database would be classified, and then it would find the sample in the face database in which the Euclidean distance is closer to the test face to be the result. The experiments of the real-time face recognition system are shown in Figure 12.

4.2. Experimental Comparison. In the part of the comparison with the experiments, the main comparisons made are on the training time, computing time, and recognition rate for three ways. The first way is using PCA and Euclidean distance with the bubble sort to the face recognition system. The second is to apply the PCA and SVM based classification to the face recognition system. The third is to adopt PSO as a parameter adjusted scheme for SVM based classification. There are fifty samples and one hundred samples used in the analyses of experimental results. Table 1 expresses the comparison with the experiments with fifty samples. Table 2 summarizes the comparison with the experiments with one hundred samples, in which ED denotes the Euclidean distance with the bubble sort.

From the results in Tables 1 and 2, one can see that the proposed algorithm has really raised the recognition

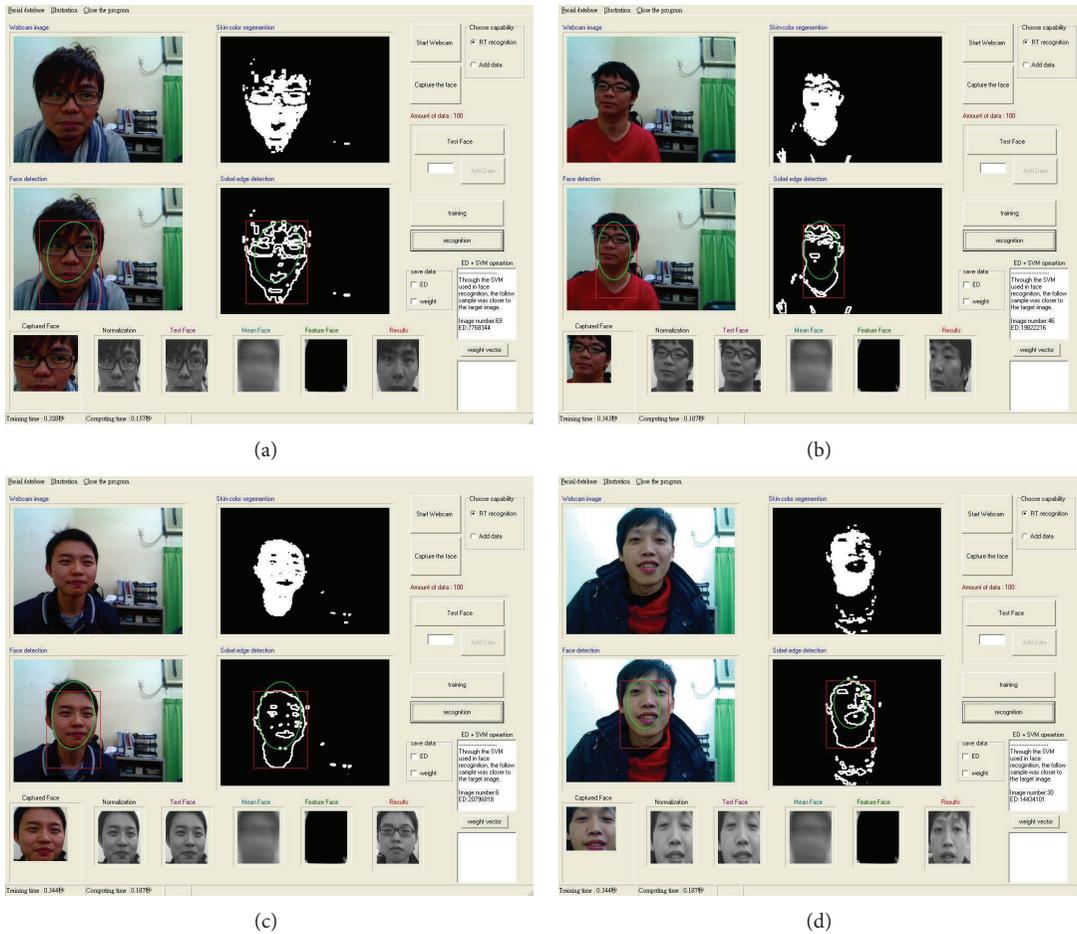


FIGURE 12: The experimental results of the real-time face recognition system.

rate and reduced the computing time for the real-time face recognition system. For the method of combining PCA and Euclidean distance with bubble sort, when the number of samples is doubled from 50 to 100, the average computing time also increases almost linearly from 0.149 s to 0.296 s. It is noted that the time needed for the way of the PCA combined SVM-PSO only increases by 40% correspondingly. Based on such view, one can say that the proposed algorithm is clearly superior for large-sample-size cases. It also concludes that the proposed method is faster and more efficient than other common methods for face recognition.

5. Conclusions

A real-time face recognition system is designed by using a combination of PCA and hybrid biology algorithm face recognition system application and this method has effectively reduced the computing time. There is a time savings of 60% after doubling samples from 50 to 100 samples as compared to other methods. Furthermore, the SVM-PSO scheme is designed to speed up the recognition and also enhances the performance of the face recognition. In the future, the result of the face recognition system can be further developed in a chip of an embedded system.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work is supported by the National Science Council, Taiwan, under Grant nos. NSC 102-2221-E-218-017 and NSC 100-2632-E-218-001-MY3.

References

- [1] M. M. Dehshibi and A. Bastanfard, "A new algorithm for age recognition from facial images," *Signal Processing*, vol. 90, no. 8, pp. 2431–2444, 2010.
- [2] W.-J. Zeng, X.-L. Li, X.-D. Zhang, and E. Cheng, "Kernel-based nonlinear discriminant analysis using minimum squared errors criterion for multiclass and undersampled problems," *Signal Processing*, vol. 90, no. 8, pp. 2333–2343, 2010.
- [3] J. Zhang and L. Ye, "Local aggregation function learning based on support vector machines," *Signal Processing*, vol. 89, no. 11, pp. 2291–2295, 2009.

- [4] Y. E. Shao, C.-J. Lu, and Y.-C. Wang, "A hybrid ICA-SVM approach for determining the quality variables at fault in a multivariate process," *Mathematical Problems in Engineering*, vol. 2012, Article ID 284910, 12 pages, 2012.
- [5] J.-S. Chiou and K.-Y. Wang, "Application of a hybrid controller to a mobile robot," *Simulation Modelling Practice and Theory*, vol. 16, no. 7, pp. 783–795, 2008.
- [6] C.-C. Wong, H.-Y. Wang, and S.-A. Li, "PSO-based motion fuzzy controller design for mobile robots," *International Journal of Fuzzy Systems*, vol. 10, no. 1, pp. 284–292, 2008.
- [7] Y. Liu and B. Niu, "A novel PSO model based on simulating human social communication behavior," *Discrete Dynamics in Nature and Society*, vol. 2012, Article ID 791373, 21 pages, 2012.
- [8] W. Zou, Y. Zhu, H. Chen, and X. Sui, "A clustering approach using cooperative artificial bee colony algorithm," *Discrete Dynamics in Nature and Society*, vol. 2010, Article ID 459796, 16 pages, 2010.
- [9] P. Umopathy, C. Venkataseshiaiah, and M. S. Arumugam, "Particle swarm optimization with various inertia weight variants for optimal power flow solution," *Discrete Dynamics in Nature and Society*, vol. 2010, Article ID 462145, 15 pages, 2010.
- [10] N. Kazakova, M. Margala, and N. G. Durdle, "Sobel edge detection processor for a real-time volume rendering system," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 2, pp. II913–II916, May 2004.
- [11] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley & Sons, 2001.
- [12] C.-Y. Lee, J.-J. Leou, and H.-H. Hsiao, "Saliency-directed color image segmentation using modified particle swarm optimization," *Signal Processing*, vol. 92, no. 1, pp. 1–18, 2012.
- [13] H.-Y. Chen and J.-J. Leou, "Saliency-directed image interpolation using particle swarm optimization," *Signal Processing*, vol. 90, no. 5, pp. 1676–1692, 2010.
- [14] R. Eberhart and J. Kennedy, "New optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, pp. 39–43, October 1995.
- [15] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.
- [16] D. Srinivasan, W. H. Loo, and R. L. Cheu, "Traffic incident detection using particle swarm optimization," in *Proceedings of IEEE Swarm Intelligence Symposium*, pp. 144–151, 2003.

Research Article

Binary Structuring Elements Decomposition Based on an Improved Recursive Dilation-Union Model and RSAPSO Method

Yudong Zhang,¹ Shuihua Wang,^{1,2} Yi Sun,³ Genlin Ji,¹
Preetha Phillips,⁴ and Zhengchao Dong⁵

¹ School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China

² School of Electronic Science and Engineering, Nanjing University, Nanjing, Jiangsu 210046, China

³ State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

⁴ School of Natural Sciences and Mathematics, Shepherd University, Shepherdstown, WV 25443, USA

⁵ Translational Imaging Division and MRI Unit, Columbia University and New York State Psychiatric Institute, New York, NY 10032, USA

Correspondence should be addressed to Yudong Zhang; zhangyudongnuaa@gmail.com

Received 2 January 2014; Accepted 20 January 2014; Published 27 April 2014

Academic Editor: Baozhen Yao

Copyright © 2014 Yudong Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposed an improved approach to decompose structuring elements of an arbitrary shape. For the model of this method, we use an improved dilation-union model, adding a new termination criterion, as the sum of 3-by-3 matrix should be less than 5. Next for the algorithm of this method, we introduced in the restarted simulated annealing particle swarm optimization method. The experiments demonstrate that our method can find better results than Park's method, Anelli's method, Shih's SGA method, and Zhang's MFSGA method. Besides, our method gave the best decomposition tree of different SE shapes including "ship," "car," "heart," "umbrella," "vase," "tree," "cat," "V," "bomb," and "cup."

1. Introduction

Mathematical morphology (MM) is a theory and technique for the analysis and processing of geometrical structures based on set theory, lattice theory, topology, and random functions. MM not only is commonly applied to digital images, but also can be employed to graphs, surface meshes, solids, and many other spatial structures [1].

In these applications, the inherent strategy in MM is to explore the characteristics of an object by probing its microstructure with various forms, known as structuring element (SE). Most image processing architectures adapted to morphological operations use SEs of a limited size. However, implementation becomes difficult when a large-sized SE is employed. Hence, the techniques for decomposing a large-sized SE into combined small-sized SEs are of importance [2].

In the last decade, techniques have been proposed for the decomposition of SE, but they are intricate and

sometimes suffer from indecomposable cases [3]. Hashimoto and Barrera indicated that traditional algorithms have the disadvantage of being unable to decompose many simply connected decomposable SEs [4]. Shih and Wu developed a method for decomposing arbitrarily shaped binary SEs by standard genetic algorithm (SGA). The algorithm performed an iterative process to create new ones that minimize a predefined fitness function [2]. Zhang and Wu proposed a recursive model and used the migration fitness scaling genetic algorithm (MFSGA) for SE decomposition [5].

The abovementioned SGA and MFSGA algorithms are time-consuming, and they can be easily trapped into local optimal points, leading to a wrong solution. Particle swarm optimization (PSO) is an efficient algorithm compared to SGA [6–8]. Zhang and Wu proposed a restarted simulated annealing PSO (RSAPSO) algorithm [9] for further improvement. In their paper, they reported that (1) RSAPSO combined the global search ability of PSO and the local search

ability of restarted simulated annealing (RSA) algorithm, and (2) RSAPSO offset the weakness of both PSO and RSA. They also proved that the RSAPSO is superior to SGA, RSA, and PSO by six benchmark functions. In this paper, we proposed to introduce the recursive model and the RSAPSO algorithm for SE decomposition.

The rest of the paper is organized as follows. Section 2 described the methodology, including the concept of SE decomposition, the recursive dilation-union model, the encoding strategy, and the objective function. We also introduced the RSAPSO algorithm. Experiments in Section 3 compared the RSAPSO algorithm with Park's method, Shi's method, Anelli's SGA method, and Zhang's MFSGA method. Finally, Section 4 is devoted to discussions and conclusions.

2. Methodology

2.1. Concept of SE Decomposition. Suppose A denotes a binary image and S denotes a SE. If we decompose S into $S_1 \oplus S_2 \oplus \dots \oplus S_k$, the dilation and erosion will become as follows according to associative law:

$$A \oplus S = A \oplus (S_1 \oplus S_2 \oplus \dots \oplus S_k) = (((A \oplus S_1) \oplus S_2) \dots \oplus S_k),$$

$$A \ominus S = A \ominus (S_1 \oplus S_2 \oplus \dots \oplus S_k) = (((A \ominus S_1) \ominus S_2) \dots \ominus S_k). \quad (1)$$

The computational cost (CC) for dilation/erosion operators is equal to the number of nonzero element of S . Usually the CC of S is extremely large; however, the sum of CC of S_1, S_2, \dots and S_k are relatively small. Therefore, the SE decomposition can dramatically reduce the CC.

2.1.1. Dilation Model. Dilation model is to decompose SE by only dilation operator. Suppose S is decomposed into $S_1 \oplus S_2 \oplus \dots \oplus S_k$; the serial computational cost (SCC) of dilation model is equal to the sum of CC of S_1, S_2 , and S_k .

Consider

$$\min \quad \text{SCC} = \text{CC}(S_1) + \text{CC}(S_2) + \dots + \text{CC}(S_k) \quad (2)$$

s.t. $S = S_1 \oplus S_2 \oplus \dots \oplus S_k$.

Figure 1 shows three examples. The first decomposes a square SE of size 3×3 into a row vector of size 1×3 and a column vector of size 3×1 . The SCC decreases from 9 to $3 + 3 = 6$. The second decomposes a square SE of size 5×5 into two row vectors of size 1×3 and two column vectors of size 3×1 , and the SCC decreases from 25 to $3 + 2 + 3 + 2 = 10$. The third example decomposes a diamond SE of diameter 7 into three small SEs of size 3×3 , and the SCC decrease from 25 to $5 + 4 + 4 = 13$.

2.1.2. Dilation-Union Model. Dilation-union model is to decompose SE by both dilation and union operators. There are two types of computational cost for dilation-union model: the SCC and the parallel computational cost (PCC).

Consider

$$\text{PCC} = \max(\text{CC}(S_1) + \text{CC}(S_2), \text{CC}(S_3)) \quad (3)$$

s.t. $S = S_1 \oplus S_2 \cup S_3$.

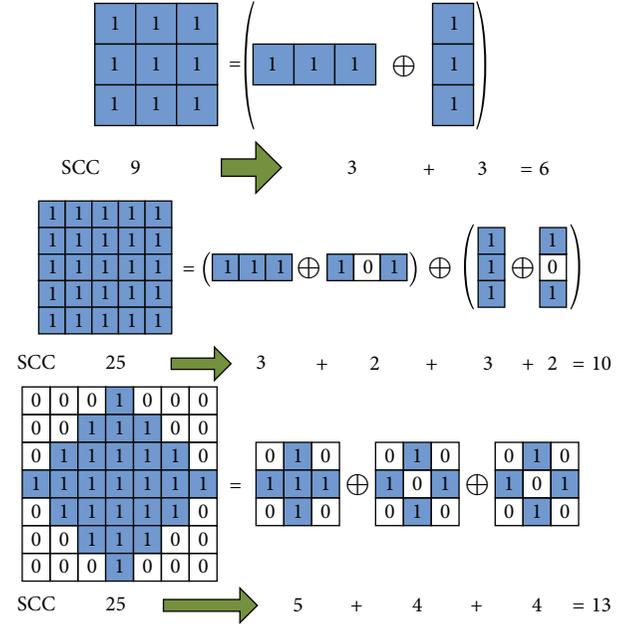


FIGURE 1: Examples of SE decomposition based on dilation model: (a) 3×3 square; (b) 5×5 square; (c) 7×7 diamond.

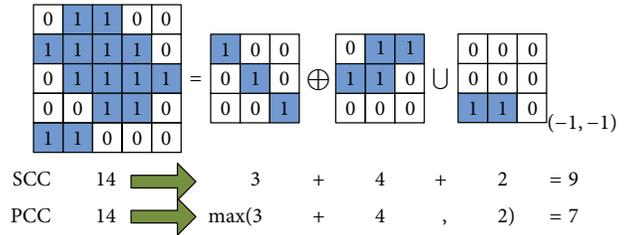


FIGURE 2: An example of SE decomposition based on dilation-union model.

Suppose S can be decomposed into the union of $S_1 \oplus S_2$ and S_3 ; then the SCC is equal to the sum of CC of S_1, S_2 , and S_3 ; the PCC is equal to the maximal value of $\text{CC}(S_1) + \text{CC}(S_2)$ and $\text{CC}(S_3)$. Figure 2 gives an example; here the subscript $(-1, -1)$ denotes that this square matrix should be translated -1 at both the x -axis and the y -axis. The SCC decreases from 14 to $3 + 4 + 2 = 9$. The PCC decreases from 14 to $\max(3 + 4, 2) = 7$.

2.1.3. Recursive Dilation-Union Model. Let S_{NN} denote a SE of size $N \times N$. The first level decomposition is written as

$$S_{NN} = V_{(N-2)(N-2)} \oplus F_{33} \cup R_{NN}. \quad (4)$$

Here we use the dilation-union model: V denotes the variable-size matrix, F denotes the fixed-size prime component, and R denotes the residue component. The R can be easily decomposed into union of factors of size 3×3 because the size 3×3 is the golden standard as the elementary structuring component for decomposition in the literature.

Consider

$$S_{NN} = V_{(N-2)(N-2)} \oplus F_{33}^{\text{PC}} \cup \left[(R_{33}^1)_{(x_1, y_1)} \cup (R_{33}^2)_{(x_2, y_2)} \cup \dots \cup (R_{33}^n)_{(x_n, y_n)} \right]. \quad (5)$$

Here n denotes the number of 3-by-3 residual matrix. The subscript (x_i, y_i) denotes that the R^i should be translated by (x_i, y_i) . Therefore, the iteration continues until the termination criteria are satisfied. In Zhang and Wu's paper [5], they defined the termination criteria (TC) as that the size of S is smaller than or equal to 3×3 :

$$\text{TC1: size}(S) \leq 3. \quad (6)$$

However, this is not perfect because the 3×3 binary SE can be decomposed further as shown in Figure 1(a). So our new criterion is set as

$$\text{TC2: size}(S) \leq 3, \quad \text{sum}(S) \leq 5. \quad (7)$$

We improved the TC by adding the rule $\text{sum}(S) \leq 5$. The reason is that any 3-by-3 SE with less than five 1s is indecomposable.

The flowchart of the recursive dilation-union model is depicted in Figure 3, which indicates that we record 4 variables ($V_k, F_k, R_k,$ and P_k) at each iteration level k , and the decomposition tree can be depicted via those variables. The following task is to develop an effective method to perform formulas (4) and (5).

2.2. Optimization Problem. The SE decomposition problem can be transformed into an optimization problem with the help of dilation-union model. In what follows we will briefly discuss the encoding strategy and the objective function of the optimization problem.

2.2.1. Encoding Strategy. Formula (4) or formula (5) can be regarded as an optimization problem. We transform the prime component F_k into a row vector string of chromosomes. For any SE at any iteration, the F_k is a 3-by-3 two-value image; therefore, the chromosome can be written as

$$\xi = \text{vector}(F) = f_1 f_2 \dots f_9. \quad (8)$$

Here ξ denotes the chromosome and f_i denotes the locus. Figure 4(a) illustrates the numbering scheme for ξ , namely, from left to right and then from top to bottom. Two examples are shown in Figures 4(b)-4(c), and their ID strings of chromosomes are "001001101" and "101110101," respectively.

2.2.2. Objective Function. Since the prime component F is encoded, variable matrix of SEV and residual matrix R can be obtained through the following formula:

$$\begin{aligned} V &= S \odot F, \\ R &= S - V \oplus F. \end{aligned} \quad (9)$$

Then, we can extract two different types of costs, SCC and PCC, as follows:

$$\begin{aligned} \text{SCC} &= \sum V + \sum F + \sum R, \\ \text{PCC} &= \max(\sum V + \sum F, \sum R). \end{aligned} \quad (10)$$

Note that those variables satisfy the equality as $S = V \oplus F \cup R$. In this paper we choose SCC as the objective function, since the PCC needs pipelined architecture, which is difficult to implement in practice.

2.3. RSAPSO Algorithm. Now that the encoding strategy and objective function are established, we employed the RSAPSO algorithm [9] to search the optimal points. PSO is a method that optimizes a problem by iteratively improving a candidate solution with regard to a given measure of quality [10]. It is commonly known as metaheuristic method as it makes few or no assumptions about the problem being optimized and can search very large spaces of candidate solutions. However, PSO does not use the gradient of the problem being optimized, which means that PSO does not require the optimization problem to be differentiable as is required by classic optimization methods such as gradient descent and quasi-Newton methods [11]. PSO can therefore also be used in optimization problems that are partially irregular, noisy, adaptive, and so forth [12].

