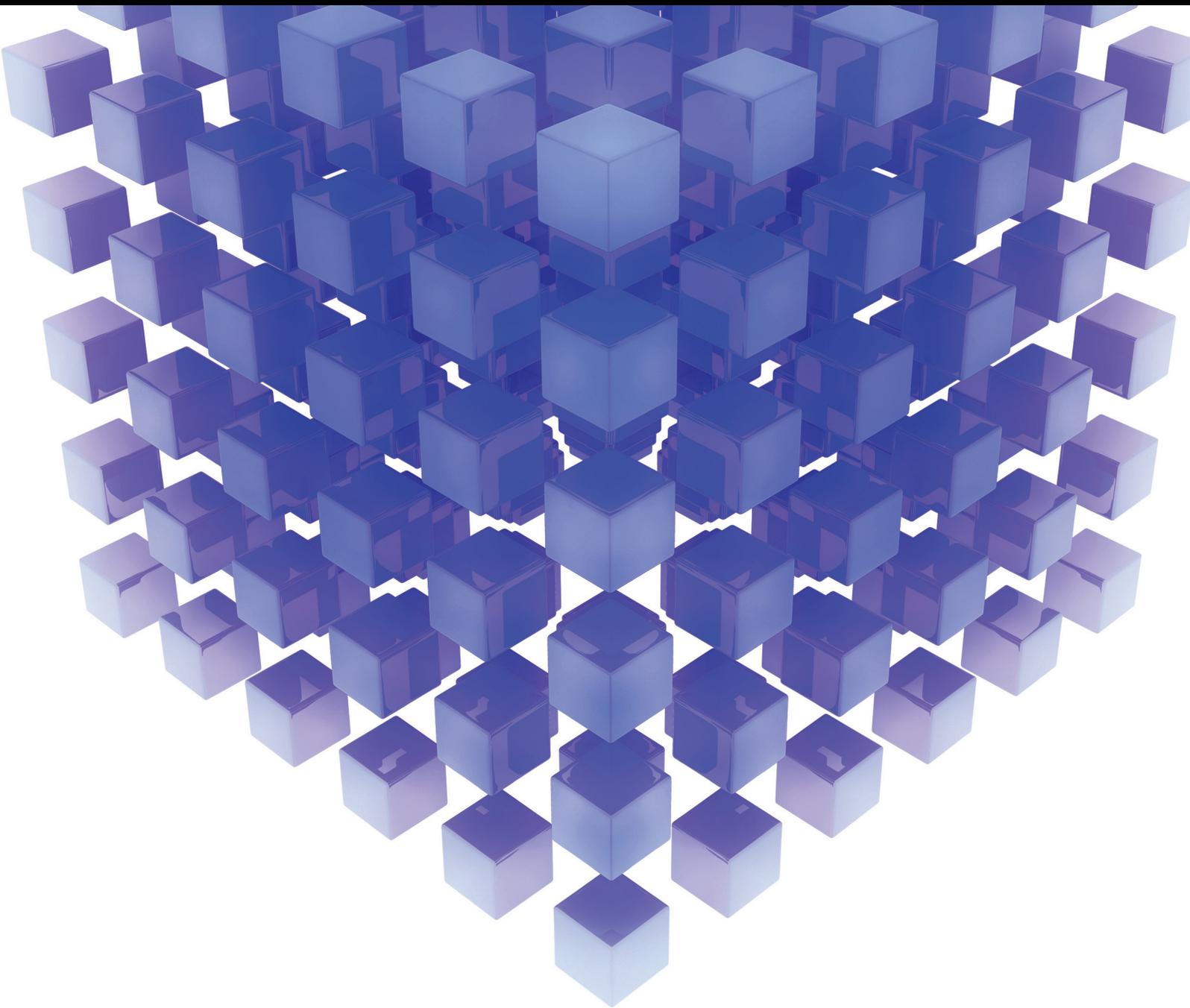


Mathematical Problems in Engineering

Artificial Intelligence and Its Applications 2014

Guest Editors: Yudong Zhang, Saeed Balochian, Praveen Agarwal, Vishal Bhatnagar, and Orwa Jaber Housheya





Artificial Intelligence and Its Applications 2014

Mathematical Problems in Engineering

Artificial Intelligence and Its Applications 2014

Guest Editors: Yudong Zhang, Saeed Balochian,
Praveen Agarwal, Vishal Bhatnagar, and Orwa Jaber Housheya



Copyright © 2016 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
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
Mohammad D. Aliyu, Canada
Juan A. Almendral, Spain
Lionel Amodeo, France
Sebastian Anita, Romania
Renata Archetti, Italy
Felice Arena, Italy
Sabri Arik, Turkey
Fumihiko Ashida, Japan
Hassan Askari, Canada
Mohsen Asle Zaeem, USA
Romain Aubry, 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
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 Buldu', Spain
Tito Busani, USA
Raquel Caballero-Águila, Spain
Pierfrancesco Cacciola, UK
Salvatore Caddemi, Italy
Jose E. Capilla, Spain
Riccardo Caponetto, Italy
Ana Carpio, Spain
Inmaculada T. Castro, Spain
Gabriele Cazzulani, Italy
Miguel E. Cerrolaza, Spain
M. Chadli, France
Gregory Chagnon, France
Ching-Ter Chang, Taiwan
Michael J. Chappell, UK
Kacem Chehdi, France
Peter N. Cheimets, USA
Chunlin Chen, China
Xinkai Chen, Japan
Francisco Chicano, Spain
Hung-Yuan Chung, Taiwan
Joaquim Ciurana, Spain
John D. Clayton, USA
Jean-Pierre Corriou, France
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
Manuel Doblaré, Spain
- Alexandre B. Dolgui, France
Florent Duchaine, France
George S. Dulikravich, USA
Bogdan Dumitrescu, Romania
Horst Ecker, Austria
Ahmed El Hajjaji, France
Fouad Erchiqui, Canada
Anders Eriksson, Sweden
Giovanni Falsone, Italy
Hua Fan, China
Yann Favennec, France
Giuseppe Fedele, Italy
Roberto Fedele, Italy
Jacques Ferland, Canada
Jose R. Fernandez, Spain
Eric Feulvarch, France
Simme Douwe Flapper, Netherlands
Thierry Floquet, France
Eric Florentin, France
Francesco Franco, Italy
Tomonari Furukawa, USA
Mohamed Gadala, Canada
Matteo Gaeta, Italy
Zoran Gajic, USA
Ugo Galvanetto, Italy
Akemi Gálvez, Spain
Rita Gamberini, Italy
Maria Gandarias, Spain
Arman Ganji, Canada
Xin-Lin Gao, USA
Zhong-Ke Gao, China
Giovanni Garcea, Italy
Fernando García, Spain
Laura Gardini, Italy
Alessandro Gasparetto, Italy
Vincenzo Gattulli, Italy
Oleg V. Gendelman, Israel
Mergen H. Ghayesh, Australia
Anna M. Gil-Lafuente, Spain
Hector Gómez, Spain
Francisco Gordillo, Spain
Rama S. R. Gorla, USA
Oded Gottlieb, Israel
Kannan Govindan, Denmark
Antoine Grall, France