Simulated annealing (SA) was chosen as the local search method. SA comes from annealing in metallurgy [13], a technique involving heating and controlled cooling of a material to increase the size of its crystals and reduce their defects [14]. The heat causes the atoms to become unstuck from their initial positions (a local minimum of the internal energy) and wander randomly through states of higher energy; the slow cooling gives them more chances of finding configurations with lower internal energy than the initial one [15]. We introduced the restarted simulated annealing (RSA) technique to improve the performance of SA.

The proposed algorithm RSAPSO combines both the exploitation ability from RSA and the exploration ability from PSO. It divides the population into two halves: one half runs PSO and the other half runs RSA. In what follows, we will describe those three algorithms in depth.

2.3.1. Particle Swarm Optimization. PSO is a population based stochastic optimization technique, which simulates the social behavior of a swarm of bees, a flock of birds, or a school of fish. By randomly initializing the algorithm with candidate solutions, the PSO successfully leads to a global optimum [16]. This is achieved by an iterative procedure based on the processes of movement and intelligence in an evolutionary system [17]. Figure 5(a) shows the flow chart of a PSO algorithm.

In PSO, each potential solution is represented as a particle. Two properties (position x and velocity v) are associated

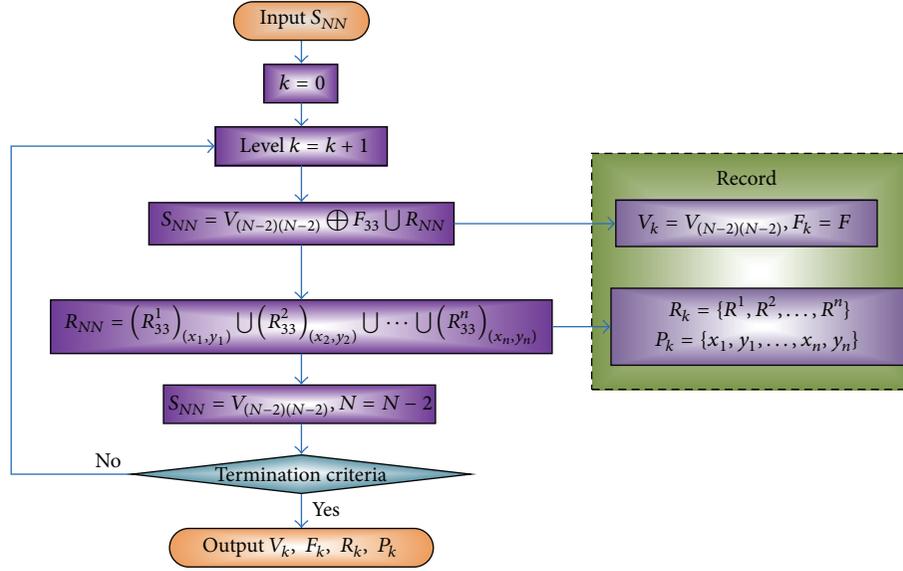


FIGURE 3: Flowchart of recursive dilation-union model.

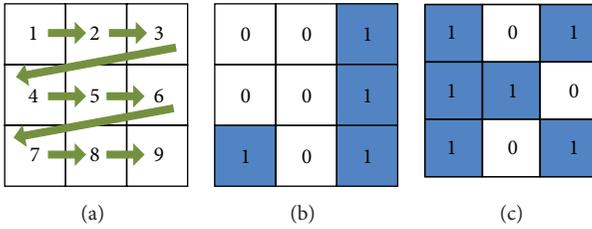


FIGURE 4: (a) Encoding order: left to right and then top to bottom; (b) an example of "001001101"; (c) an example of "101110101".

with each particle. Suppose that x and v of the i th particle are given as

$$\begin{aligned} x &= (x_{i1}, x_{i2}, \dots, x_{iN}), \\ v &= (v_{i1}, v_{i2}, \dots, v_{iN}), \end{aligned} \quad (11)$$

where N stands for the dimensions of the problem. In each iteration, a fitness function is evaluated for all the particles in the swarm [18]. The velocity of each particle is updated by keeping track of the two best positions: one is the best position a particle has traversed so far and is called " p Best" and the other is the best position that any neighbor of a particle has traversed so far. It is a neighborhood best called " n Best." When a particle takes the whole population as its neighborhood, the neighborhood best becomes the global best and is accordingly called " g Best." Hence, a particle's velocity and position are updated as follows:

$$\begin{aligned} v &= \omega \cdot v + c_1 r_1 (p\text{Best} - x) + c_2 r_2 (n\text{Best} - x), \\ x &= x + v \Delta t, \end{aligned} \quad (12)$$

where ω is called the "*inertia weight*" that controls the impact of the previous velocity of the particle on its current one.

The parameters c_1 and c_2 are positive constants, called "*acceleration coefficients*." The parameters r_1 and r_2 are random numbers that are uniformly distributed in the interval $[0, 1]$. These random numbers are updated every time when they occur. The parameter Δt stands for the given time step. The population of particles is then moved according to (12) and tends to cluster together from different directions. However, a maximum velocity v_{\max} should not be exceeded by any particle to keep the search within a meaningful solution space [19]. The PSO algorithm runs through these processes iteratively until the termination criterion is satisfied [20–22].

2.3.2. Restarted Simulated Annealing. SA algorithm is a probabilistic hill-climbing technique that is based on the annealing/cooling process of metals [23]. This annealing process occurs after the heat source is removed from a molten metal and its temperature starts to decrease. As the temperature decreases, the energy of the metal molecules reduces, and the metal becomes more rigid. The procedure continues until the metal temperature has reached the surrounding ambient temperature, at which stage the energy has reached its lowest value and the metal is perfectly solid [24].

The SA procedure begins by generating an initial solution at random. At initial stages, a small random change is made in the current solution X_c . The new solution is called X_n . The perturbation depends on a temperate parameter T and a scaling constant k .

Consider

$$\text{pert}(T) = k \times T \times r_3. \quad (13)$$

Here r_3 is a random value between 0 and 1 with uniform distribution. The temperature T decreases with each iteration of the algorithm, thus reducing the size of the perturbations as the search progresses. This mechanism produces large perturbation in the initial stages of the search and ensures that

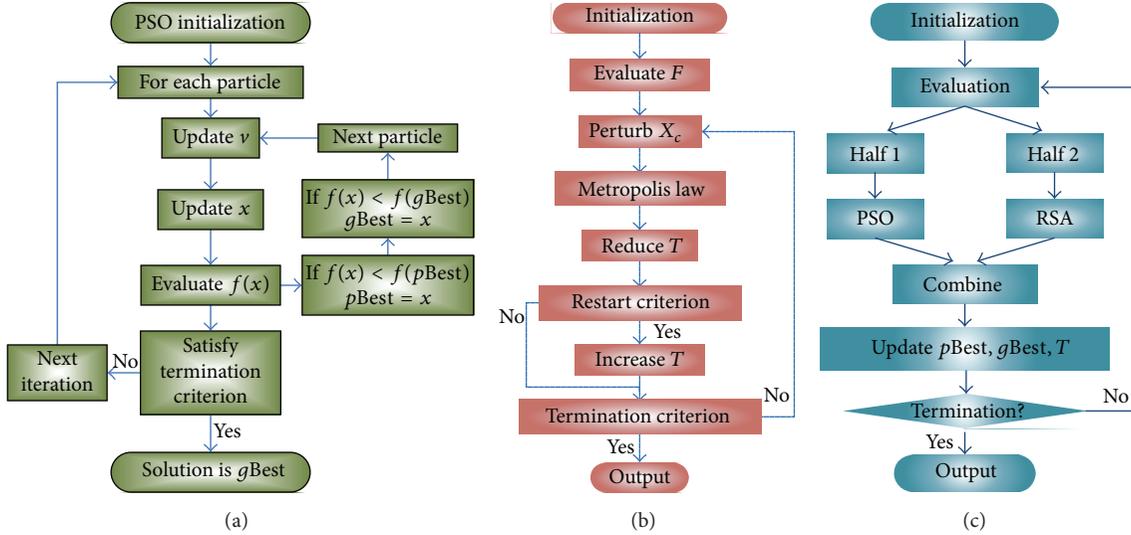


FIGURE 5: Flowchart of the (a) PSO; (b) RSA; and (c) RSAPSO.

the resulting parameters are fine-tuned towards the end of the optimization [25].

A move is made to the new solution X_n if it has smaller energy F or if the probability function has a higher value than a randomly generated number. Otherwise a new solution is generated, evaluated, and compared again. The probability p of accepting a new solution X_n which is called “Metropolis law” is given as follows:

$$p = \begin{cases} 1, & \text{if } F(X_n) < F(X_c), \\ \exp\left(\frac{F(X_c) - F(X_n)}{T}\right), & \text{otherwise.} \end{cases} \quad (14)$$

In order to avoid getting trapped at local extrema points, the reduction rate of T should be slow enough. In this study the following method to reduce the temperature has been used:

$$T_n = T_0 \times \beta^n. \quad (15)$$

Here T_0 is the initial temperature, β is the reduction constant, and n is the number of iterations. In general, most worsening moves may be accepted at initial stages, but at the final stage only improving ones are likely to be allowed. This can help the procedure jump out of a local minimum.

However, sometimes it is better to move back to a former solution that was significantly better rather than always moving from the current state. This process is called “restarting” of SA [26]. To do this we set the temperature to a former value and restart the annealing schedule. The decision to restart can be based on several criteria, including whether a fixed number of steps had passed, whether the current energy is too high compared with the best obtained so far, or whether the random number falls within prescribed range (randomly restart). In this paper, we restart the SA when the current energy is too high compared with the best energy because it performs best among all criteria [27]. The flowchart of RSA is shown in Figure 5(b).

2.3.3. Pseudocodes of RSAPSO. Traditional PSO algorithm suffers from getting trapped at the early stage [28]. On the other hand, RSA accepts a worse solution so it can escape from a local minimum, resist premature convergence, and increase the diversity [12]. Therefore, a new hybrid strategy was proposed and referred to as RSAPSO [9]. The proposed RSAPSO algorithm offsets the weaknesses of PSO and RSA. The main idea lies in the fact that it divides the population into two halves: one half runs PSO and the other half runs RSA. During each step, the results are combined and updated by the best result that is picked up from the whole population. The flowchart of RSAPSO is depicted in Figure 5(c), and its pseudocodes are given as follows.

Step 1 (initialization). Generate the population randomly.

Step 2 (evaluation). Evaluate each particle’s objective function f .

Step 3 (segmentation). Halve the population randomly: one half was updated by PSO according to formula (12) and the other half was updated by RSA according to formulas (13) and (14).

Step 4 (update). Update $pBest$ and $gBest$; update T according to formula (15).

Step 5 (repeat). Repeat Step 2 to Step 4 until the termination criterion is satisfied.

Step 6 (output). Output the final results.

3. Experiments and Discussions

The experiments were carried out on the platform of P4 IBM with 2.2 GHz Intel Core i3-2330 M CPU and 6 GB RAM, running under 64-bit Windows 7 operating system. The algorithm was in-house developed via the global optimization

TABLE 1: Parameters setting.

Algorithm	Parameters
SGA	$N = 20, P_C = 0.8, P_M = 0.1$
MFSGA	$N = 20, P_C = 0.8, P_M = 0.1, M_I = 10, M_R = 2$
RSAPSO	$N = 20, \omega = 0.6, c_1 = 1, c_2 = 1, T_0 = 100, \beta = 0.95$

toolbox of MATLAB 2013a. Readers can repeat the results of the experiment on any desktop installing MATLAB.

3.1. Parameters Setting. We compared the proposed method RSAPSO with Park's method, Anelli's method, Shih's SGA method, and Zhang's MFSGA method. Some important parameters are obtained through trial-and-error method and listed in Table 1. Here, N denotes the number of population, P_C denotes the crossover probability, P_M denotes the mutation probability, M_I denotes the migration interval, and M_R denotes the migration rate.

3.2. Comparison with Park's Method. The SE in Figure 6 is indecomposable by Park's algorithm [4]. Its original SCC is 21. We ran our RSAPSO method 20 times, and all runs obtained the optimal decomposition shown in Figure 6(a) with SCC as 10. Besides, we ran SGA and MFSGA 20 times for comparison. Their results were shown in Table 2. The second row shows the result of all 20 runs. Take SGA as an example; "19(10) + 1(11)" represents that 19 runs obtained SCC as 10 and the left one run obtained SCC as 11. The third row "Averaged SCC" shows either the averaged SCC result of heuristic methods or the SCC result of deterministic methods. Table 2 shows that SGA obtained optimal result (10) 19 times and one suboptimal result (11) as shown in Figure 6(b). MFSGA obtained optimal result (10) 19 times and one suboptimal result (11) as shown in Figure 6(c). The SCC of the two suboptimal SE decomposition results was 11. The averaged SCC of SGA, MFSGA, and RSAPSO were 10.05, 10.05, and 10, respectively.

3.3. Comparison with Anelli's Method. We compared the proposed RSAPSO with Anelli's method [29], SGA, and MFSGA. We ran SGA, MFSGA, and RSAPSO 20 times, since the distribution of their initial population is random. Figures 7(a) and 7(b) show two successful decomposition trees of Anelli's SE, and both of their SCC are 18. Figures 7(c) and 7(d) show two failed decomposition trees, of which their SCC are 22 and 23, respectively.

The comparison results are shown in Table 3. The SCC of original SE was 41. Anelli's method reduced it to 22, and the corresponding decomposition tree can be found in [29]. Among 20 runs, SGA obtained 15 times of optimal results, MFSGA obtained 17 times of optimal results, while the proposed RSAPSO method obtained all 20 successful runs. The averaged SCC of SGA, MFSGA, and RSAPSO were 19.05, 18.6, and 18, respectively.

3.4. Comparison with SGA. Shih's paper proclaimed that the optimal SCC results of decomposed trees for "ship" and "car"

shapes by SGA were 47 and 46, respectively. After searching by RSAPSO, we obtained better results (see Table 5). For the "ship" shape, the original SCC was 125, the optimal SCC found by SGA was 47, and our method found better SCC result as 44 (see 2nd row in Table 4). For the "car" shape, the original SCC was 168, the optimal SCC by SGA was 46, and our method achieved better SCC as 43 (see the 3rd row in Table 4).

3.5. Comparison with MFSGA. In what follows, we compared the proposed RSAPSO with MFSGA method. Zhang et al. proclaimed that the optimal SCC for "heart" and "umbrella" by MFSGA were 32 and 44, respectively. Using RSAPSO, we obtained better results. For SE of "heart," the original SCC was 142. MFSGA minimized SCC to 32. Our method obtained better SCC result as 30. For SE of "umbrella," the original SCC was 94. MFSGA minimized SCC to 44. Our method obtained better result as 41.

3.6. Other Examples. We used the RSAPSO for six other benchmarks SEs of different shapes, such as vase, tree, cat, V, bomb, and cup. The results were shown in Table 6. For the "vase" shape, the SCC of the original and decomposed SE was 149 and 32, respectively. For the "tree" shape, the SCC of the original and decomposed SE was 73 and 26, respectively. For the "cat" shape, the SCC of the original and decomposed SE was 102 and 30, respectively. For letter "V," the SCC of the original and decomposed SE was 94 and 34, respectively. For the "bomb" shape, the SCC of the original and decomposed SE was 91 and 26, respectively. For the "cup" shape, the SCC of the original and decomposed SE was 113 and 25, respectively.

4. Conclusions

In this paper, a novel decomposition method for arbitrarily shaped SE was proposed. The SE decomposition problem was first transformed into an optimization problem with virtue of the improved recursive dilation-union model that contains a new termination criterion. Afterwards, the RSAPSO method was introduced as the searching algorithm. In the experiments, we compared our method with Park's method [4], Anelli's method [29], Shih's SGA method [2], and Zhang's MFSGA method [5]. The results based on several benchmark shapes showed that our method was more robust than the aforementioned algorithms and was able to find the optimal decomposition tree.

The contribution of the paper lies in the following 4 aspects: (1) we proposed improved termination criteria for the recursive dilation-union model, adding the rule "sum(S) \leq 5"; (2) we used the serial computational time as the objective function; (3) we introduced the RSAPSO algorithm and proved it is superior to Park's method, Anelli's method, SGA, and MFSGA for SE decomposition application; and (4) we gave the best decomposition results for the shapes "heart," "ship," and "car" among all the state-of-the-art SE decomposition methods.

The future tentative research will focus on other optimization algorithms, such as cuckoo search [30], harmony

TABLE 2: Decomposition of Park's SE (20 runs).

	Original	Park's	SGA	MFSGA	RSAPSO
SCC			19 (10) + 1 (11)	19 (10) + 1 (11)	20 (10)
Averaged SCC	21	21	10.05	10.05	10

TABLE 3: Decomposition of Anelli's SE (20 runs).

	Original	Anelli's	SGA	MFSGA	RSAPSO
SCC			15 (18) + 4 (22) + 1 (23)	17 (18) + 3 (22)	20 (18)
Averaged SCC	41	22	19.05	18.6	18

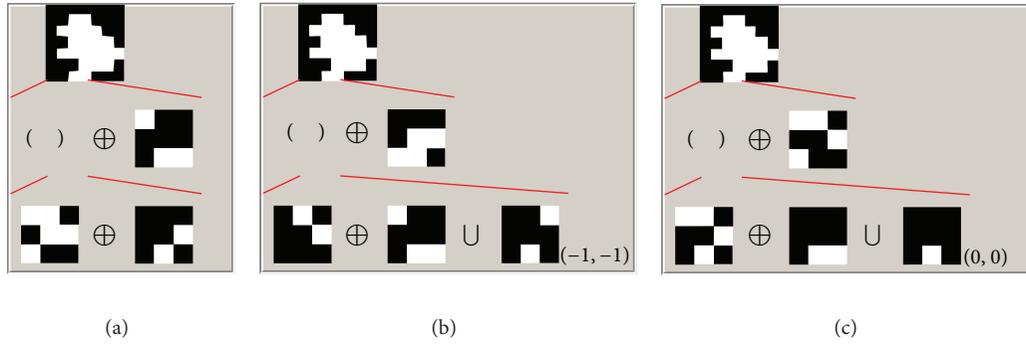


FIGURE 6: Decomposition tree of Park's SE (SCC = 21): (a) a successful run (SCC = 10); (b) a failed run by SGA (SCC = 11); and (c) a failed run by MFSGA (SCC = 11).

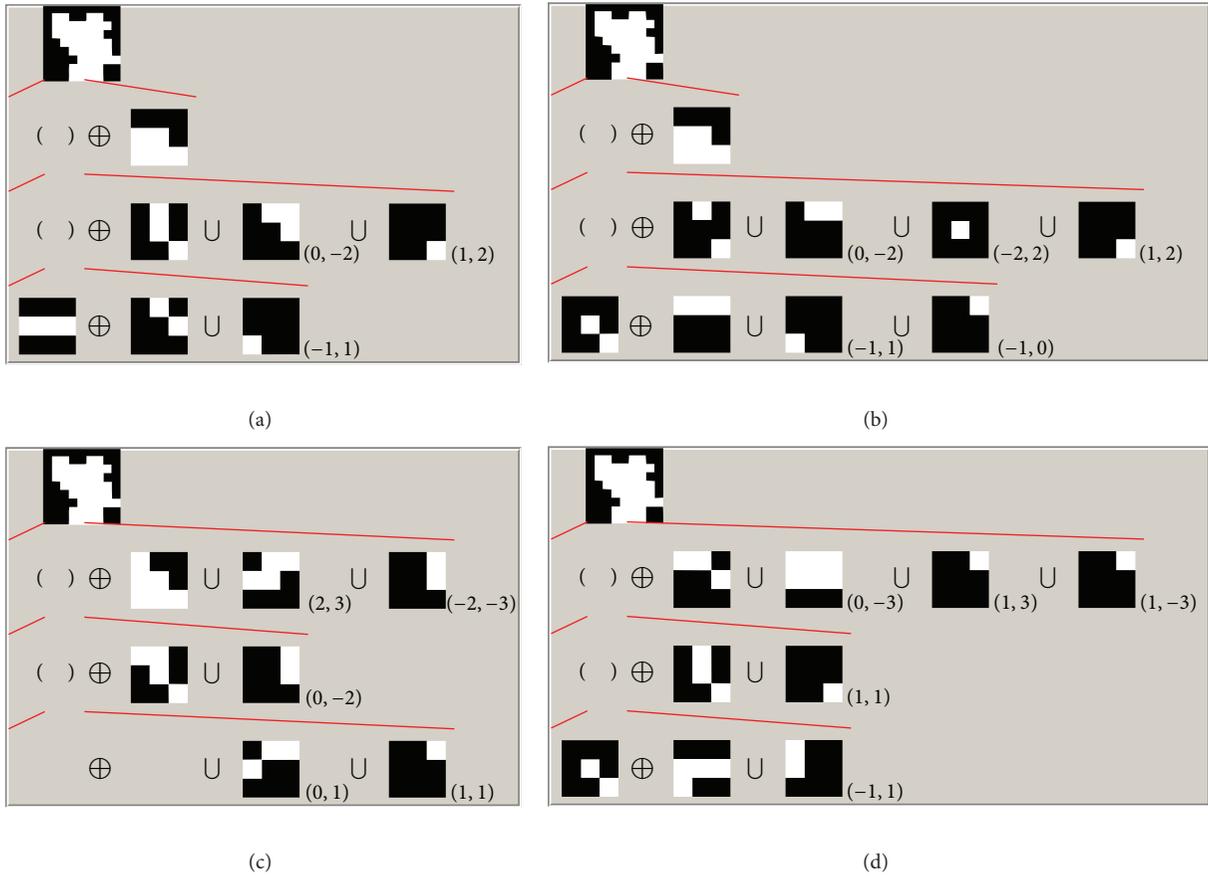
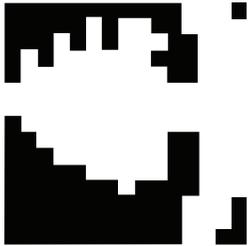
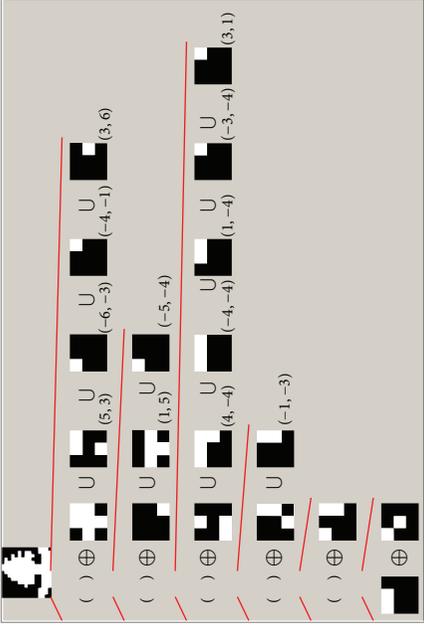
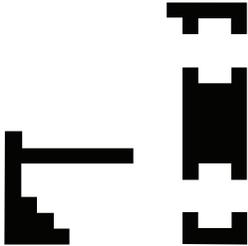
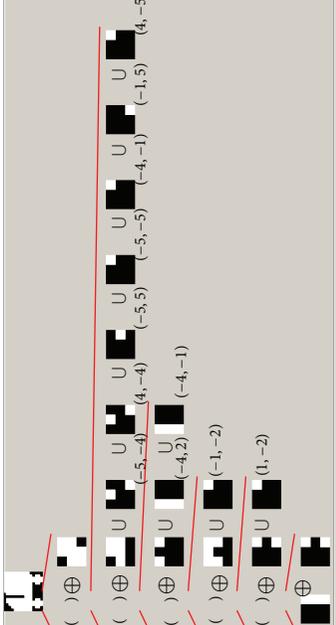


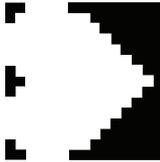
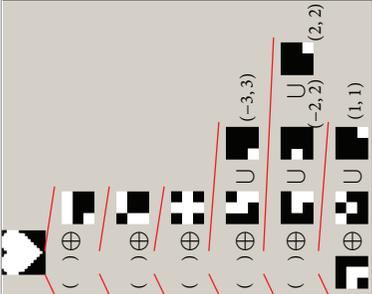
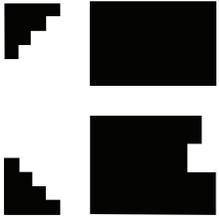
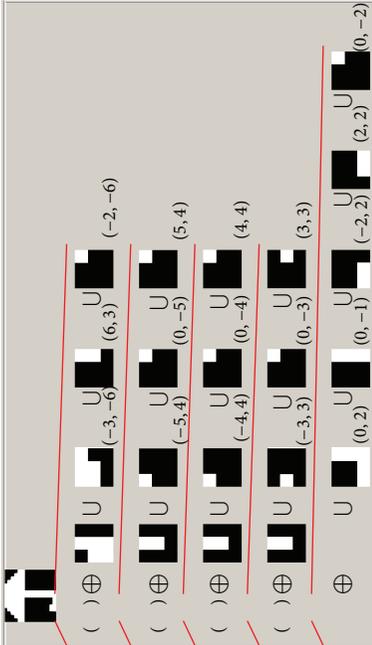
FIGURE 7: Decomposition tree of Anelli's SE (SCC = 41): (a) a successful run (SCC = 18); (b) another successful run (time = 18); (c) a failed run (SCC = 22); and (d) a failed run (SCC = 23).

TABLE 4: Two-SE decomposition by RSAPSO.

Name	SE	Decomposition by RSAPSO	Original	SCC SGA	RSAPSO
Ship			125	47*	44
Car			168	46*	43

*Data from Shih and Wu [2].

TABLE 5: Two-SE decomposition by RSAPSO.

Name	SE	Decomposition by RSAPSO	Original	SCC MFSGA	RSAPSO
Heart			142	32*	30
Umbrella			94	44*	41

*Data from Zhang and Wu [5].

TABLE 6: Six-SE decomposition examples by our RSAPSO method.

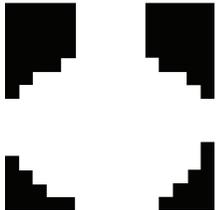
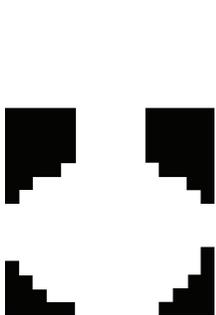
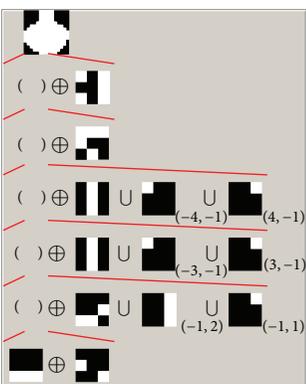
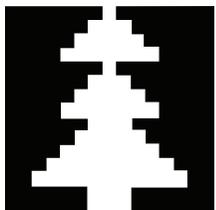
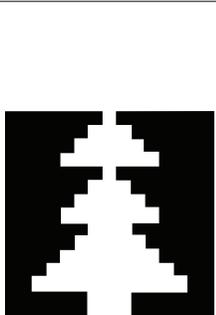
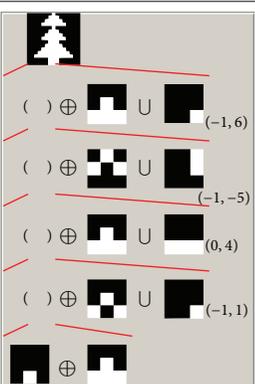
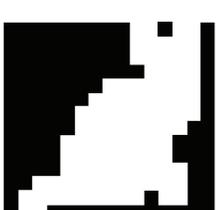
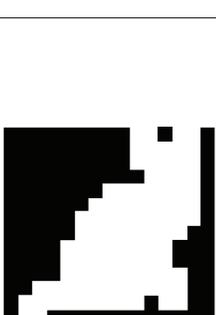
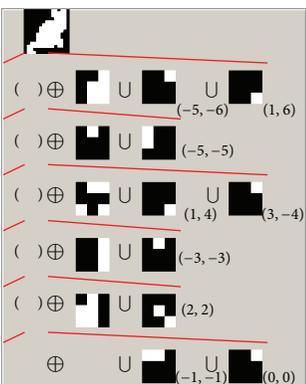
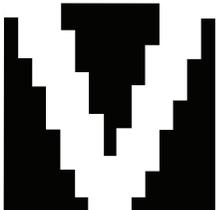
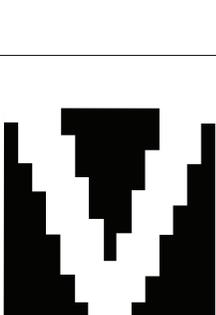
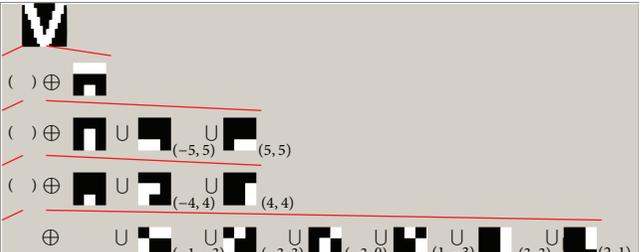
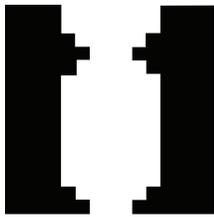
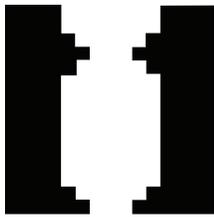
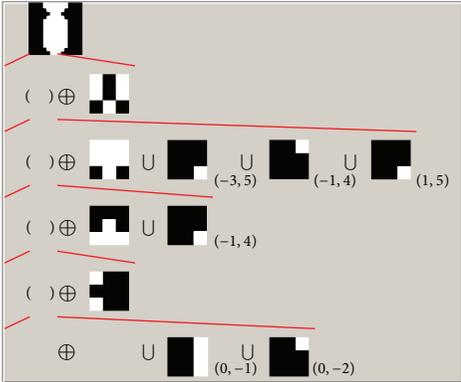
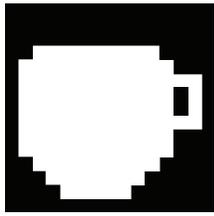
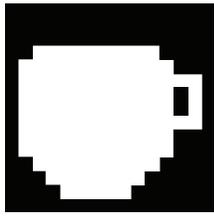
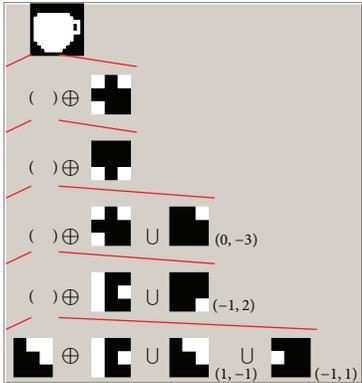
Name	SE	Original	SCC	Tree	Decomposition	SCC
Vase			149			32
Tree			73			26
Cat			102			30
V			94			34

TABLE 6: Continued.

Name	SE	Original	SCC	Tree	Decomposition	SCC
Bomb			91			26
Cup			113			25

search [31], genetic pattern research [32], Tabu search [33], firefly algorithm [34], honey bee mating [35], and artificial bee colony [36, 37]. We will also try other termination criteria [38] and check the effectiveness of the new SE decomposition methods.

Conflict of Interests

The authors declare that they do not have any commercial or associative interest that represents a conflict of interests in connection with the work submitted.

Acknowledgments

The authors would like to express their gratitude to the three anonymous reviewers. This paper is supported from the Nanjing Normal University Research Foundation for Talented Scholars (no. 2013119XGQ0061).

References

[1] O. Cuisenaire, “Locally adaptable mathematical morphology using distance transformations,” *Pattern Recognition*, vol. 39, no. 3, pp. 405–416, 2006.
 [2] F. Y. Shih and Y.-T. Wu, “Decomposition of binary morphological structuring elements based on genetic algorithms,”

Computer Vision and Image Understanding, vol. 99, no. 2, pp. 291–302, 2005.
 [3] H. Park and J. Yoo, “Structuring element decomposition for efficient implementation of morphological filters,” *IEE Proceedings: Vision, Image and Signal Processing*, vol. 148, no. 1, pp. 31–35, 2001.
 [4] R. F. Hashimoto and J. Barrera, “A note on Park and Chin’s algorithm,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 139–144, 2002.
 [5] Y. Zhang and L. Wu, “Recursive structure element decomposition using migration fitness scaling genetic algorithm,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6728, no. 1, pp. 514–521, 2011.
 [6] G. Fornarelli and A. Giaquinto, “Adaptive particle swarm optimization for CNN associative memories design,” *Neurocomputing*, vol. 72, no. 16–18, pp. 3851–3862, 2009.
 [7] B. Alatas and E. Akin, “Chaotically encoded particle swarm optimization algorithm and its applications,” *Chaos, Solitons and Fractals*, vol. 41, no. 2, pp. 939–950, 2009.
 [8] M. Rabbani, M. A. Bajestani, and G. Baharian Khoshkhou, “A multi-objective particle swarm optimization for project selection problem,” *Expert Systems with Applications*, vol. 37, no. 1, pp. 315–321, 2010.
 [9] Y. Zhang and L. Wu, “A robust hybrid restarted simulated annealing particle swarm optimization technique,” *Advances in Computer Science and Its Applications*, vol. 1, no. 1, pp. 5–8, 2012.

- [10] M. S. Arumugam, M. V. C. Rao, and A. W. C. Tan, "A novel and effective particle swarm optimization like algorithm with extrapolation technique," *Applied Soft Computing Journal*, vol. 9, no. 1, pp. 308–320, 2009.
- [11] A. Y. Abdelaziz, F. M. Mohammed, S. F. Mekhamer, and M. A. L. Badr, "Distribution Systems Reconfiguration using a modified particle swarm optimization algorithm," *Electric Power Systems Research*, vol. 79, no. 11, pp. 1521–1530, 2009.
- [12] Y. Zhang, Y. Jun, G. Wei, and L. Wu, "Find multi-objective paths in stochastic networks via chaotic immune PSO," *Expert Systems with Applications*, vol. 37, no. 3, pp. 1911–1919, 2010.
- [13] A. Nakib, H. Oulhadj, and P. Siarry, "Image histogram thresholding based on multiobjective optimization," *Signal Processing*, vol. 87, no. 11, pp. 2516–2534, 2007.
- [14] H. Mohamadi, J. Habibi, M. S. Abadeh, and H. Saadi, "Data mining with a simulated annealing based fuzzy classification system," *Pattern Recognition*, vol. 41, no. 5, pp. 1841–1850, 2008.
- [15] S. Rubenthaler, T. Rydén, and M. Wiktorsson, "Fast simulated annealing in Rd with an application to maximum likelihood estimation in state-space models," *Stochastic Processes and their Applications*, vol. 119, no. 6, pp. 1912–1931, 2009.
- [16] M. Dorigo, M. Birattari, and T. Stützle, "Ant colony optimization artificial ants as a computational intelligence technique," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2006.
- [17] Y. Zhang, P. Agarwal, V. Bhatnagar et al., *Swarm Intelligence and Its Applications*, The Scientific World Journal, 2013.
- [18] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization," *Mathematical Problems in Engineering*, vol. 2013, Article ID 705238, 9 pages, 2013.
- [19] L. Lamberti, "An efficient simulated annealing algorithm for design optimization of truss structures," *Computers and Structures*, vol. 86, no. 19–20, pp. 1936–1953, 2008.
- [20] A. Liu, Y. Yang, Q. Xing, H. Yao, Y. Zhang, and Z. Zhou, "Improved collaborative particle swarm algorithm for job shop scheduling optimization," *Advanced Science Letters*, vol. 4, no. 6–7, pp. 2180–2183, 2011.
- [21] Y. Zhang and L. Wu, "Bacterial chemotaxis optimization for protein folding model," in *Proceedings of the 5th International Conference on Natural Computation (ICNC '09)*, pp. 159–162, Tianjian, China, August 2009.
- [22] Y. Zhang, S. Wang, G. Ji et al., "An MR brain images classifier system via particle swarm optimization and kernel support vector machine," *The Scientific World Journal*, vol. 2013, Article ID 130134, 9 pages, 2013.
- [23] E. Cosola, K. Genovese, L. Lamberti, and C. Pappalettere, "A general framework for identification of hyper-elastic membranes with moiré techniques and multi-point simulated annealing," *International Journal of Solids and Structures*, vol. 45, no. 24, pp. 6074–6099, 2008.
- [24] E. W. McGookin and D. J. Murray-Smith, "Submarine manoeuvring controller's optimisation using simulated annealing and genetic algorithms," *Control Engineering Practice*, vol. 14, no. 1, pp. 1–15, 2006.
- [25] Y. Zhang, L. Wu, N. Neggaz, S. Wang, and G. Wei, "Remote-sensing image classification based on an improved probabilistic neural network," *Sensors*, vol. 9, no. 9, pp. 7516–7539, 2009.
- [26] N. Cretu and M.-I. Pop, "Acoustic behavior design with simulated annealing," *Computational Materials Science*, vol. 44, no. 4, pp. 1312–1318, 2009.
- [27] I. A. Gheyas and L. S. Smith, "Feature subset selection in large dimensionality domains," *Pattern Recognition*, vol. 43, no. 1, pp. 5–13, 2010.
- [28] T. Niknam, H. D. Mojarrad, and M. Nayeripour, "A new fuzzy adaptive particle swarm optimization for non-smooth economic dispatch," *Energy*, vol. 35, no. 4, pp. 1764–1778, 2010.
- [29] G. Anelli, A. Broggi, and G. Destri, "Decomposition of arbitrarily shaped binary morphological structuring elements using genetic algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 2, pp. 217–224, 1998.
- [30] S. Walton, O. Hassan, K. Morgan, and M. R. Brown, "Modified cuckoo search: a new gradient free optimisation algorithm," *Chaos, Solitons and Fractals*, vol. 44, no. 9, pp. 710–718, 2011.
- [31] I. Ahmad, M. G. Mohammad, A. A. Salman, and S. A. Hamdan, "Broadcast scheduling in packet radio networks using Harmony Search algorithm," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1526–1535, 2012.
- [32] Y. Zhang, S. Wang, G. Ji et al., "Genetic pattern search and its application to brain image classification," *Mathematical Problems in Engineering*, vol. 2013, Article ID 580876, 8 pages, 2013.
- [33] B. Yao, P. Hu, X. Lu, J. Gao, and M. Zhang, "Transit network design based on travel time reliability," *Transportation Research Part C: Emerging Technologies*, 2014.
- [34] Y. Zhang, L. Wu, and S. Wang, "Solving two-dimensional HP model by firefly algorithm and simplified energy function," *Mathematical Problems in Engineering*, vol. 2013, Article ID 398141, 9 pages, 2013.
- [35] M.-H. Horng, "Multilevel minimum cross entropy threshold selection based on the honey bee mating optimization," *Expert Systems with Applications*, vol. 37, no. 6, pp. 4580–4592, 2010.
- [36] Y. Zhang, L. Wu, and S. Wang, "Magnetic resonance brain image classification by an improved artificial bee colony algorithm," *Progress in Electromagnetics Research*, vol. 116, pp. 65–79, 2011.
- [37] B. Yao, B. Yao, B. Yao et al., "Artificial bee colony algorithm with scanning strategy for periodic vehicle routing problem," *SIMULATION: Transactions of the Society For Modeling and Simulation International*, vol. 89, no. 6, pp. 762–770, 2013.
- [38] J. Sun, B.-Z. Yao, C.-Y. Yang et al., "Tunnel surrounding rock displacement prediction using support vector machine," *International Journal of Computational Intelligence Systems*, vol. 3, no. 6, pp. 843–852, 2010.

Research Article

Decision of Multimodal Transportation Scheme Based on Swarm Intelligence

Kai Lei,¹ Xiaoning Zhu,¹ Jianfei Hou,² and Wencheng Huang¹

¹ School of Traffic & Transportation, Beijing Jiaotong University, Beijing 100044, China

² China Academy of Transportation Science, Beijing 100029, China

Correspondence should be addressed to Xiaoning Zhu; leikai@bjtu.edu.cn

Received 10 January 2014; Revised 20 March 2014; Accepted 20 March 2014; Published 24 April 2014

Academic Editor: Rui Mu

Copyright © 2014 Kai Lei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, some basic concepts of multimodal transportation and swarm intelligence were described and reviewed and analyzed related literatures of multimodal transportation scheme decision and swarm intelligence methods application areas. Then, this paper established a multimodal transportation scheme decision optimization mathematical model based on transportation costs, transportation time, and transportation risks, explained relevant parameters and the constraints of the model in detail, and used the weight coefficient to transform the multiobjective optimization problems into a single objective optimization transportation scheme decision problem. Then, this paper is proposed by combining particle swarm optimization algorithm and ant colony algorithm (PSACO) to solve the combinatorial optimization problem of multimodal transportation scheme decision for the first time; this algorithm effectively combines the advantages of particle swarm optimization algorithm and ant colony algorithm. The solution shows that the PSACO algorithm has two algorithms' advantages and makes up their own problems; PSACO algorithm is better than ant colony algorithm in time efficiency and its accuracy is better than that of the particle swarm optimization algorithm, which is proved to be an effective heuristic algorithm to solve the problem about multimodal transportation scheme decision, and it can provide economical, reasonable, and safe transportation plan reference for the transportation decision makers.