Jason Gu, Canada
Quang Phuc Ha, Australia
Masoud Hajarian, Iran
Frédéric Hamelin, France
Zhen-Lai Han, China
Thomas Hanne, Switzerland
Takashi Hasuike, Japan
Xiao-Qiao He, China
M.I. Herreros, Spain
Vincent Hilaire, France
Eckhard Hitzer, Japan
Jaromir Horacek, Czech Republic
Muneo Hori, Japan
András Horváth, Italy
Gordon Huang, Canada
Sajid Hussain, Canada
Asier Ibeas, Spain
Giacomo Innocenti, Italy
Emilio Insfran, Spain
Nazrul Islam, USA
Benoit Iung, France
Payman Jalali, Finland
Reza Jazar, Australia
Khalide Jbilou, France
Linni Jian, China
Bin Jiang, China
Zhongping Jiang, USA
Ningde Jin, China
Grand R. Joldes, Australia
Tadeusz Kaczorek, Poland
Tamas Kalmar-Nagy, Hungary
Tomasz Kapitaniak, Poland
Haranath Kar, India
Konstantinos Karamanos, Belgium
C. Masood Khaliq, South Africa
Do Wan Kim, Republic of Korea
Nam-Il Kim, Republic of Korea
Oleg Kirillov, Germany
Manfred Krafczyk, Germany
Frederic Kratz, France
Jurgen Kurths, Germany
Kyandoghere 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, UAE
Peide Liu, China
Peter Liu, Taiwan
Wanquan Liu, Australia
Yan-Jun Liu, China
Alessandro Lo Schiavo, Italy
Jean J. Loiseau, France
Paolo Lonetti, Italy
Luis M. López-Ochoa, Spain
Vassilios C. Loukopoulos, Greece
Valentin Lychagin, Norway
José María Maestre, Spain
Fazal M. Mahomed, South Africa
Noureddine Manamanni, France
Didier Maquin, France
Paolo Maria Mariano, Italy
Benoit Marx, France
Georges 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
Xesús Nogueira, Spain
Ben T. Nohara, Japan
Mohammed Nouari, France
Mustapha Nourelfath, Canada
Sotiris K. Ntouyas, Greece
Roger Ohayon, France
Mitsuhiro Okayasu, Japan
Javier Ortega-Garcia, Spain
Alejandro Ortega-Moñux, Spain
Naohisa Otsuka, Japan
Erika Ottaviano, Italy
Alkis S. Paipetis, Greece
Alessandro Palmeri, UK
Anna Pandolfi, Italy
Elena Panteley, France
Manuel Pastor, Spain
Pubudu N. Pathirana, Australia
Francesco Pellicano, Italy
Haipeng Peng, China
Mingshu Peng, China
Zhike Peng, China
Marzio Pennisi, Italy
Matjaz Perc, Slovenia
Francesco Pesavento, Italy
M de Rosário Pinho, Portugal
Antonina Pirrotta, Italy
Vicent Pla, Spain
Javier Plaza, Spain
Jean-Christophe Ponsart, France
Mauro Pontani, Italy
Stanislav Potapenko, Canada
Sergio Preidikman, USA
Christopher Pretty, New Zealand
Carsten Proppe, Germany
Luca Pugi, Italy
Yuming Qin, China
Dane Quinn, USA
Jose Ragot, France
Kumbakonam Ramamani Rajagopal, USA
Gianluca Ranzi, Australia
Sivaguru Ravindran, USA
Alessandro Reali, Italy
Oscar Reinoso, Spain
Nidhal Rezg, France
Ricardo Riaza, Spain
Gerasimos Rigatos, Greece
José Rodellar, Spain
Rosana Rodriguez-Lopez, Spain

Ignacio Rojas, Spain
Carla Roque, Portugal
Debasish Roy, India
Rubén Ruiz García, Spain
Antonio Ruiz-Cortes, Spain
Ivan D. Rukhlenko, Australia
Mazen Saad, France
Kishin Sadarangani, Spain
Mehrdad Saif, Canada
Miguel A. Salido, Spain
Roque J. Salterén, Spain
Francisco J. Salvador, Spain
Alessandro Salvini, Italy
Maura Sandri, Italy
Miguel A. F. Sanjuan, Spain
Juan F. San-Juan, Spain
Roberta Santoro, Italy
Ilmar Ferreira Santos, Denmark
José A. Sanz-Herrera, Spain
Nickolas S. Sapidis, Greece
Evangelos J. Sapountzakis, Greece
Andrey V. Savkin, Australia
Valery Sbitnev, Russia
Thomas Schuster, Germany
Mohammed Seaid, UK
Lotfi Senhadji, France
Joan Serra-Sagrasta, Spain
Leonid Shaikhet, Ukraine
Hassan M. Shanechi, USA
Bo Shen, Germany
Babak Shotorban, USA
Zhan Shu, UK