1. Introduction

1.1. Basic Concepts

(1) *Swarm Intelligence*. Swarm intelligence refers to the overall intelligent behavior emerging from interaction of many simple acts of individual; the swarm intelligence can produce a number of different individual collective behaviors. Swarm intelligence is the creative application and development of the collective behaviors and forms a more advanced whole wisdom. Swarm intelligence is a hot research topic in artificial intelligence researches in recent years, with no centralized control, does not provide a global model, and provides a complex distributed solution.

(2) *Multimodal Transportation*. Multimodal transportation is defined as completing the transportation process by using at least two transportation tools to connect and transport

together; it is a kind of transportation organization form which uses optimum efficiency as the goal. There are no new transportation channels and tools in the multimodal transportation process; it is a combination of modern organization means and single transportation mode. It has important and realistic meaning for saving transportation costs and transportation time and improving transportation service quality by researching the decision of multimodal transportation scheme in the process of transportation.

1.2. The Current Analysis Situation of the Domestic and Foreign Research

(1) *Swarm Intelligence*. In recent years, the scholars at home and abroad carried out a lot of research works about swarm intelligence; the swarm intelligence theory in transportation related fields has been applied extensively. Chen and Ma

[1] gave a new algorithm of glowworm swarm optimization algorithm which was based on swarm intelligence optimization principle and successfully applied it to the 0-1 knapsack problem; Li et al. [2] proposed the Harvard structure and swarm intelligence optimization system based on the CTLS model, which used swarm intelligence based on calculation and source of agent in agent technology and the simulation based optimization theory, and the feasibility and reliability of the modeling optimization method are verified by simulation analysis of CTLS half bridge scheduling problem; Mi [3] proposed an optimization algorithm for execution efficiency and convergence problems of integrated design process using genetic algorithm for task scheduling; the genetic algorithm was improved and combined with the particle swarm algorithm by considering the effect of coupling relationship between tasks to the task execution results. It is verified with an example; the results showed that the algorithm has a fast convergence speed and stable result; Li et al. [4] established automated warehouse single pock horizontally rotating storage shelf system mathematical model and used the ant colony optimization algorithm to solve the path planning problem based on the goods selection. This method is capable of rotating storage rack system for rapid selection of goods and finds the optimal goods picking path in the global, high quality solution and short calculation time; Wu and Shi [5] established ant colony algorithm based on the improved meeting algorithm and proposed ant colony algorithm ant touring quality, meeting algorithm, and a parallel algorithm which is combined with the proposed strategy; the experimental results show that this algorithm has better effectiveness; Xiao et al. [6] proposed the mathematical model of concurrent tolerance optimization design and then mapped into a special traveling salesman problem order multiple traveling salesman problem; so as to reduce the difficulty of problem solving, they proposed a hybrid swarm intelligence algorithm for concurrent tolerance optimization design problem. Through calculation examples, the efficiency would be hybrid swarm intelligence algorithm with genetic algorithm and simulated annealing algorithm has better search ability and higher. Ma et al. [7] minimized the energy consumption and delay time, improved the evolutionary algorithms for solving the traveling salesman problem based on artificial neural network algorithm, and established the swarm intelligence analysis model and used it to analyze the lines planning for public travel. The experimental results showed that, the improved hybrid intelligent computing method was simple and effective and helped to overcome the blindness of algorithm selection, to further expand the research direction of computational intelligence. From the above research literatures, it is not difficult to see that the swarm intelligence has not only attracted much attention in recent years but also has a lot of space in the field of transportation applications. This paper was proposed by combining particle swarm optimization algorithm and ant colony algorithm (PSACO) to solve the combinatorial optimization problem of multimodal transportation scheme decision for the first time which can make up shortcomings of their own at the same time.

(2) *Decision of Multimodal Transportation Scheme.* In recent years, scholars at home and abroad did a lot of research works on how to choose the proper transportation mode in the multimodal transportation access. Among them, Lozano and Storchi's [8] research was about the problem of shortest feasible path in the multimodal transportation and used the sequential algorithm to solve the problem; Calvete et al. [9] studied the VRP with soft time window vehicles path problem and used the goal programming method to solve the problem. Li and Zhao [10] used the minimum total cost as the optimization objective, which included transportation cost, transfer cost and penalty cost; the goods delivery time and transportation capacity were used as constraint conditions; they built mixed integer linear programming model and designed the integer coding genetic algorithm. Wang et al. [11] studied and established the multimodal transportation 0-1 integer programming model under the time and capacity constraints, designed the natural number coding genetic algorithm based on the characteristics of the model, and finally tested validity of the model and the algorithm with examples. Feng and Zhang [12] considered the environmental pollution, energy consumption, and safety factors into multimodal transportation sustainable development and then introduced the concept of total society cost of multimodal transportation, used this minimum cost value as a whole object, and built a multimodal transportation collaborative optimization model. Tong and Nie [13] transformed intermodal path optimization problem into a generalized shortest path problem, used time and cost as optimization objective, established multimodal transportation mathematical model for the optimization of path changes in volume, and solved and verified actual problem by using ant colony algorithm. Zhang [14] deeply studied the whole container multimodal transportation network, expounded the key elements and the operation mechanism of multimodal container transportation network in the whole system, and, in accordance with the existing problems, put forward the basic countermeasures rationalization. Li and Zeng [15] improved path shortest time model under the multimodal transportation in the fourth party logistics, gave the transportation cost model based on satisfactory time path under the condition of multimodal transportation, and improved genetic algorithm to solve the problem. Wen [16] used multimodal container transportation optimization as the research object and discussed the optimization method of container multimodal transportation routes, transportation, and time constraints modes. Li [17] discussed the development of transportation mode choice as well as the main transportation mode selection method and then put forward the concept of generalized transportation cost. Wang [18], in order to achieve time delivery, reduced the cost, according to the actual situation of the key problem of logistics distribution in the process of transportation and traffic network of multimodal transportation with time window of minimum cost problem. Zhang [19] analyzed Chinese railway container transportation features and operation process, used the transportation path selection as the goal to improve the efficiency of container transport and shorten the container transport cycle, and reduced transportation costs for the purpose; then he used the optimization theory and

operations research theory to establish the railway container transportation route choice model.

There are some shortcomings existing in the domestic and foreign documents all above about the multimodal transportation modes selection; the main problem is that the impact of transportation cost and time selection on multimodal transportation modes was considered in the past; only a single target situation for multimodal transportation mode optimal selection failed to consider other factors in the problems. But, in the actual decision process selection in multimodal transportation modes, in order to ensure the continuity and safety, we need to consider transportation safety and transportation risk factors comprehensively; only in this way we can meet the actual needs of transportation.

Multimodal transportation is a kind of holistic and integrated operation process about the mutual coordination and cooperation between multimodal transportation network nodes, node enterprises lines, social economic conditions, and natural environment factors. Risk factors existed in several aspects of the multimodal transportation network, including multimodal transportation enterprises, transportation channels, external economic, social environment, and natural factors; these risk factors are not only the necessary conditions for the formation of multimodal transportation network risk but also an important prerequisite for the existence of multimodal transportation network risk.

The transportation risk is one of the important factors of the multimodal transportation selection in this paper, but it can be found from the domestic and foreign documents' researches that few scholars considered transportation risk as the important factor, when they conducted researches in the choice of transportation mode and multimodal transportation problems; so the present researches cannot be well adapted to the actual situation of multimodal transportation modes selection; it is also difficult to give correct guidance to practice and reference.

2. Establishment of the Optimization Model

2.1. Problem Description. We suppose that a company has a batch of goods to transport from origin point O to destination point D through a multimodal transportation network; a plurality of hub nodes composed the multimodal transportation network; several transportation modes can be chosen between two arbitrary adjacent nodes; each node is a transit hub for different transportation modes; when we switched the goods from one mode to another mode, we need to spend some time and cost, but total transportation time cannot exceed the scope of delivery time. We hoped to deliver the goods safely, economically, and conveniently through the proper choice of the transportation modes during the agreed period.

2.2. Making the Assumption

- (1) We have known and fixed all the relevant constants;
- (2) there is one transportation mode existent at least on any section of known delivery path;

- (3) the same batch of goods cannot be separated during the transport process; that is to say, we can only choose one transportation mode in the known arbitrary section path when delivering one batch of goods;
- (4) the goods are transited at hub nodes, and the transit for each batch of goods in each hub node occurs one time at most.
- (5) when choosing the transportation model, just consider the influence of the transportation operation risk, not including the others.

2.3. Description of the Model Parameters

$C_{i,i+1}^k$ is the per unit cost of the selected k transportation mode from the node i to the node $i + 1$.

$d_i^{k,l}$ is the unit transfer cost from the transportation mode k to transportation mode l in the node i .

$t_{i,i+1}^k$ is transportation time of the selected k transportation mode from the node i to the node $i + 1$.

$f_i^{k,l}$ is transfer time from the transportation mode k to transportation mode l in the node i .

RT is the risk factors of the delivery goods.

$RC_{i,i+1}^k$ is the transportation risk factor of the selected k transportation mode from the node i to the node $i + 1$.

$RF_i^{k,l}$ is the transfer risk factor from the transportation mode k to transportation mode l in the node i .

$A_{i,i+1}^k$ is the transportation ability of the selected k transportation mode from the node i to the node $i + 1$. Consider

$$\begin{aligned}
 x_{i,i+1}^k &= \begin{cases} 1, & \text{select } k \text{ transportation mode from} \\ & \text{the node } i \text{ to the node } i + 1 \\ 0, & \text{otherwise,} \end{cases} \\
 y_i^{k,l} &= \begin{cases} 1, & \text{from the transportation mode} \\ & k \text{ to transportation mode } l \text{ in the node } i \\ 0, & \text{otherwise.} \end{cases} \tag{1}
 \end{aligned}$$

T is the actual goods delivery time, w is the earliest goods delivery time, and v is the latest goods delivery time, $v \geq T \geq w > 0$.

I_a is each unit goods warehousing cost when delivery advanced and V_a is each unit goods penalty cost when delivery is late.

J_a is all goods warehousing cost when delivery advanced; $J_a = \max[0, (w - T)I_a]$.

S_d is all goods penalty cost when delivery is late; $S_d = \max[0, (T - v)V_d]$;

M is a sufficiently large penalty factor.

2.4. Establishment of the Model

- (1) Objective 1: the least transportation cost for the decision of multimodal transportation scheme is as follows:

$$\begin{aligned} \text{Min } Z = & \sum_{i=1}^n \sum_{k=1}^m C_{i,i+1}^k X_{i,i+1}^k + \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m d_i^{k,l} y_i^{k,l} \\ & + \max[0, (w - T)I_a] + \max[0, (T - v)V_d]. \end{aligned} \quad (2)$$

- (2) Objective 2: the least transportation time for the decision of multimodal transportation scheme is as follows:

$$\text{Min } T = \sum_{i=1}^n \sum_{k=1}^m t_{i,i+1}^k x_{i,i+1}^k + \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m f_i^{k,l} y_i^{k,l}. \quad (3)$$

- (3) Objective 3: the lowest transportation risk for the decision of multimodal transportation scheme is as follows:

$$\begin{aligned} \text{Min } R = & RT + \sum_{i=1}^n \sum_{k=1}^m x_{i,i+1}^k RC_{i,i+1}^k \\ & + \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m y_i^{k,l} RF_i^{k,l}. \end{aligned} \quad (4)$$

2.5. Constraints of the Model

- (1) We can only choose one transportation mode between two nodes. Consider

$$\sum_{k=1}^m x_{i,i+1}^k = 1, \quad \forall i \in n. \quad (5)$$

- (2) Only transportation modes can be changed in a node. Consider

$$\sum_{k=1}^m \sum_{l=1}^m y_i^{k,l} = 1, \quad \forall i \in n. \quad (6)$$

- (3) Ensure continuity during the transportation process. Consider

$$x_{i-1,i}^k + x_{i,i+1}^l \geq 2y_i^{k,l}, \quad \forall i \in n, \forall k \in m, \forall l \in m. \quad (7)$$

- (4) Goods transportation time should meet the actual time range allowed. Consider

$$\begin{aligned} w \leq & \sum_{i=1}^n \sum_{k=1}^m t_{i,i+1}^k x_{i,i+1}^k \\ & + \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m f_i^{k,l} y_i^{k,l} \leq v, \quad v > w > 0. \end{aligned} \quad (8)$$

- (5) Cargo weight should not exceed the transportation capacity. Consider

$$Qx_{i,i+1}^k \leq A_{i,i+1}^k, \quad \forall i \in n, \forall k \in m, \forall l \in m. \quad (9)$$

- (6) The decision variable is 0 or 1. Consider

$$\begin{aligned} x_{i,i+1}^k & \in \{0, 1\}, \quad \forall i \in n, \forall k \in m, \forall l \in m, \\ y_i^{k,l} & \in \{0, 1\}, \quad \forall i \in n, \forall k \in m, \forall l \in m. \end{aligned} \quad (10)$$

- (7) Nonnegative constraints of variables. Consider

$$\begin{aligned} t_{i,i+1}^k & \geq 0, \quad \forall i \in n, \forall k \in m, \forall l \in m, \\ f_i^{k,l} & \geq 0, \quad \forall i \in n, \forall k \in m, \forall l \in m. \end{aligned} \quad (11)$$

2.6. Model Processing. Multiobjective programming model is difficult to solve because of the incoordination of the target vectors and the existence of constraints; in this paper, the objective function was simplified into a single objective function; usually we use the linear weighted method directly and sum multiple objective functions into a single objective function, but this method is only applicable to the transformation of the dimensionless objective function or uniform dimension objective function; it must be dimensionless or a dimensionally unified treatment for the multiobjective problem whose dimension is not uniform. We should change the original objective functions (2), (3), and (4) into the following:

$$\begin{aligned} \text{Max } Z' = & Z_{\max} - \sum_{i=1}^n \sum_{k=1}^m C_{i,i+1}^k X_{i,i+1}^k \\ & - \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m d_i^{k,l} y_i^{k,l} - J_a - S_d, \\ \text{Max } T' = & T_{\max} - \sum_{i=1}^n \sum_{k=1}^m t_{i,i+1}^k x_{i,i+1}^k \\ & - \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m f_i^{k,l} y_i^{k,l}, \\ \text{Max } R' = & R_{\max} - RT - \sum_{i=1}^n \sum_{k=1}^m x_{i,i+1}^k RC_{i,i+1}^k \\ & - \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m y_i^{k,l} RF_i^{k,l}. \end{aligned} \quad (12)$$

According to the evaluation function method of multiple objective function optimization method, referring to the weight degree of each target weight, we gave, respectively, the transportation cost, transportation time, and transportation risk weight as α , β , and δ and $\alpha + \beta + \delta = 1$, $\alpha, \beta, \delta \in [0, 1]$.

At the same time, in order to solve the dimensional consistency problem of the objective function, we need to transform the target function (12):

$$\begin{aligned} \text{Max } Z^* &= \left(Z_{\max} - \sum_{i=1}^n \sum_{k=1}^m C_{i,i+1}^k X_{i,i+1}^k \right. \\ &\quad \left. - \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m d_i^{k,l} y_i^{k,l} - J_a - S_d \right) \\ &\quad \times (Z_{\max})^{-1}, \\ \text{Max } T^* &= \left(T_{\max} - \sum_{i=1}^n \sum_{k=1}^m t_{i,i+1}^k x_{i,i+1}^k \right. \\ &\quad \left. - \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m f_i^{k,l} y_i^{k,l} \right) \\ &\quad \times (T_{\max})^{-1}, \\ \text{Max } R^* &= \left(R_{\max} - RT - \sum_{i=1}^n \sum_{k=1}^m x_{i,i+1}^k RC_{i,i+1}^k \right. \\ &\quad \left. - \sum_{i=1}^n \sum_{k=1}^m \sum_{l=1}^m y_i^{k,l} RF_i^{k,l} \right) \\ &\quad \times (R_{\max})^{-1}. \end{aligned} \quad (13)$$

In the objective function (13), Z^* , T^* , and R^* are the goal function after dimensionless treatment, the range for the three is $[0, 1]$. Among them, Z_{\max} , T_{\max} , and R_{\max} are the respective maxima of target functions (2), (3), and (4) in each generation of the genetic algorithm; so, in this paper, we established the comprehensive objective function:

$$f(Z, T, R) = \alpha \times Z^* + \beta \times T^* + \delta \times R^*. \quad (14)$$

The states and effects of the three indexes weights in the comprehensive evaluation results are not the same in the comprehensive objective function (14); it is so important for the model solution when choosing the weight values. Generally speaking, in the decision of intermodal transportation scheme, the selected weight values mostly depend on the type of goods and the decision preference.

3. Particle Swarm Optimization (PSO)

3.1. Introduction of the Algorithm. Particle swarm optimization is an evolutionary computation technique, proposed in 1995 by Dr. Eberhart and Dr. Kennedy, from the behavior of

birds' predation. The algorithm is initially rules by birds cluster activities of the enlightenment, using swarm intelligence to establish a simplified model. Particle swarm optimization algorithm is based on the observation of animal collective behavior; because of information sharing, the whole population movement from disorderly to orderly evolution process in the problem space by the individual in the group, then, obtained the optimal solution. PSO is similar to genetic algorithm; it is an optimization algorithm based on iteration. The system is initialized with a set of random solutions, searching for the optimal value through the iterative, but it has no genetic algorithm with the crossover and mutation; the particles search in the solution space to follow the optimal particle; PSO is simple and easy to realize and there is no need to adjust many parameters.

3.2. The Advantages of PSO Algorithm

- (1) The algorithm is simple, with less adjustable parameters, and easy to implement.
- (2) The random initialization population has strong global searching ability, which is similar to genetic algorithm.
- (3) It uses the evaluation function to measure the individual searching speed.
- (4) It has strong scalability.

3.3. The Disadvantages of PSO Algorithm

- (1) The algorithm cannot make full use of the feedback information in the system.
- (2) Its ability to solve combinatorial optimization problems is not strong.
- (3) Its ability to solve the optimization problem is not very well.
- (4) It is easy for this algorithm to obtain local optimal solution.

3.4. The Basic Principle of the Algorithm. The algorithm is described as follows: hypothesis in a D dimension target searching space, N particle to form a community, the i particle vector to represent a D dimension, and the position vector. Consider

$$X_i = (x_{i1}, x_{i2}, \dots, x_{iD}), \quad i = 1, 2, \dots, N. \quad (15)$$

The velocity of vector i particle is also a D dimension, the velocity vector

$$V_i = (v_{i1}, v_{i2}, \dots, v_{iD}), \quad i = 1, 2, \dots, N. \quad (16)$$

The i particle is the searched best position called individual extreme value by far and is denoted as

$$P_{\text{best}} = (p_{i1}, p_{i2}, \dots, p_{iD}), \quad i = 1, 2, \dots, N. \quad (17)$$

The optimal position of the whole particle swarm searched so far is called global extreme and is denoted as

$$g_{\text{best}} = (p_{g1}, p_{g2}, \dots, p_{gD}). \quad (18)$$

After finding out these two optimal values, particles updated their velocity and position according to the following formulas:

$$v_{id} = w * v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}), \quad (19)$$

$$x_{id} = x_{id} + v_{id}. \quad (20)$$

Among them, w is the inertia factor; c_1 and c_2 are the learning factors, and r_1 and r_2 are the ranges of $[0, 1]$ for the uniform random numbers. The formula (19) is composed of three parts on the right: the first part reflects the movement of particles and represents particles which have maintained their previous velocity trend; the second part reflects memory particles on the historical experience and represents particles which are close to the optimal position of its historical trend; the third part reflects the cooperation and knowledge, sharing group history experience synergy between particles.

4. Ant Colony Algorithm (ACO)

4.1. Introduction of the Algorithm. Ant colony algorithm (ACO) is a new intelligent optimization algorithm proposed by Italian scholars M. Dorigo, V. Maniezzo, and A. Colorni in the twentieth century and early 90s; it has been applied to the TSP, secondary distribution and quadratic assignment problems, graph coloring problem, and many classical combination optimization problems and gained very good results.

4.2. The Advantages of ACO Algorithm

- (1) It uses a kind of positive feedback mechanism and gets converges to the optimal path on the objective through the pheromone update ceaselessly.
- (2) It is a kind of optimization method of distribution; it is easy to imply in parallel.
- (3) It is a kind of method for global optimization, used not only for the single objective optimization problems, but also for solving the multiobjective optimization problems.
- (4) It is suitable to solve the discretization problems.
- (5) It has strong robustness.

4.3. The Disadvantages of ACO Algorithm

- (1) The solving speed is slow because of the shortage in the initial stage of information algorithm.
- (2) The algorithm based on the discrete problem cannot directly solve continuous optimization problems.
- (3) This algorithm lacks strong searching ability.

4.4. The Basic Principle of the Algorithm. The basic principle of the algorithm is using K ants jointly to construct a feasible solution; each ant is an independent one for the construction process of solution; the ants between each other exchange information through information value and constantly optimize the cooperative solution and at the same time give the ants distribution list (recorded the visited point of this ant); introduce the other one ant to follow the first ant's behavior; continue to search in the remaining points. The ant's task is under many constraints; look for communicating nodes one by one, and leave the pheromone in the connected node arc. Use arc pheromone and heuristic information to determine assignment order and use local optimization of pheromone to update; after all, the k ants are constructed completing the application, then apply to the global pheromone update.

$\tau_{ij}(t)$ stated at the t moment in the node i, j connection information; α is the importance degree of residual information in ant path i, j ; η_{ij} is the heuristic information, the expected degree of ants from node i to node j ; in this paper, $\eta_{ij} = 1/d_{ij}$ (d_{ij} is the distance between node i and node j); β is the importance degree of the heuristic information. The initial time of nodes, each ant was placed on the random selection path construction according to the transition probabilities; the K ant's transfer probability of node i to node j at t time is P_{ij}^k , which is feasible in the ant K sentinel node i set. Consider

$$P_{ij}^k(t) = \begin{cases} 0, & \text{otherwise,} \\ \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{i \in N_i^k} [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}, & j \in N_i^k. \end{cases} \quad (21)$$

According to the rules and update time determined by different update quantity of information, all the ants of each group after the completion of travel in the set the pheromone should be adjusted; the ant density model was used in this paper; that is to say, ants moved from node i to node j and updated pheromone of the path i, j . Among them, $\Delta\tau_{ij}^k$ shows the number of unit length of messages of ant K in the path from i to j ; γ is evaporation coefficient of pheromone; constant Q is ants leaving tracks number. Consider

$$\Delta\tau_{ij}^k(t, t+1) = \begin{cases} Q, & \text{ant } k \text{ transfers from node } i \text{ to} \\ & \text{node } j \text{ at the } t \text{ time to } t+1 \text{ time,} \\ 0, & \text{otherwise.} \end{cases} \quad (22)$$

Update rules corresponding information is as follows:

$$\tau_{ij}(t+1) = (1 - \gamma) \tau_{ij}(t) + \Delta\tau_{ij}(t, t+1), \quad (23)$$

$$\Delta\tau_{ij}(t, t+1) = \sum_{k=1}^m \Delta\tau_{ij}^k(t, t+1). \quad (24)$$

5. Particle Swarm and Ant Colony Algorithm (PSACO)

5.1. Introduction of the Algorithm. In this paper, PSACO algorithm which combines the particle swarm algorithm and ant colony algorithm is proposed, combining the advantages of two kinds of algorithms and make up for the disadvantages of the two algorithms. Particle swarm algorithm is more suitable for solving continuous optimization problems and is slightly inferior in solving combinatorial optimization problems, but, due to the random distribution of the initial particle, we use it to solve the combinatorial optimization problems; the algorithm has better global search ability and fast solution speed; the ant colony algorithm is better in solving combinatorial optimization problems than the particle swarm optimization algorithm, because the initial distribution is uniform, which makes the ant colony algorithm with blindness at the beginning of the algorithm, so it does not converge very quickly. PSACO algorithm combines particle swarm algorithm and ant colony algorithm, absorbs the advantages of the two algorithms, and overcomes the shortcomings; the advantage is complementary and better than that of ant colony algorithm in time efficiency and the particle swarm algorithm in accuracy and efficiency.

5.2. The Basic Ideas and Steps of PSACO Algorithm. The PSACO algorithm combines advantages of particle swarm algorithm and ant colony algorithm, and its basic idea is as follows: in the first stage, the PSACO algorithm makes full use of random, rapid, global features of the particle swarm optimization algorithm, and, after a certain number of iterations, finds out the suboptimal solution of the problem and adjusts the initial distribution; in the second stage of the algorithm, PSACO algorithm uses information distribution in the first stage, makes full use of parallelism, positive feedback, and high accuracy advantages of ant colony algorithm, and completes the entire problem solving progress. Specifically, according to the characteristics of the solution problem, combining two groups of algorithms is effective to solve the combinatorial optimization problem of multimodal transportation choice mode.

The PSACO algorithm is divided into two parts; firstly, particle swarm algorithm generates transportation connectivity nodes from node start to node finish and then uses the ant colony algorithm to find the optimal combination of the sequence nodes of the transportation mode; the basic steps are as follows.

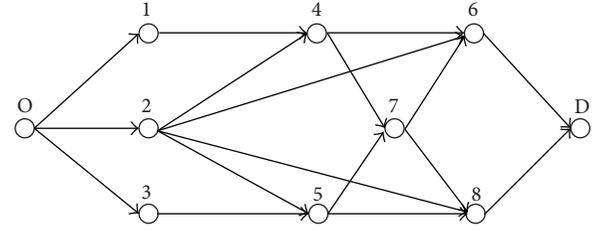


FIGURE 1: Schematic diagram of a multimodal transportation network.

Step 1. Particle swarm algorithm parameters and variables are initialized.

Step 2. Set the ant colony algorithm parameters initialization and each ant's position as follows:

- (1) according to the selection probability formula (21) and the roulette wheel method, choose the path for each ant;
- (2) when structural path, each ant updates pheromone locally at the same time according to the formula (23);
- (3) execute (1), (2), and (3) circularly until every ant generates a complete path;
- (4) record and calculate the objective function value of each ant according to the tabu list and give the optimal and worst ant;
- (5) update global pheromone for the best ant and worst ant according to formula (23);
- (6) execute (2)–(6) until they meet the end condition.

Step 3. Call the ant colony algorithm to get each particle's fitness and update the historical optimum particle position vector and global optimum particle position vector.

Step 4. Update the particle velocity vector and the position vector according to the formula (19) and (20).

Step 5. Execute Steps 2–4 to satisfy the end conditions and use the ant colony algorithm to get the optimal solution.

Step 6. End.

6. Example

For example, a transportation enterprise needs to transport 100 tons of goods from city O to city D ; each passing section has four kinds of transportation modes to select. Figure 1 shows the schematic diagram of a multimodal transportation network. Table 1 shows the transportation modes that we can choose between two adjacent city nodes, the transportation costs, transportation time, and risk values about different transportation modes. Table 2 shows the unit transfer cost, unit transit time, and transfer risk about different transportation modes. The cargo delivery time interval was [130 h, 165 h]

TABLE 1: Transportation costs/transportation time/transportation risk/transportation capacity of different transportation modes.

Start node	Terminal node	Transportation mode	Transportation costs/yuan	Transportation time/hour	Transportation risk	Transportation capacity
O	1	Railway	35	1	1	120
O	1	Road	20	2	3	115
O	1	Waterway	18	1.5	1	90
O	1	Aviation	40	0.5	2	120
O	2	Railway	25	0.8	1	100
O	2	Road	15	1.5	2	115
O	2	Waterway	13	1	1	120
O	2	Aviation	35	0.3	2	125
O	3	Railway	30	1.2	1	120
O	3	Road	18	2.2	2.5	95
O	3	Waterway	13	1.8	1.2	110
O	3	Aviation	36	0.5	1.8	100
1	4	Railway	45	1.5	1	90
1	4	Road	25	2.5	3	110
1	4	Waterway	20	2	1	120
1	4	Aviation	42	0.7	2	110
2	4	Railway	50	2	2	100
2	4	Road	30	3	4	105
2	4	Waterway	30	2.5	2	105
2	4	Aviation	44	0.9	2	90
2	5	Railway	55	2.5	2	100
2	5	Road	35	3.5	4	100
2	5	Waterway	32	2.8	2	130
2	5	Aviation	46	1.2	2	95
2	6	Railway	65	3.5	3	95
2	6	Road	45	4.5	4.5	100
2	6	Waterway	33	3.5	3	110
2	6	Aviation	50	1.8	2.5	110
2	8	Railway	70	4	4	110
2	8	Road	50	5	5	100
2	8	Waterway	58	4	3	150
2	8	Aviation	55	2	2.5	100
3	5	Railway	43	1.4	1	100
3	5	Road	22	2.3	2.5	110
3	5	Waterway	20	2	1	120
3	5	Aviation	40	0.8	1.8	110
4	6	Railway	24	1	1	100
4	6	Road	16	1.5	2	115
4	6	Waterway	24	1	1	120
4	6	Aviation	30	0.3	2	125
4	7	Railway	35	2	3	105
4	7	Road	25	3	4	95
4	7	Waterway	20	2	2	120
4	7	Aviation	38	1.5	2.5	100
5	7	Railway	35	2	3.5	100
5	7	Road	24	2.5	3.5	100
5	7	Waterway	30	1.8	2	115

TABLE 1: Continued.

Start node	Terminal node	Transportation mode	Transportation costs/yuan	Transportation time/hour	Transportation risk	Transportation capacity
5	7	Aviation	36	1.5	2.5	100
5	8	Railway	35	1	1	120
5	8	Road	20	2	3	115
5	8	Waterway	32	1.5	1	90
5	8	Aviation	40	0.5	2	120
6	D	Railway	30	1.5	2	100
6	D	Road	20	2	3	95
6	D	Waterway	18	1.5	2	120
6	D	Aviation	35	1	2	125
7	6	Railway	40	1.5	2	100
7	6	Road	20	2.5	2.5	110
7	6	Waterway	24	2	1	120
7	6	Aviation	43	1	2	110
7	8	Railway	42	2	3	100
7	8	Road	20	2.5	3	110
7	8	Waterway	25	2.5	2	120
7	8	Aviation	45	1.5	4	110
8	D	Railway	45	1.5	1	90
8	D	Road	25	2.5	3	110
8	D	Waterway	20	2	1	120
8	D	Aviation	42	0.7	2	110

TABLE 2: Unit transfer costs (yuan)/unit transit time (second)/transfer risk of different transportation modes.

	Railway	Road	Waterway	Aviation
Railway	0/0/0	2/2/2	3/4/3	4/4/3
Road	2/2/2	0/0/0	3/4/4	4/3/4
Waterway	3/4/3	3/4/4	0/0/0	4/3/3
Aviation	4/4/3	4/3/4	4/3/3	0/0/0

in the contract; the goods' penalty cost coefficient was 6 yuan/hour, when delivery advanced, and 10 yuan/hour, when delivery is late, the risk factor to send the goods was 3. How to choose the multimodal transportation path and let the transportation enterprise find the optimal scheme to save transportation costs, reduce the total transportation time, and reduce the total risk.

According to the example, by combining decision makers' multimodal transportation needs, the weights of transportation costs, transportation time, and transportation risk value were set to 0.6, 0.1, and 0.3 (different decisions will lead to different weights setting); this paper adopted MATLAB to verify the effectiveness of the proposed algorithm; first the 20 iterations of particle swarm algorithm were the initial solution of the problem; then we used the results of the initial solution by ant colony algorithm for distribution. $\gamma = 0.02$ in ant colony algorithm; the number of ants was equal to the multimodal transportation city node number; $\alpha = 2$ and $\beta = 5$;

Table 3 shows the comparison of PSACO algorithm and basic ant colony algorithm in solving ability and efficiency.

It can be seen from the simulation results that the PSACO algorithm is compared with the basic ACO algorithm; it has not only better results than the ant colony algorithm, but also higher computation efficiency; that is, because, after the particle swarm algorithm, initial pheromone distribution was improved, avoiding the basic ACO algorithm for blind initial pheromone distribution searching; thus, it is helpful for ant colony algorithm to get more accurate searches. Table 4 shows the optimal transport scheme.

7. Conclusion

This paper is about the decision problem of multimodal transportation scheme; it searched and analyzed relevant research literatures at home and abroad, combined them with the diverse needs of transportation scheme decision, and proposed and established combinatorial optimization mathematical model of intermodal transportation plan decision based on transportation time and transportation cost and risk; at the same time, it used heuristic algorithm which combined particle swarm optimization algorithm and ant colony algorithm to solve the model and got the transportation scheme optimization. The experiments showed that the algorithm has achieved very good results in time performance and optimization performance; it was a feasible algorithm. The content of this paper enriched and perfected

TABLE 3: The results and comparison of PSACO algorithm.

Algorithm	The comprehensive evaluation value	Transportation time/hour	Transportation costs/yuan	Transportation risks	Operate time/second
PSACO algorithm	0.438	150.6	4258	6.7	28
ACO algorithm	0.486	156.8	4456	7.3	43

TABLE 4: The optimal transport scheme.

The optimal transport scheme	
Transportation route	O-2-4-7-8-D
Transportation mode	Railway-road-waterway-railway-aviation

the theories and methods of the hot issues in the study of intermodal transportation scheme decision, and the results provided the references and reasonable arrangements for the transportation process and transportation project decision making.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] K. Chen and L. Ma, "Glowworm swarm optimization algorithm for 0-1 knapsack problem," *Application Research of Computers*, vol. 4, no. 30, pp. 993–995, 2013.
- [2] B. Li, X. Q. Yan, and J. X. Hu, "Model of container terminal logistics optimization system based on the Harvard structure and swarm intelligence," *Journal of Jiangsu University of Science and Technology*, vol. 3, no. 25, pp. 282–286, 2011.
- [3] J. Mi, "Improved swarm intelligence optimization algorithm used in the task sorting," *Journal of Beijing Information Science and Technology University*, vol. 3, no. 25, pp. 13–18, 2010.
- [4] H. Li, X. Chen, and M. Zhang, "The path planning problem based on swarm intelligence algorithm," *Journal of Tsinghua University*, vol. 3, pp. 1770–1773, 2007.
- [5] B. Wu and Z. Z. Shi, "A kind of TSP problem based on ant colony algorithm partition algorithm," *Journal of Computers*, vol. 12, pp. 1328–1333, 2001.
- [6] R.-B. Xiao, Z.-W. Tao, and H.-F. Zou, "Concurrent tolerance optimization design based on hybrid swarm intelligence algorithm," *Computer Integrated Manufacturing Systems*, vol. 13, no. 4, pp. 668–674, 2007.
- [7] Q. L. Ma, W. N. Liu, D. H. Sun, and Y. F. Dan, "Travel route planning model based on fusion of multi intelligent algorithm," *Computer Science*, vol. 10, pp. 211–221, 2010.
- [8] A. Lozano and G. Storchi, "Shortest viable path algorithm in multimodal networks," *Transportation Research A: Policy and Practice*, vol. 35, no. 3, pp. 225–241, 2001.
- [9] H. I. Calvete, C. Galé, M.-J. Oliveros, and B. Sánchez-Valverde, "A goal programming approach to vehicle routing problems with soft time windows," *European Journal of Operational Research*, vol. 177, no. 3, pp. 1720–1733, 2007.
- [10] Y. Li and J. Zhao, "Choice of multimodal transportation modal with fixed freight," *Journal of Southwest Jiaotong University*, vol. 47, no. 5, pp. 881–887, 2012.
- [11] X. Wang, Z. B. Chi, and X. L. Ge, "Whole vehicle multimodal transportation model research and analysis with time windows," *Computer Research and Application*, vol. 28, no. 2, pp. 563–565, 2011.
- [12] F. L. Feng and Q. Y. Zhang, "Organization collaborative optimization model of multimodal transportation based on environment," *Journal of Railway Science and Engineering*, vol. 9, no. 6, pp. 95–100, 2012.
- [13] L. Tong and L. Nie, "Multimodal transportation path optimization model and method research," *Logistics Technology*, vol. 29, no. 5, pp. 57–60, 2010.
- [14] J. W. Zhang, "Multimodal container transportation network path rationalization process," *Journal of Beijing Jiaotong University*, 2011.
- [15] L. Li and Y. C. Zeng, "Multimodal transportation model and algorithm in logistics transportation," *Statistics and Decision Making*, vol. 20, pp. 27–29, 2009.
- [16] J. Wen, "Optimization model to control the cost of container multimodal transportation," *Changsha Science and Technology University*, 2010.
- [17] L. Li, "Multimodal transportation modes generalized cost optimal choice under conditions," *Journal of Beijing Jiaotong University*, 2006.
- [18] X. X. Wang, "The multimodal transportation minimum cost problem with time window," *Beijing Posts and Telecommunications University*, 2009.
- [19] J. Zhang, "Railway container transportation network model and algorithm," *Central South University*, 2007.
- [20] Z. J. Jia, "The risks analysis and control of the goods multimodal transportation," *Chongqing Jiaotong University*, 2009.
- [21] F. Liu, "The container multimodal transportation corridor planning and transportation modes selection," *Journal of Southwest Jiaotong University*, 2011.
- [22] X. K. Luo, "Optimization algorithm research of logistics and multimodal transportation," *Shanghai Jiaotong University*, 2004.

Research Article

Comparison of Three Different Curves Used in Path Planning Problems Based on Particle Swarm Optimizer

J. J. Liang,¹ H. Song,¹ B. Y. Qu,² and Z. F. Liu²

¹ School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China

² School of Electric and Information Engineering, Zhongyuan University of Technology, Zhengzhou 450007, China

Correspondence should be addressed to B. Y. Qu; qby1984@hotmail.com

Received 23 December 2013; Accepted 11 March 2014; Published 17 April 2014

Academic Editor: Fang Zong

Copyright © 2014 J. J. Liang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In path planning problems, the most important task is to find a suitable collision-free path which satisfies some certain criteria (the shortest path length, security, feasibility, smoothness, and so on), so defining a suitable curve to describe path is essential. Three different commonly used curves are compared and discussed based on their performance on solving a set of path planning problems. Dynamic multiswarm particle swarm optimizer is employed to optimize the necessary parameters for these curves. The results show that Bezier curve is the most suitable curve for producing path for the certain path planning problems discussed in this paper. Safety criterion is considered as a constrained condition. A new constraint handling method is proposed and compared with other two constraint handling methods. The results show that the new method has a better characteristic to improve the performance of algorithm.

1. Introduction

The mobile robot path planning is an important research field of robotics. One of the most important tasks to realize navigation and control of the robots is path planning. In an environment with obstacles, the aim of path planning is to find a suitable collision-free path, which satisfies some certain optimal criteria (such as the shortest path length, security, and feasibility), for a mobile robot to move from a start position to a target position. Most researches have focused on finding the shortest path, the minimum-time path, or the safest path, but the generated paths may be discontinued. Smoothness of the path is essential for the navigation of mobile robots, because nonsmooth motions have effect on slip [1–5]. So finding a suitable curve to describe the path is a very important task in path planning problems.

In [6], η^3 curve with parallel variable-length genetic algorithm has been used to realize path planning problems. Ferguson is another curve which is used commonly with particle swarm optimizer in [7, 8] and particle filter in [9]. Bezier curve is one of the most common curves which is combined with de Casteljau algorithm in [10], genetic

algorithm in [11], and particle swarm optimizer in [12] in recent years. These curves which are used to generate curve have their own specific characteristics and requirements, but which one is the best has not been discussed.

The approaches of traditional path planning are artificial potential field [13], neural network [14], D^* algorithm [15], and so on. With the appearance and development of evolutionary computation algorithms, many nature inspired optimization computing methods have been proposed to solve path planning problems, including genetic algorithms [16, 17] and differential evolution [18, 19].

Particle swarm optimization (PSO) which was proposed by Eberhart and Kennedy in 1995 [20, 21] is based on swarm intelligence. It has been applied to many areas successfully such as artificial neural network training [22], path planning problems [23, 24], multiworking modes product-color planning [25], and robust control of 3RPS parallel manipulators [26], for its easiness to use, robustness, and strong ability of global optimization. An improved particle swarm optimizer is applied to solve the path planning problems in this paper.

The robotic path planning problem is to find a suitable path for a mobile robot to move from the start location to

the target location, which satisfies some optimum criteria in an environment full of obstacles. In this paper, we define the security and the shortest path as the optimum criteria. The security means no collision between the robot and all the obstacles and the shortest path describes the distance the robot moves from the initial point to the end point [5, 27]. The three times Bezier curve, Ferguson curve, and η^3 curve are used to generate the path and their performances are compared. With these curve generating methods the path planning can be transformed into optimizing a few limited anchor points which are used to form the path. Then dynamic multiswarm particle swarm optimizer (DMS-PSO) is employed to optimize the locations of these anchor points.

In the previous work, two different constraint handling methods, dynamic threshold ε and dynamic balance function, have been tested [28]. Based on the analysis on the weakness of these two constraint handling methods, a novel constraint handling method "dynamic compared Δ " is proposed to be incorporated into the algorithm to improve the search efficiency. The experimental result shows that this new method has a better performance on most path planning problems discussed in this paper.

The rest of this paper is organized as follows. The characteristics of the three curves are introduced in detail in Section 2. Section 3 gives a brief introduction on the dynamic multiswarm particle swarm optimizer and the constraint handling mechanisms employed in this work. The experimental setup and the results are presented in Section 4. Conclusions and future work are given in Section 5.

2. Description of Curves

2.1. The Definition and Properties of Bezier Curve. Bezier curve was proposed by the French engineer Pierre Bezier, who used Bezier curve to design for the body of the car in 1962 [29]. In recent years, Bezier curve was applied to various occasions for its advantages on describing both straight line and curve.

A Bezier curve of degree n is a parametric curve composed of Bernstein basis polynomials of degree n [22]:

$$P(t) = \sum_i^N P_i B_{i,N}(t), \quad t \in [0, 1]. \quad (1)$$

In this equation, basis function $\{B_{i,T}(t)\}$ is a famous n times Bernstein polynomial [23], which is defined as

$$B_{i,N}(t) = C_N^i t^i (1-t)^{N-i} \quad (i = 0, 1, \dots, n). \quad (2)$$

The parameter equation of every point for three times Bezier curve could be generated by formulas (1) and (2) as follows:

$$P(t) = P_0(1-t)^3 + 3P_1t(1-t)^2 + 3P_2t^2(1-t) + P_3t^3, \quad (3)$$

where t is in the range of $[0, 1]$. Bezier curve starts at $t = 0$ and ends at $t = 1$.

The properties of Bezier curves [22] can be described as follows.

- (1) Bezier curves start at the start point and stop at the end point.
- (2) First derivatives of the start point and the end point are only related to the two near control points and in the same direction of the line of the two points.

The calculation formula

$$P'(0) = 3 \times (P_1 - P_0), \quad P'(n) = 3 \times (P_n - P_{n-1}). \quad (4)$$

A complex first-order continuous Bezier curve can be formed by connecting several segments of low-order Bezier curves. Each segment has four control points. Assuming we have two segments, $P_1(P_{10}, P_{11}, P_{12}, P_{13})$ and $P_2(P_{20}, P_{21}, P_{22}, P_{23})$, in order to ensure the continuousness of the curve after connection, the following equation should be satisfied:

$$P_{13} - P_{12} = P_{21} - P_{20}, \quad P_{13} = P_{20}. \quad (5)$$

Therefore, in order to meet the property of first-order continuous when using n segments of Bezier curves to describe a path, $2n$ points ($4n$ parameters) are needed. The path can be generated using the following:

$$P(t) = \begin{cases} P_0(1-t)^3 + 3P_1^i t(1-t)^2 + 3P_2^i t^2(1-t) + P_3^i t^3, & i = 1 \\ P_3^{i-1}(1-t)^3 + 3(2P_3^{i-1} - P_2^{i-1})t(1-t)^2 + 3P_2^i t^2(1-t) + P_3^i t^3, & 1 < i < n \\ P_3^{i-1}(1-t)^3 + 3(2P_3^{i-1} - P_2^{i-1})t(1-t)^2 + 3P_2^i t^2(1-t) + P_1 t^3, & i = n, \end{cases}$$

$$P(t) = [x(t), y(t)]^T, \quad (6)$$

where P_0 represents the start point while P_1 stands for the end point. When t changes in the interval $(0, 1)$, we can get a cubic Bezier curve of segment i . These n segments of cubic Bezier curve constitute the entire path of the curve.

2.2. The Properties of Ferguson Curve. Ferguson curve is also a famous curve which has many excellent properties and plays an important role in the shape description.

Since Ferguson curve is smooth and easy to implement, it is also often used to describe the path in path planning problems. One segment of Ferguson curve can be defined as follows:

$$C(t) = [x(t), y(t)]^T,$$

$$k : C(t) = P_0 F_1(t) + P_1 F_2(t) + P_0' F_3(t) + P_1' F_4(t), \quad (7)$$

$$t \in [0, 1].$$

P_0 and P_1 are the start point and the end point of the curve, respectively, and P_i and P_i' are control points which control

the shape of the curve. F_i represents Ferguson polynomial and is defined as follows:

$$\begin{aligned} F_1(t) &= 2t^3 - 3t^2 + 1, \\ F_2(t) &= -2t^3 + 3t^2, \\ F_3(t) &= t^3 - 2t^2 + t, \\ F_4(t) &= t^3 - t^2. \end{aligned} \quad (8)$$

Assuming that a curve consists of n segments of Ferguson curves, these Ferguson curves should satisfy certain requirements. Two-segment Ferguson curves are taken as an example to illustrate the requirements they should satisfy to ensure smooth connection between these two segments. Suppose that the other Ferguson curve is described as follows:

$$\bar{k} : C(t) = \bar{P}_0 F_1(t) + \bar{P}_1 F_2(t) + \bar{P}_0' F_3(t) + \bar{P}_1' F_4(t). \quad (9)$$

In order to make the path smooth, the curve which is used to describe the path must be first-order continuous, and then k and \bar{k} must satisfy

$$P_1 = \bar{P}_0, \quad P_1' = \bar{P}_0'. \quad (10)$$

The same as Bezier curve, if n segments Ferguson curves are used to generate the path, there will be $2n$ control points which means that $4n$ variables are to be optimized.

2.3. The Properties of η_3 Curve. The same as the Bezier curve and Ferguson curve, η_3 curve which is used in path planning problems for its good properties in describing lines, arcs, and clothoid is also a widely used curve. As shown in [8], first we set two arbitrary combinations as follows:

$$\begin{aligned} \Omega_A &= \left[x_A, y_A, \theta_A, k_A, \dot{k}_A \right]^T, \\ \Omega_B &= \left[x_B, y_B, \theta_B, k_B, \dot{k}_B \right]^T. \end{aligned} \quad (11)$$

Here x , y , and θ represent coordinates and direction, respectively, while k and \dot{k} denote curvature and curvature derivative of the path at one point. A 7th-order polynomial of η_3 curve can be formed by the following formulas:

$$\begin{aligned} P(t) &= [x(t), y(t)]^T, \\ P_x(t) &= \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3 \\ &\quad + \alpha_4 t^4 + \alpha_5 t^5 + \alpha_6 t^6 + \alpha_7 t^7; \quad t \in [0, 1], \\ P_y(t) &= \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 \\ &\quad + \beta_4 t^4 + \beta_5 t^5 + \beta_6 t^6 + \beta_7 t^7. \end{aligned} \quad (12)$$

In order to ensure the smoothness of the curve after connecting, the following formula should be satisfied:

$$\begin{aligned} \Omega_A &= \left[x_A, y_A, \theta_A, k_A = 0, \dot{k}_A = 0 \right]^T, \\ \Omega_B &= \left[x_B, y_B, \theta_B, k_B = 0, \dot{k}_B = 0 \right]^T. \end{aligned} \quad (13)$$

In addition, the polynomial has extra six degrees of freedom. In order to reduce the calculation of degrees of freedom, we use Euclidean distance of two terminal configurations to represent some variables of vectors while the other variables of vectors are set to 0. Therefore, x coordinate coefficients used to generate the curve can be obtained according to the above formulas as follows:

$$\begin{aligned} \alpha_0 &= x_A, \\ \alpha_1 &= \|(x_A - x_B, y_A - y_B)\| \cos \theta_A; \\ \gamma_1 &= \|(x_A - x_B, y_A - y_B)\| \cos \theta_B; \\ \alpha_2 &= 0; \quad \gamma_2 = 0; \\ \alpha_3 &= 0; \quad \gamma_3 = 0; \\ \alpha_4 &= 35(x_B - x_A) - 20\alpha_1 - 10\alpha_2 \\ &\quad - 4\alpha_3 - 15\gamma_1 + 5\gamma_2 - \gamma_3; \\ \alpha_5 &= -84(x_B - x_A) + 45\alpha_1 + 20\alpha_2 \\ &\quad + 6\alpha_3 + 39\gamma_1 - 14\gamma_2 + 3\gamma_3; \\ \alpha_6 &= 70(x_B - x_A) - 36\alpha_1 - 15\alpha_2 \\ &\quad - 4\alpha_3 - 34\gamma_1 + 13\gamma_2 - 3\gamma_3; \\ \alpha_7 &= -20(x_B - x_A) + 10\alpha_1 + 4\alpha_2 \\ &\quad + \alpha_3 + 10\gamma_1 - 4\gamma_2 + \gamma_3. \end{aligned} \quad (14)$$

y coordinate coefficients can be obtained by changing $\cos \theta$ into $\sin \theta$. When m segments of η_3 curve are used to describe the path, $m + 1$ control points are needed. However, the start point and the end point are known in the path planning problems discussed in this paper; thus the number of the control points which are needed to be optimized is $m - 1$. In other words, there are $2(m - 1)$ variables to be optimized. Except the location of the control points, the tangent directions of each control point for the path are also controllable, so there are other $m + 1$ points to be optimized for m segments. Therefore, there are $3m - 1$ variables for an η_3 curve with m segments.

So from the above information, we could know that $x(t)$ and $y(t)$ are the coordinates of every point which should be optimized. What is more, these three curves are smooth and suitable for path planning for robots. If the same number of segments is needed to generate the path, there are $4n$ parameters to be optimized for the first two curves and $3n - 1$ variables for η_3 curve.

3. Brief Introduction about Algorithm and Constraint Handling Mechanisms

Particle swarm optimizer is an intelligent evolutionary algorithm which is constructed by mimicking the birds' behavior of preying food [21]. The basic idea of particle swarm optimization algorithm is to find the optimal solution through collaboration among groups and information sharing among individuals.

The idea of dynamic multiswarm based on periodically changed neighborhood structure was firstly proposed by Liang and Suganthan in 2005 [27]. The good information obtained by each subswarm is exchanged among the subswarms and the diversity of the population is increased simultaneously by using the dynamic changing topology. Considering its good performance on complex optimization problems, the dynamic multiswarm particle swarm optimizer (DMS-PSO) is employed to solve the path planning problems in this paper.

The position updating equations of DMS-PSO with crossover can be described as follows [28]:

If $\text{rand} < 0.5$

$$\begin{aligned} V_i^d &\leftarrow \omega * V_i^d \\ &+ c_1 * \text{rand}1_i^d * (pbest_i^d - X_i^d) \\ &+ c_2 * \text{rand}2_i^d * (lbest_k^d - X_i^d), \\ V_i^d &\leftarrow \min(V_{\max}^d, \max(-V_{\max}^d, V_i^d)), \\ X_i^d &\leftarrow X_i^d + V_i^d. \end{aligned} \quad (15)$$

Otherwise

$$X_i^d \leftarrow pbest_i^d,$$

where X_i^d represents the position of the i th particle in dimension d . V_i^d represents the velocity of the i th particle in dimension d . $pbest_i^d$ is the best position in history of the i th particle in dimension d . V_{\max}^d is the predefined maximum value in dimension d .

DMS-PSO was firstly used in path planning problems in [5], where the path planning problem has been solved by the following means.

- (1) Security and the shortest path criteria are combined into a punitive function with a constant to balance them.
- (2) Path length is regarded as the objective function, while the security criterion is regarded as a constraint for the shortest path.

In these path planning problems, a series of circles are used to represent obstacles, and the safe distance between path and obstacles is set as D_{safe} (which is radius of the circle). The minimum distance between path and obstacles is d_{min} . If and only if d_{min} is larger than D_{safe} , the path could be defined as

secure. Otherwise, penalty will be imposed. f_{safe} is treated as security penalty function as follows:

$$f_{\text{safe}} = \begin{cases} 0, & d_{\text{min}} > D_{\text{safe}} \\ d_{\text{min}}, & 0 \leq d_{\text{min}} \leq D_{\text{safe}}, \end{cases} \quad (16)$$

$$f_{\text{len}} = \min_{o \in C_{\text{obs}}} \min_{t \in [0,1]} \sqrt{(x(t) - o_x)^2 + (y(t) - o_y)^2},$$

where o_x and o_y are the centre of the obstacles and C_{obs} is a collection of all the obstacles in the space. The total cost is calculated as

$$f(x) = f_{\text{len}}(x) + \alpha f_{\text{safe}}(x), \quad \alpha = 1000. \quad (17)$$

α is a constant value, which is used to balance the proportion of f_{safe} and f_{len} . A large α will lead to local optimum easily while a small α will make a collision with obstacles, so choosing a suitable value is difficult.

In order to overcome this drawback, two constraint handling methods have been used to improve the above static constrain in [27]; FEs is the current fitness evaluation times and MaxFEs is the predefined max fitness evaluation times.

Constraint Handling Method 1 (Dynamic Threshold ϵ). One has

$$\epsilon = \max \left(\min \left(D_{\text{safe}} \left(1 - 2 * \frac{\text{FEs}}{\text{MaxFEs}} \right), \text{mean}(f_{\text{safe}}), 0 \right) \right). \quad (18)$$

Figure 1 describes the dynamic changing process of ϵ . x is considered to be better than y if

- (1) $f_{\text{safe}}(x) \leq \epsilon$ and $f_{\text{safe}}(y) \leq \epsilon$ and $f_{\text{len}}(x) < f_{\text{len}}(y)$
 - (2) $f_{\text{safe}}(x) \leq \epsilon$ and $f_{\text{safe}}(y) > \epsilon$
 - (3) $\epsilon < f_{\text{safe}}(x) < f_{\text{safe}}(y)$.
- (19)

We observed that the potential good solutions which locate near to the global optimum but do not satisfy the current constraint will be replaced. In this way, the useful information obtained along the search process may be lost.

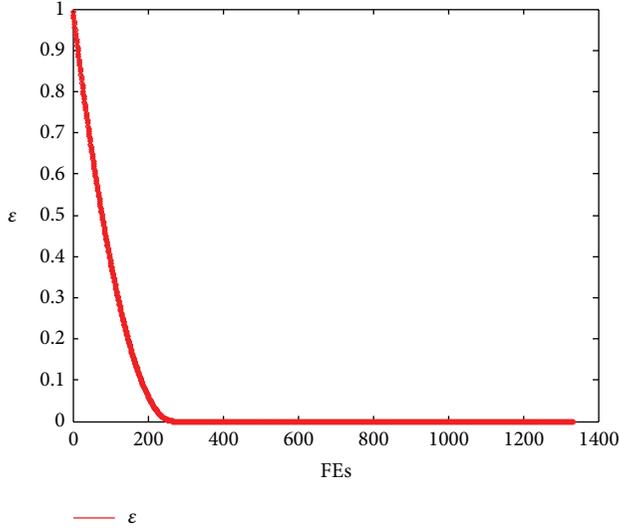
Constraint Handling Method 2 (Dynamic Balance Function). This method is similar to the previous static penalty function (17) except that the balance factor α is gradually increasing. The dynamic α is defined as follows:

$$\alpha = 10000 * \max \left(\max(\text{mean}(f_{\text{safe}}), 0.2), \frac{\text{FEs}}{\text{MaxFEs}} \right)^2. \quad (20)$$

x is considered to be better than y if

$$F(x) < F(y). \quad (21)$$

The dynamic changing process of α with the FEs is presented in Figure 2. α is changed with the FEs and its rising


 FIGURE 1: Dynamic change of ε .

trend is gentle and continuous. It is better than the static penalty function, but it is still difficult to control and select a suitable value to avoid losing some potential solutions.

Constraint Handling Method 3 (Dynamic Compared Function, Described with Δ). Constraint handling methods 1 and 2 improved the feature of algorithm which has been discussed in previous work, but they still have some drawbacks: the first method may lose some potential solutions which have been abandoned for dissatisfying constraint condition in current generation while the second method may not find the best solution for its gentle change. On the other hand, the first one has large space while the second has small space to be improved. A new constraint handling method which has a larger constraint range is introduced to overcome the shortage of the first constraint handling method. It is expected to have a better ability of global search. In this new method, two different ε , ε_1 and ε_2 , are employed to judge if a solution satisfies the constraint. And the mean value of f_{safe} values of current particles is used to control the value of ε_1 :

$$\begin{aligned} \varepsilon_1 &= \min \left(\max \left(D_{\text{safe}} * \left(\max \left(1 - \frac{\text{FEs}}{(0.2 * \text{MaxFEs})}, 0 \right) \right)^2, \text{mean}(f_{\text{safe}}) \right), \right. \\ &\quad \left. D_{\text{safe}} * \left(\max \left(1 - \frac{\text{FEs}}{(0.9 * \text{MaxFEs})}, 0 \right) \right)^{0.5} \right); \\ \varepsilon_2 &= D_{\text{safe}} * \left(\max \left(1 - \frac{\text{FEs}}{(0.5 * \text{MaxFEs})}, 0 \right) \right)^2. \end{aligned} \quad (22)$$

Figure 3 provides the possible range of ε_1 and ε_2 . ε is the effective range of ε_1 and ε_2 . The new constraint method is generated by the comparison of ε_1 and ε_2 . For any two solutions x and y to be compared, the following comparison criterion is used:

$$tp = \begin{cases} (1) f_{\text{safe}}(x) < \varepsilon_2, f_{\text{safe}}(y) \leq \varepsilon_2, f_{\text{len}}(x) < f_{\text{len}}(y) \\ (2) f_{\text{safe}}(x) < \varepsilon_2, f_{\text{safe}}(y) > \varepsilon_2 \\ (3) \varepsilon_2 \leq f_{\text{safe}}(x) < f_{\text{safe}}(y), f_{\text{safe}}(x) \leq \varepsilon_1 \\ (4) f_{\text{safe}}(x) > \varepsilon_2, f_{\text{safe}}(y) > \varepsilon_2, f_{\text{len}}(x) < f_{\text{len}}(y). \end{cases} \quad (23)$$

If tp is equal to 1, x is considered to be better than y . This constraint handling method overcomes the shortage of the dynamic threshold ε which may be trapped into local optimum and improves the exploration property of the algorithm.

4. Experimental Setup and Results

From the previous test in the path planning problems, some conclusions have been made that DMS-PSO with crossover outperforms DMS-PSO and PSO with crossover performs better than PSO. So in this task, DMS-PSO with crossover and PSO with crossover are combined with the above three

constraint handling methods which are designed to test the characteristics of the curves in path planning problems.

- (1) The following six algorithms are used to test characteristics of each curve in path planning problems:
 - (i) PSO- ε : basic particle swarm optimizer with dynamic ε and crossover operator;
 - (ii) PSO-DP: basic particle swarm optimizer with dynamic balance and crossover operator;
 - (iii) PSO- Δ : basic particle swarm optimizer with dynamic compared Δ and crossover operator;
 - (iv) DMS-PSO- ε : dynamic multiswarm particle swarm optimizer with dynamic ε and crossover operator;
 - (v) DMS-PSO-DP: dynamic multiswarm particle swarm optimizer with dynamic balance and crossover operator;
 - (vi) DMS-PSO- Δ : dynamic multiswarm particle swarm optimizer with dynamic compared Δ and crossover operator.
- (2) Some parameters settings during the experiment are as follows:

MaxFEs (the max fitness evaluation): 40000;

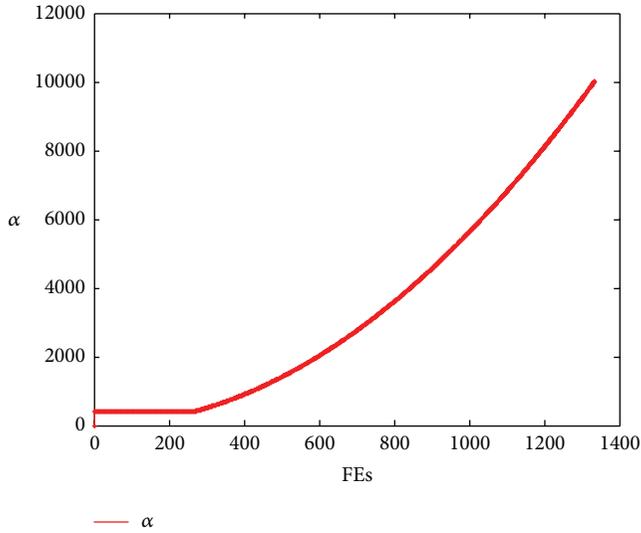


FIGURE 2: Dynamic change of α .

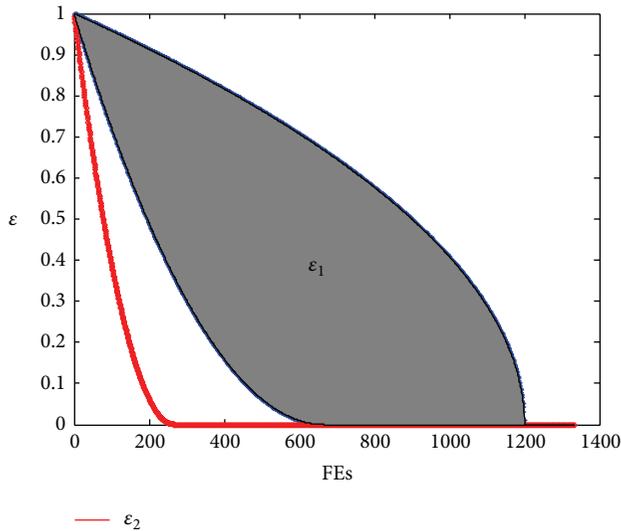


FIGURE 3: Dynamic change of ε_1 and ε_2 (plot in condition of $R = 1$ which means the radius of obstacles is 1 and the maximum ε is 1).

independent runs for every algorithm: 25;
 population size of PSO: 30;
 number of subswarms in DMS-PSO: 10;
 particles in each subswarm in DMS-PSO: 3.

- (3) The settings of every curve and the parameters need to be optimized.

In this task, three different curves are used to generate path. The segment of each curve and parameters needed to be optimized are set in detail as follows. Generally speaking, more points make the path more smooth and complex, while fewer points take less time in optimization.

- (a) The first case is that the segments of every curve are uniform ($n = 2$), so there are 8 parameters to be optimized in Bezier curve and Ferguson curve while 5 parameters should be optimized in η^3 curve.
- (b) The second case is that the optimized parameters of every curve are equal (8 parameters for all curves), so three segments of η^3 curve are used to describe the path.

4.1. Comparison of Best Satisfied Paths for Each Problem. Eight artificial designed path planning problems which have different properties are used to test characteristics of Bezier curve, Ferguson curve, and η^3 curve. In order to show how the robot moves in an environment full of obstacles, the following landscapes with the best path of these tested problems are plotted in Figure 4. The yellow circles describe the dangerous distance around the obstacles. What is more, A represents the start point while B stands for the end point.

These eight path planning problems can be classified into two classes. F1, F2, F3, and F4 are simple problems which have less local optima and are easier to find the shortest path that satisfies the safety criterion. F5, F6, F7, and F8 can be classified into complex problems which have more local optima and make the algorithms be easily trapped into the local optima.

4.2. Comparison Results of the Different Curves. Nonparametric statistical method t -test is used to evaluate the difference between two algorithms. For each problem, the results of the best algorithm which obtains the best average value in the 25 independent runs are compared with those of other algorithms by t -test method. $h = 1$ indicates a rejection of the null hypothesis at the 5% significance level. $h = 0$ indicates a failure to reject the null hypothesis at the 5% significance level.

Case 1. Two segments for all curves are used to describe the path, so there are eight points for Bezier curve and Ferguson curve to be optimized while five parameters are needed for η^3 curve. The experiment results are listed in Tables 1 to 3.

Some conclusions could be drawn from Table 1 as follows.

- (1) DMS-PSO outperforms PSO in all constraint handling methods correspondingly, which shows that DMS-PSO has better global search ability.
- (2) The result of t -test 2 shows that there is no obvious difference between these two algorithms, so this phenomenon is regarded as these two algorithms have the similar performance on these problems. But DMS-PSO- Δ performs better on problems F1, F2, F4, F5, and F6 while DMS-PSO- ε outperforms on F4, F6, and F7 on average.
- (3) Compared with the best solutions obtained by DMS-PSO- ε and DMS-PSO- Δ , the distribution of optimal solutions of DMS-PSO-DP is significantly different on problems F2, F3, F4, F5, F6, and F7 which could be seen from the results of t -test 2.

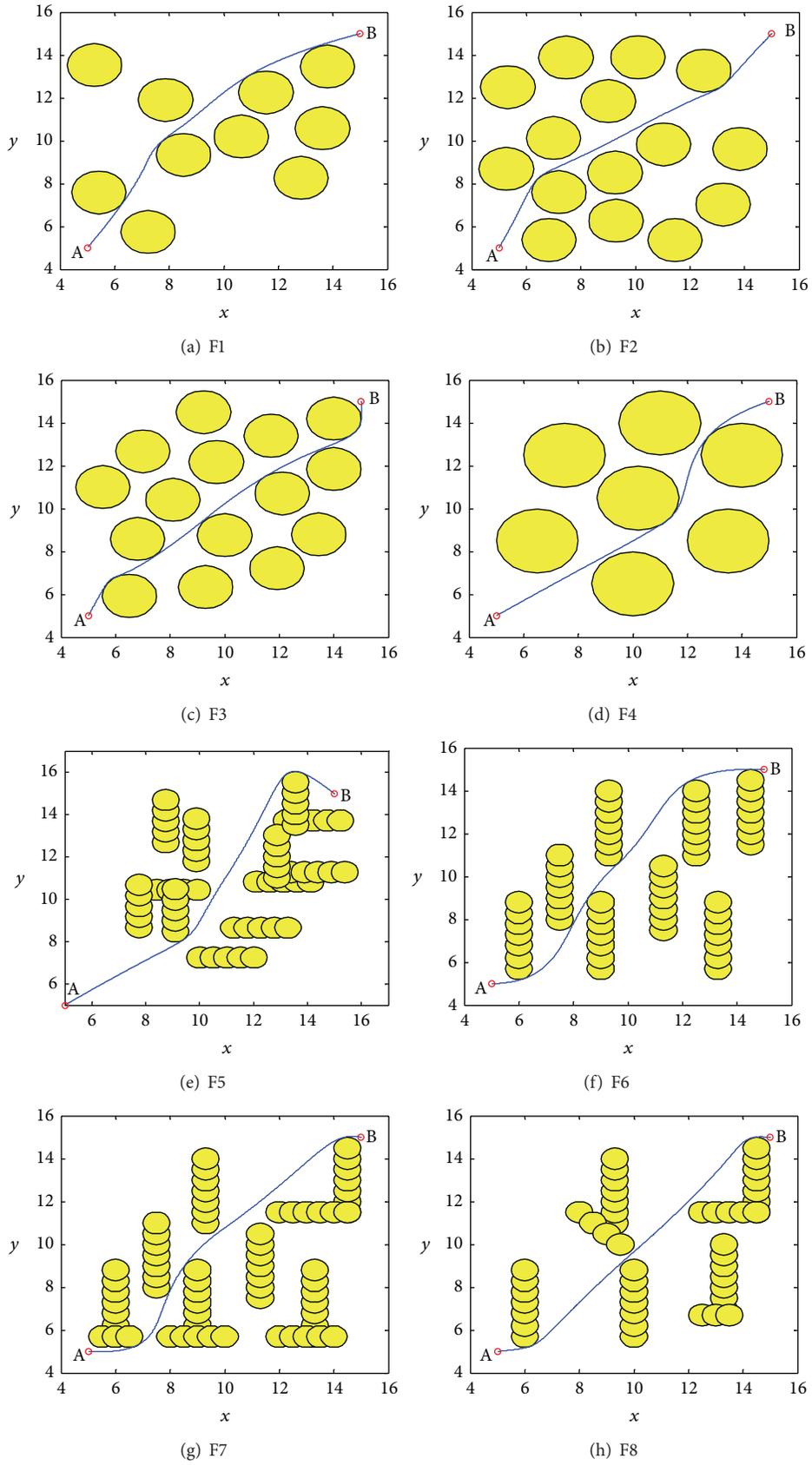


FIGURE 4: Landscapes of the test problems.

TABLE 1: Result of Bezier curve.

Problems	PSO- ϵ	PSO-DP	PSO- Δ	DMS-PSO- ϵ	DMS-PSO-DP	DMS-PSO- Δ	
F1	Mean	14.9058	15.1830	14.8969	14.9294	14.7792	14.7559
	Std.	0.2910	0.4454	0.0282	0.4494	0.0500	0.0087
	Min	14.6726	14.6638	14.6969	14.6547	14.6566	14.6626
	Max	16.9975	16.9843	15.2326	17.1550	15.7071	15.0132
	h	0	1	1	0	0	—
F2	Mean	14.7260	15.6209	15.1451	14.6492	14.7087	14.6486
	Std.	0.0034	1.8989	0.1415	0.00039	0.0034	0.0002
	Min	14.6533	14.6889	14.8150	14.6251	14.6274	14.6242
	Max	14.8920	19.4449	16.2972	14.7009	14.8882	16.3361
	h	0	1	1	0	1	—
F3	Mean	15.9481	16.7858	16.5723	14.9475	16.3286	14.9701
	Std.	1.1006	1.1209	0.9196	0.2976	0.4483	0.1306
	Min	14.8502	15.7943	15.1810	14.7240	15.6394	14.7290
	Max	18.8319	18.7404	19.0984	17.2080	17.2206	15.7963
	h	1	1	1	—	1	0
F4	Mean	14.8243	15.9185	15.4251	14.7366	15.0635	14.7363
	Std.	0.0407	1.5323	1.2592	$8.586E - 5$	0.2979	$1.451E - 5$
	Min	14.7430	14.7225	14.7851	14.7150	14.7151	14.7313
	Max	15.7812	18.1838	19.8821	14.7713	17.3141	14.7455
	h	1	1	1	0	1	—
F5	Mean	16.5000	16.6254	16.4928	16.3466	16.3800	16.3361
	Std.	0.0544	0.1889	0.0048	0.0007	0.0029	0.0012
	Min	16.3327	16.3102	16.3666	16.2868	16.3246	16.2827
	Max	17.3413	17.9407	16.6432	16.4001	16.5518	16.4123
	h	0	1	1	0	1	—
F6	Mean	16.2485	16.4073	16.2432	15.3892	16.1031	15.2860
	Std.	0.3219	0.2167	0.2125	0.2282	0.3366	0.0103
	Min	15.3576	15.2832	15.5306	15.2319	15.2461	15.2400
	Max	16.6793	16.6649	16.8466	17.2616	16.6399	15.7535
	h	1	1	1	0	1	—
F7	Mean	16.2040	16.4125	16.4490	15.1967	15.9038	15.2458
	Std.	0.4474	0.3349	0.2576	0.1794	0.5311	0.2027
	Min	15.1081	15.1079	15.1656	15.0355	15.0516	15.0504
	Max	16.6948	16.8264	16.9785	16.6090	16.6292	16.9341
	h	1	1	1	—	1	0
F8	Mean	15.0239	15.2377	14.8827	14.6644	14.6787	14.6699
	Std.	0.5322	0.6693	0.0905	0.0003	0.0011	0.0003
	Min	14.6303	14.6327	14.6697	14.6291	14.6262	14.6296
	Max	16.6699	16.6459	16.2430	14.7046	14.7831	14.7093
	h	1	1	1	—	0	0

These three points show that although DMS-PSO- Δ and DMS-PSO- ϵ have the same characteristic on t -test 2, DMS-PSO- Δ overcomes the drawback of DMS-PSO- ϵ which is easy to be trapped into local optimum on average. Compared with other algorithms, the feature of DMS-PSO- Δ stands out on path planning problems where Bezier curve is used to generate path. The result also tells that Bezier curve is suitable on path planning problems for its stable feature when we employ evolutionary algorithm to optimize its parameters.

Table 2 gives us the following information.

- (1) DMS-PSO has a better global search ability compared with PSO on the whole.
- (2) All best solutions about Ferguson curve spread in DMS-PSO- ϵ and DMS-PSO- Δ , while the best results of problems F1, F6, F7, and F8 accept DMS-PSO-DP in the distribution of optimal solutions for 25 independent runs. The result of problem F6 has no difference with DMS-PSO-DP, DMS-PSO- ϵ , and DMS-PSO- Δ on the distribution of 25 independent

TABLE 2: Result of Ferguson curve.

Problems	PSO- ε	PSO-DP	PSO- Δ	DMS-PSO- ε	DMS-PSO-DP	DMS-PSO- Δ	
F1	Mean	15.4745	14.9230	15.8539	14.6720	14.7393	15.0440
	Std.	1.0221	0.3217	0.8098	$3.065E - 5$	0.1175	0.1913
	Min	14.6617	14.6565	14.7062	14.6626	14.6532	14.6601
	Max	17.3684	16.3953	17.5340	14.6830	16.3743	16.3967
	h	1	1	1	—	0	1
F2	Mean	14.6541	14.8765	14.7016	14.6399	14.6550	14.6395
	Std.	$1.507E - 4$	0.9416	0.0013	$2.085E - 5$	$2.167E - 4$	$6.640E - 5$
	Min	14.6335	14.6344	14.6415	14.6320	14.6309	14.6245
	Max	14.6787	19.5190	14.8158	14.6481	14.6860	14.6531
	h	1	1	1	0	1	—
F3	Mean	17.6613	16.8770	15.1573	14.8069	16.3085	15.7254
	Std.	1.0504	1.1620	0.9308	0.0133	0.4618	0.0223
	Min	15.5203	15.5515	14.6219	14.6170	15.9185	15.3465
	Max	19.6494	19.4469	19.7183	15.2238	17.8514	15.9672
	h	1	1	1	—	1	1
F4	Mean	14.7474	15.4290	14.7654	14.7361	14.7525	14.7364
	Std.	$1.037E - 4$	1.0917	$7.884E - 4$	$2.494E - 5$	$1.735E - 4$	$2.205E - 5$
	Min	14.7356	14.7337	14.7378	14.7291	14.7373	14.7288
	Max	14.7714	18.3901	14.8401	14.7497	14.7928	14.7488
	h	0	1	1	—	1	0
F5	Mean	18.9011	17.8758	17.8816	17.7712	17.5169	16.8374
	Std.	19.3752	0.9286	4.7299	4.3717	0.3750	0.7757
	Min	16.4034	16.4773	16.6205	16.4129	16.9094	16.3204
	Max	28.7950	19.3831	26.4327	25.2178	18.6831	17.8547
	h	1	1	1	1	1	—
F6	Mean	20.4977	17.2938	21.4213	16.7939	17.0746	16.5777
	Std.	21.7799	7.5871	23.4940	5.1557	1.6796	0.3785
	Min	17.4722	15.7391	15.4976	15.2908	15.7525	15.6083
	Max	29.3754	27.9776	29.9679	22.5225	21.5453	17.9068
	h	1	1	1	0	0	—
F7	Mean	23.5527	17.7175	19.9267	17.3071	18.5039	19.3598
	Std.	27.7538	0.5265	35.7375	15.6542	1.9334	32.9936
	Min	17.8796	16.8212	15.1296	15.0326	16.8277	15.6002
	Max	30.0131	20.5929	29.5523	28.4035	21.4855	30.3578
	h	1	1	1	—	0	0
F8	Mean	18.9859	15.9555	17.1540	15.4485	15.6366	18.54158
	Std.	8.6686	0.6078	11.4477	0.0218	0.6935	3.5024
	Min	17.4710	15.3587	15.4066	15.1707	15.4429	16.2014
	Max	28.6708	17.9973	28.8277	15.7448	19.6240	26.7133
	h	1	1	1	—	0	1

runs though there is much difference between these four algorithms on the mean value.

- (3) The constraint handling method ε is little better than dynamic balance function and dynamic compared Δ when path is produced by Ferguson curve.
- (4) Except for problems F1, DMS-PSO- ε outperforms obviously than other algorithms on the whole no matter on the result of mean value or t -test 2.

In a word, ε and Δ are suitable and DMS-PSO- ε and DMS-PSO- Δ are smart choices when Ferguson curve is applied in path planning problems especially on complex problems.

From Table 3, we could observe the following.

- (1) Dynamic multiswarm has improved the search ability of traditional particle swarm optimizer which means that DMS-PSO with crossover performs better than PSO under all constraint handling methods.

TABLE 3: Result of η_3 curve ($n = 2$).

Problems		PSO- ϵ	PSO-DP	PSO- Δ	DMS-PSO- ϵ	DMS-PSO-DP	DMS-PSO- Δ
F1	Mean	15.1590	15.3283	15.2681	14.9161	15.1670	14.8622
	Std.	0.3263	0.3658	0.4875	0.2284	0.1987	0.2007
	Min	14.6853	14.6728	14.6932	14.6608	14.6801	14.6614
	Max	16.98428	16.72636	17.6536	16.4115	16.2402	16.9015
	h	1	1	1	0	1	—
F2	Mean	15.0589	15.0396	15.7143	14.7317	14.9698	14.9601
	Std.	0.4267	0.2972	1.4186	0.0902	0.1457	0.4676
	Min	14.6589	14.6626	14.7605	14.65536	14.6640	14.6555
	Max	16.2536	16.3979	19.6726	16.1707	16.2063	17.2056
	h	1	1	1	—	1	0
F3	Mean	16.9309	16.7421	16.6452	16.2764	16.7070	15.8886
	Std.	0.3723	0.5148	0.8288	0.8953	0.3677	0.2801
	Min	15.6267	15.6372	15.6718	15.6049	15.7619	15.6068
	Max	17.4024	17.3621	19.6036	18.6914	17.3590	17.3939
	h	1	1	1	0	1	—
F4	Mean	15.1204	15.6049	15.0711	15.8886	15.3697	14.9027
	Std.	0.2261	0.7872	0.0990	0.2801	0.4204	0.4031
	Min	14.7725	14.7932	14.7791	15.6068	14.7799	14.7403
	Max	15.8243	18.1527	15.9261	17.3939	16.7863	17.9495
	h	1	1	1	—	1	0
F5	Mean	16.5948	16.4176	16.61561	16.4206	16.4134	16.4329
	Std.	0.1251	1.910E - 4	0.1206	0.0097	1.865E - 4	0.0102
	Min	16.4121	16.4029	16.4001	16.3722	16.3787	16.3842
	Max	17.6328	16.4500	17.5537	16.8877	16.4457	16.9001
	h	1	0	1	0	—	0
F6	Mean	16.6868	16.6765	16.7663	16.5985	16.5424	16.0209
	Std.	0.0188	0.0731	0.0481	0.1177	0.1248	0.3553
	Min	16.1306	15.4302	15.8551	15.4915	15.4417	15.3031
	Max	16.9352	16.9246	16.9671	16.9530	16.7489	16.8915
	h	1	1	1	1	1	—
F7	Mean	16.5802	16.3344	16.4030	16.2061	15.9442	15.6069
	Std.	0.2244	0.45243	0.3946	0.4480	0.4248	0.3488
	Min	15.2971	15.2954	15.3358	15.2914	15.2947	15.2917
	Max	16.9642	16.8365	16.9896	16.8971	16.8196	16.8491
	h	1	1	1	1	0	—
F8	Mean	15.4772	15.0808	15.5568	15.1574	15.0251	15.0064
	Std.	0.5238	0.0665	0.4020	0.2922	9.807E - 4	4.680E - 4
	Min	15.0152	14.9885	15.0526	14.9821	14.9887	14.9863
	Max	16.9498	16.3030	16.9771	16.9514	15.0988	15.0621
	h	1	0	1	0	1	—

(2) DMS-PSO- ϵ has a better mean value than other algorithms on the whole.

(3) DMS-PSO- Δ is better than DMS-PSO- ϵ and DMS-PSO-DP except for F2, F4, and F5 while DMS-PSO- Δ is similar to DMS-PSO- ϵ on problem F2 and DMS-PSO-DP+ on problem F5 on average.

(4) Except for problems F6 and F7, DMS-PSO- Δ and DMS-PSO- ϵ have similar performance.

(5) Although DMS-PSO-DP is better than other algorithms on F3, DMS-PSO- Δ , DMS-PSO- ϵ , and DMS-PSO-DP have the same acceptance which means they have no difference under 25 independent runs at the 5% significance level.

Generally speaking, DMS-PSO- Δ possesses a good feature so that it could be applied in path planning problems when η_3 curve is used to describe path.

TABLE 4: Result of $\eta\beta$ curve ($n = 3$).

Problems		PSO- ϵ	PSO-DP	PSO- Δ	DMS-PSO- ϵ	DMS-PSO-DP	DMS-PSO- Δ
F1	Mean	15.1269	15.2927	15.8549	14.8915	15.2233	14.8945
	Std.	0.2000	0.3962	0.6258	0.1053	0.2890	0.1533
	Min	14.6743	14.6922	14.8753	14.6599	14.6810	14.6746
	Max	16.4027	16.9510	17.1939	15.7512	16.5228	16.4695
	h	1	1	1	—	1	0
F2	Mean	15.1828	15.8735	17.3363	14.8915	15.5943	14.7678
	Std.	0.4999	0.5877	2.4413	0.1053	0.4150	0.0323
	Min	14.6786	14.6978	15.0084	14.6599	14.6649	14.6682
	Max	17.1109	17.5539	20.7949	15.7512	17.0959	15.5873
	h	1	1	1	0	1	—
F3	Mean	16.9903	17.3458	17.8328	16.2678	17.13297	15.3092
	Std.	0.3941	0.34185	1.5596	1.7428	0.1416	0.7615
	Min	15.1838	15.7395	15.8694	14.7341	15.9525	14.7554
	Max	17.6611	18.7940	20.4837	18.8251	17.4754	17.1826
	h	1	1	1	1	1	—
F4	Mean	15.1503	16.2299	16.5532	14.7421	16.1149	15.0292
	Std.	0.4168	1.0923	1.9814	$6.941E - 4$	1.2537	1.8730
	Min	14.7527	14.8084	14.9267	14.6988	14.7411	14.7161
	Max	17.4512	18.0449	19.4632	14.8068	18.0394	21.5966
	h	1	1	1	—	1	0
F5	Mean	16.8508	16.6488	16.9656	16.4054	16.5114	16.3798
	Std.	0.2432	0.1425	0.1367	0.0569	0.0180	0.0037
	Min	16.4201	16.3486	16.5099	16.3191	16.3300	16.3254
	Max	17.8626	17.4785	17.9822	17.5422	16.9302	16.6216
	h	1	1	1	0	1	—
F6	Mean	16.6472	16.5453	16.9163	16.2758	16.2548	15.7087
	Std.	0.0727	0.1016	0.0788	0.3668	0.2496	0.32256
	Min	15.4516	15.5077	16.6653	15.2211	15.2660	15.2387
	Max	16.9744	16.9084	18.13603	17.0442	16.73439	16.7263
	h	1	1	1	1	1	—
F7	Mean	16.6997	16.4166	16.8786	15.8356	16.1770	15.6007
	Std.	0.0129	0.3524	0.0291	0.5831	0.2656	0.3273
	Min	16.5565	15.1092	16.5886	15.0152	15.0720	15.0145
	Max	16.8951	16.9036	17.3592	16.7376	16.7343	16.7801
	h	1	1	1	0	1	—
F8	Mean	15.1227	14.9834	15.5447	14.6521	14.7546	14.6624
	Std.	0.3083	0.2759	0.5752	$7.786E - 4$	0.0216	0.0016
	Min	14.6780	14.6695	14.7240	14.6327	14.6510	14.6180
	Max	16.6986	16.6473	18.1863	14.7619	15.3003	14.7838
	h	1	1	1	—	1	0

Case 2. Eight parameters are satisfied to generate path; Table 4 is the result (the result concludes $\eta\beta$ curve only).

The following information is given from Table 4.

- (1) DMS-PSO is absolutely better than PSO in three-segment $\eta\beta$ curve no matter the result of mean value or the null hypothesis at the 5% significance level.
- (2) When DMS-PSO- ϵ is better than DMS-PSO- Δ on the mean value, the previous accepts the latter on all problems which means they have no difference.
- (3) When DMS-PSO- Δ is better than DMS-PSO- ϵ on the mean value, the previous rejects the latter on F3 and F6.

TABLE 5: Best result of every curve.

Problems		Bezier curve	Ferguson curve	η_3 curve $n = 2$	η_3 curve $n = 3$
F1	Mean	14.7559	14.6720	14.8622	14.8915
	Std.	0.0087	$3.065E - 5$	0.2007	0.1053
	Min	14.6626	14.6626	14.6614	14.6599
	Max	15.0132	14.6830	16.9015	15.7512
F2	Mean	14.6486	14.6395	14.7317	14.7678
	Std.	0.0002	$6.640E - 5$	0.0902	0.0323
	Min	14.6242	14.6245	14.65536	14.6682
	Max	16.3361	14.6531	16.1707	15.5873
F3	Mean	14.9475	14.8069	15.8886	15.3092
	Std.	0.2976	0.0133	0.2801	0.7615
	Min	14.7240	14.6170	15.6068	14.7554
	Max	17.2080	15.2238	17.3939	17.1826
F4	Mean	14.7363	14.7361	15.8886	14.7421
	Std.	$1.451E - 5$	$2.494E - 5$	0.2801	$6.941E - 4$
	Min	14.7313	14.7291	15.6068	14.6988
	Max	14.7455	14.7497	17.3939	14.8068
F5	Mean	16.3361	16.8374	16.4134	16.3798
	Std.	0.0012	0.7757	$1.865E - 4$	0.0037
	Min	16.2827	16.3204	16.3787	16.3254
	Max	16.4123	17.8547	16.4457	16.6216
F6	Mean	15.2860	16.5777	16.0209	15.7087
	Std.	0.0103	0.3785	0.3553	0.32256
	Min	15.2400	15.6083	15.3031	15.2387
	Max	15.7535	17.9068	16.8915	16.7263
F7	Mean	15.1967	17.3071	15.6069	15.6007
	Std.	0.1794	15.6542	0.3488	0.3273
	Min	15.0355	15.0326	15.2917	15.0145
	Max	16.6090	28.4035	16.8491	16.7801
F8	Mean	14.6644	15.4485	15.0064	14.6521
	Std.	0.0003	0.0218	$4.680E - 4$	$7.786E - 4$
	Min	14.6291	15.1707	14.9863	14.6327
	Max	14.7046	15.7448	15.0621	14.7619

(4) DMS-PSO- Δ and DMS-PSO- ε are better than DMS-PSO-DP on all problems.

So, when three-segment η_3 curve is used to generate path, it is clear to ensure DMS-PSO performs better than PSO. The constraint handling method dynamic threshold ε and dynamic compared Δ are better than dynamic balance function and dynamic compared Δ overcomes the drawback of dynamic threshold ε and outperforms it.

Case 3. The comparison of best result under each curve is as follows.

Having compared the best result of every curve, when two segments of Bezier curve, Ferguson curve, and η_3 curve are used to describe path, we could observe the following.

(1) Ferguson curve performs better than Bezier curve and η_3 curve on problems F1 to F4 which are easy to find global optimum and has a worse feature on

problems F5 to F8 which are described as complex problems.

(2) Although η_3 curve is not so good as Ferguson curve and Bezier curve on simple problem, it outperforms Ferguson curve on complex problem which means η_3 curve possesses a better search ability.

(3) Bezier curve is better in producing path compared with Ferguson curve and η_3 curve on the whole.

When three-segment η_3 curve is applied in these problems, conclusions could be made as follows.

(1) Compared with two segments, three-segment η_3 has a small range of the distribution between the best and worst solutions on the whole.

(2) There is no difference between problems F1, F2, F5, and F7 no matter it is two-segment or three-segment η_3 curve.

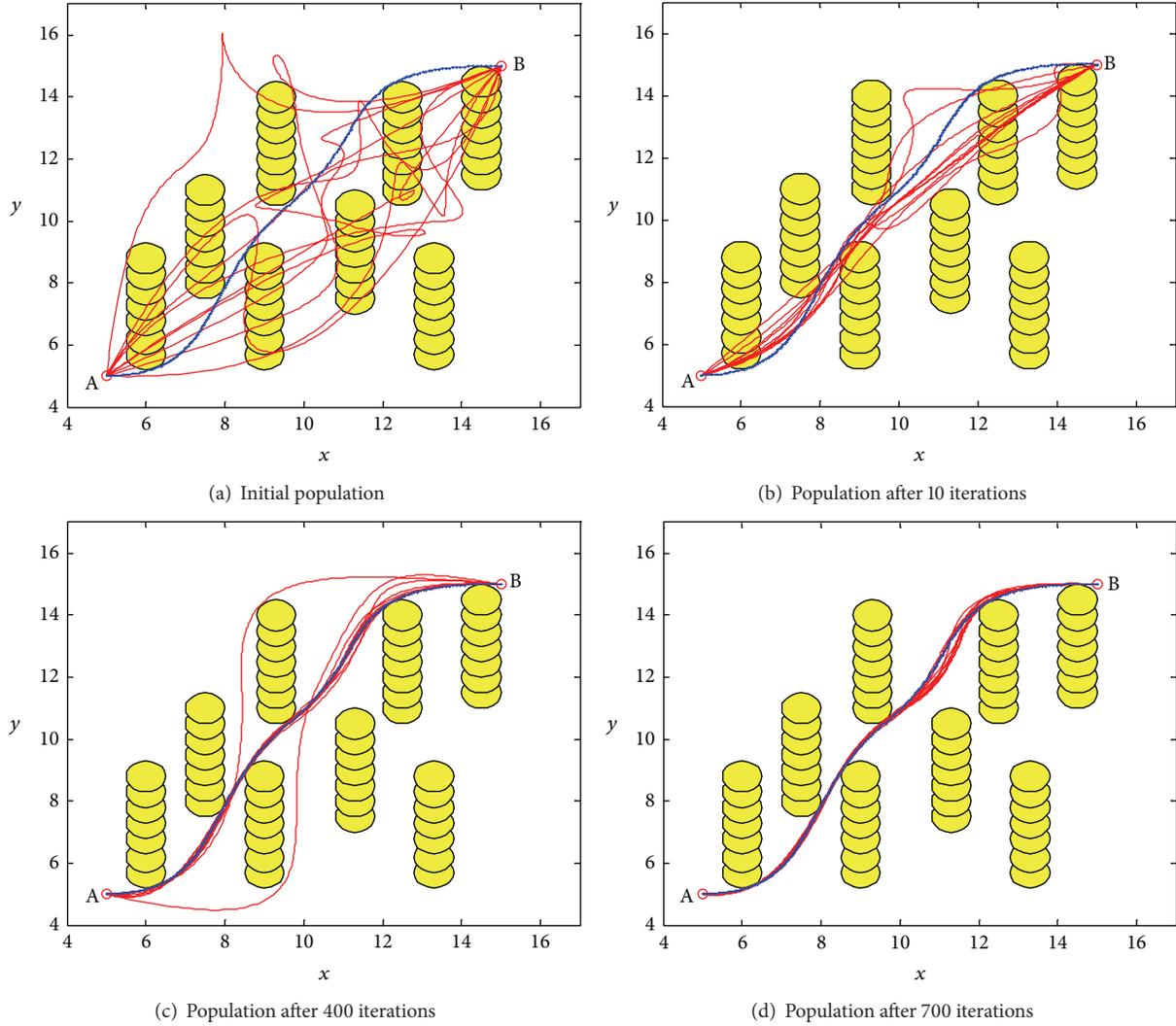


FIGURE 5: Processes of iteration.

- (3) Three-segment η_3 curve outperforms on problems F3, F4, F6, and F8 in generating path obviously.
- (4) On complex problems which are easily trapped into local optimum, three-segment η_3 curve is much better than two-segment curve which means that the previous has obvious difference compared to the latter such as F6, F8, F3, and F4.
- (5) Three-segment η_3 curve performs better than Ferguson on complex problems but obviously expresses an inferior characteristic in generating path compared with Bezier curve.

From so many points of discussion from Tables 1 to 5, conclusions could be made that DMS-PSO overcomes the drawback of PSO which is easy to fall into local optimal and premature. Dynamic compared Δ overcomes the drawback and inherits the advantage of dynamic threshold ϵ and shows better constraint characteristics than dynamic balance function, which makes it show good binding properties in path planning problems.

When all curves are composed by the same number of segments, η_3 curve outperforms Ferguson curve on complex problems specifically but is worse than Bezier curve for all problems. Fewer points would be optimized when η_3 curve is used to describe path, so less time is needed.

Three-segment η_3 curve is better than two-segment one for generating path because it has more anchor points to control and can generate a more flexible path, especially on complex problems.

Bezier curve expresses better performance on path planning problems compared with Ferguson curve and η_3 curve. The most possible reason may be that Bezier curve is easier to change the shape of the path via adjustment of a fixed number of anchor points than the other two curves. So Bezier curve is the most suitable curve to produce path in this paper and DMS-PSO with crossover combined with dynamic compared Δ is the best choice to optimize path in path planning problems.

In order to show the property of Bezier curve and DMS-PSO with crossover combined with dynamic compared Δ ,

complex problem F6 is an example to show how particles learn from their neighborhood and avoid being trapped into local optimum. The yellow circles describe the dangerous distance around the obstacles, red paths mean the current local paths, and blue path is the best path satisfying some certain criteria. The processes of iteration are in Figure 5.

Figure 5 shows the search process which could be seen that although the robot always runs into obstacles in the first 400 iterations, it is far away from obstacles step by step. After 700 iterations, solutions are converged into the best path gradually which shows that DMS-PSO with crossover combined with dynamic compared Δ has a good ability of global search in early stage and global convergence in latter stage.

5. Conclusion

In order to solve path planning problems in static environment, suitable curves and algorithms with constraint mechanisms are designed in this paper. Three curves are compared under six algorithms, and the results have proved that DMS-PSO has a better ability of global search than PSO again. At the same time, the analyses of three constraint handling methods and curves are carried on. Firstly, dynamic constraint methods are designed well for path planning problems compared with the previous work where static constraint is used. Then the dynamic compared Δ possesses a better feature than dynamic threshold ε and dynamic balance function which has overcome the drawback of dynamic threshold ε which might lose the previous good solutions found in the search process, which are near to the global optimum but do not satisfy the current constraint. So dynamic compared Δ is more suitable to be applied in path planning problems. From the results, we could observe that η^3 curve outperforms Ferguson curve when the same segment is used to describe path, especially on the distribution of solutions for complex problems. What is more, compared with two segments, three segments of η^3 curve are more suitable to generate path for the reason that more points may make the path more flexible and easier to change the direction of path. The most important is that Bezier curve outperforms η^3 curve and Ferguson curve no matter on simple or complex problems and it improves the solutions further, which is more likely depending on the property of its flexible shape changed by adjusting a fixed number of anchor points. So when PSO and its improved versions are used to solve path planning problems, Bezier curve possesses a higher status. For the limitation of experimental conditions, only the path length and security criteria are compared in this paper. Bezier curve will be used to evaluate more criteria in path planning problems as well as in the condition of dynamic environment where the obstacles are changed with time in the future.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is supported partially by National Natural Science Foundation of China (61305080, U1304602), Postdoctoral Science Foundation of China (Grant 20100480859), Specialized Research Fund for the Doctoral Program of Higher Education (20114101110005), Scientific and Technological Project of Henan Province (132102210521, 122300410264), and Key Foundation of Henan Educational Committee (14A410001).

References

- [1] T. Zhao and J. Fang, "A transit path planning model based on the heterogeneous road network," in *Proceedings of the 17th International Conference on Geoinformatics*, Fairfax, Va, USA, August 2009.
- [2] G. Chesi, "Designing image trajectories in the presence of uncertain data for robust visual servoing path-planning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 1492–1497, Kobe, Japan, May 2009.
- [3] L. Guangshun and S. Hongbo, "Study of technology on path planning for mobile robots," in *Proceedings of the Chinese Control and Decision Conference (CCDC '08)*, pp. 3295–3300, Yantai, China, July 2008.
- [4] Y.-H. Qu, Q. Pan, and J.-G. Yan, "Flight path planning of UAV based on heuristically search and genetic algorithms," in *Proceedings of the 31st Annual Conference of IEEE Industrial Electronics Society*, pp. 6–10, November 2005.
- [5] J.-J. Liang, H. Song, B.-Y. Qu, and X.-B. Mao, "Path planning based on dynamic multi-swarm particle swarm optimizer with crossover," in *Intelligent Computing Theories and Applications*, vol. 7390 of *Lecture Notes in Computer Science*, pp. 159–166, Springer, Berlin, Germany, 2012.
- [6] J.-H. Wei and J.-S. Liu, "Collision-free composite η^3 -splines generation for nonholonomic mobile robots by parallel variable-length genetic algorithm," in *Proceedings of the International Conference on Computational Intelligence for Modelling Control and Automation (CIMCA '08)*, pp. 545–550, Vienna, Austria, December 2008.
- [7] W. Xianxiang, M. Yan, and W. Juan, "An improved path planning approach based on particle swarm optimization," in *Proceedings of the 11th International Conference on Hybrid Intelligent Systems (HIS '11)*, pp. 157–161, December 2011.
- [8] M. Saska, M. Macaš, L. Přeucil, and L. Lhotská, "Robot path planning using particle swarm optimization of ferguson splines," in *Proceedings of the IEEE Conference on Emerging Technologies and Factory Automation (ETFA '06)*, pp. 833–839, September 2006.
- [9] Y. Gao, S.-D. Sun, and D.-F. He, "Global path planning of mobile robot base on particle filter," in *Proceedings of the WRI World Congress on Computer Science and Information Engineering (CSIE '09)*, pp. 205–209, Los Angeles, Calif, USA, April 2009.
- [10] J.-W. Choi, R. Curry, and G. Elkaim, "Path planning based on Bézier curve for autonomous ground vehicles," in *Proceedings of the Advances in Electrical and Electronics Engineering—IAENG Special Edition of the World Congress on Engineering and Computer Science*, pp. 158–166, San Francisco, Calif, USA, October 2008.
- [11] L. Yang, Z. Luo, Z. Tang, and W. Lv, "Path planning algorithm for mobile robot obstacle avoidance adopting Bezier curve based on genetic algorithm," in *Proceedings of the Chinese Control and*

- Decision Conference (CCDC '08)*, pp. 3286–3289, Yantai, China, July 2008.
- [12] D. W. Zhu and X. B. Mao, “Path planning algorithm based on improved particle swarm optimization of Bezier curves,” *Application Research of Computers*, vol. 5, pp. 1–6, 2012.
- [13] E. Rimon and D. E. Koditschek, “Exact robot navigation using artificial potential functions,” *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, 1992.
- [14] N. G. Bourbakis, D. Goldman, R. Fematt, I. Vlachavas, and L. H. Tsoukalas, “Path planning in a 2-D known space using neural networks and skeletonization,” in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 3, pp. 2001–2005, Orlando, Fla, USA, October 1997.
- [15] S. Koenig and M. Likhachev, “Improved fast replanning for robot navigation in unknown terrain,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 968–975, May 2002.
- [16] K. Sugihara and J. Smith, “Genetic algorithms for adaptive motion planning of an autonomous mobile robot,” in *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '97)*, pp. 138–143, Monterey, Calif, USA, July 1997.
- [17] I. Al-Taharwa, A. Sheta, and M. Al-Weshah, “A mobile robot path planning using genetic algorithm in static environment,” *Journal of Computer Science*, vol. 4, no. 4, pp. 341–344, 2008.
- [18] H. Mo and Z. Li, “Bio-geography based differential evolution for robot path planning,” in *Proceedings of the IEEE International Conference on Information and Automation (ICIA '12)*, pp. 1–6, Shenyang, China, June 2012.
- [19] J. Chakraborty, A. Konar, U. K. Chakraborty, and L. C. Jain, “Distributed cooperative multi-robot path planning using differential evolution,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '08)*, pp. 718–725, June 2008.
- [20] R. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory,” in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, pp. 39–43, Nagoya, Japan, October 1995.
- [21] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Piscataway, NJ, USA, December 1995.
- [22] Z. A. Bashir and M. E. El-Hawary, “Short-term load forecasting using artificial neural network based on particle swarm optimization algorithm,” in *Proceedings of the Canadian Conference on Electrical and Computer Engineering (CCECD '07)*, pp. 272–275, Vancouver, Canada, April 2007.
- [23] Y. Fu, M. Ding, and C. Zhou, “Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for UAV,” *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 42, no. 2, pp. 511–526, 2012.
- [24] H. E. Espitia and J. I. Sofrony, “Path planning of mobile robots using potential fields and swarms of Brownian particles,” in *Proceedings of the IEEE Congress of Evolutionary Computation (CEC '11)*, pp. 123–129, New Orleans, La, USA, June 2011.
- [25] M. Ding, W. Sun, and H. Chen, “Multi-working modes product-color planning based on evolutionary algorithms and swarm intelligence,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 871301, 15 pages, 2010.
- [26] T.-S. Zhan and C.-C. Kao, “Modified PSO method for robust control of 3RPS parallel manipulators,” *Mathematical Problems in Engineering*, vol. 2010, Article ID 302430, 25 pages, 2010.
- [27] J. J. Liang and P. N. Suganthan, “Dynamic multi-swarm particle swarm optimizer with a novel constraint-handling mechanism,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 9–16, Vancouver, Canada, July 2006.
- [28] J. J. Liang, H. Song, and B. Y. Qu, “Performance evaluation of dynamic multi-swarm particle swarm optimizer with different constraint handling methods on path planning problems,” in *Proceedings of the IEEE Workshop on Memetic Computing*, pp. 65–71, April 2013.
- [29] R. Wagner, O. Birbach, and U. Frese, “Rapid development of manifold-based graph optimization systems for multi-sensor calibration and SLAM,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '11)*, pp. 3305–3312, San Francisco, Calif, USA, September 2011.