Dan Simon, Greece
Luciano Simoni, Italy
Christos H. Skiadas, Greece
Michael Small, Australia
Francesco Soldovieri, Italy
Raffaele Solimene, Italy
Ruben Specogna, Italy
Sri Sridharan, USA
Ivanka Stamova, USA
Yakov Strelniker, Israel
Sergey A. Suslov, Australia
Thomas Svensson, Sweden
Andrzej Swierniak, Poland
Yang Tang, Germany
Sergio Teggi, Italy
Alexander Timokha, Norway
Gisella Tomasini, Italy
Francesco Tornabene, Italy
Antonio Tornambe, Italy
Fernando Torres, Spain
Sébastien Tremblay, Canada
Irina N. Trendafilova, UK
George Tsiatas, Greece
Antonios Tsourdos, UK
Vladimir Turetsky, Israel
Mustafa Tutar, Spain
Efstratios Tzirtzilakis, Greece
Filippo Ubertini, Italy
Francesco Ubertini, Italy
Hassan Ugail, UK
Giuseppe Vairo, Italy
Kuppalapalle Vajravelu, USA

Robertt A. Valente, Portugal
Pandian Vasant, Malaysia
Miguel E. Vázquez-Méndez, Spain
Josep Vehi, Spain
Kalyana C. Veluvolu, Republic of Korea
Fons J. Verbeek, Netherlands
Franck J. Vernerey, USA
Georgios Veronis, USA
Anna Vila, Spain
Rafael Villanueva, Spain
Uchechukwu E. Vincent, UK
Mirko Viroli, Italy
Michael Vynnycky, Sweden
Shuming Wang, China
Yan-Wu Wang, China
Yongqi Wang, Germany
Desheng D. Wu, Canada
Yuqiang Wu, China
Guangming Xie, China
Xuejun Xie, China
Gen Qi Xu, China
Hang Xu, China
Xing-Gang Yan, UK
Luis J. Yebra, Spain
Peng-Yeng Yin, Taiwan
Ibrahim Zeid, USA
Huaguang Zhang, China
Qingling Zhang, China
Jian Guo Zhou, UK
Quanxin Zhu, China
Mustapha Zidi, France

Contents

Artificial Intelligence and Its Applications 2014

Yudong Zhang, Saeed Balochian, Praveen Agarwal, Vishal Bhatnagar, and Orwa Jaber Housheya
Volume 2016, Article ID 3871575, 6 pages

User Adapted Motor-Imaginary Brain-Computer Interface by means of EEG Channel Selection Based on Estimation of Distributed Algorithms

Aitzol Astigarraga, Andoni Arruti, Javier Muguerza, Roberto Santana, Jose I. Martin, and Basilio Sierra
Volume 2016, Article ID 1435321, 12 pages

A Fault-Tolerant Filtering Algorithm for SINS/DVL/MCP Integrated Navigation System

Xiaosu Xu, Peijuan Li, and Jian-juan Liu
Volume 2015, Article ID 581909, 12 pages

Manifold Learning with Self-Organizing Mapping for Feature Extraction of Nonlinear Faults in Rotating Machinery

Lin Liang, Fei Liu, Maolin Li, and Guanghua Xu
Volume 2015, Article ID 873905, 11 pages

Automatic Classification of Remote Sensing Images Using Multiple Classifier Systems

Bin Yang, Chunxiang Cao, Ying Xing, and Xiaowen Li
Volume 2015, Article ID 954086, 10 pages

Locating High-Impedance Fault Section in Electric Power Systems Using Wavelet Transform, k -Means, Genetic Algorithms, and Support Vector Machine

Ying-Yi Hong and Wei-Shun Huang
Volume 2015, Article ID 823720, 9 pages

Fault Diagnosis of Supervision and Homogenization Distance Based on Local Linear Embedding Algorithm

Guangbin Wang, Jun Luo, Yilin He, and Qinyi Chen
Volume 2015, Article ID 981598, 8 pages

A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications

Yudong Zhang, Shuihua Wang, and Genlin Ji
Volume 2015, Article ID 931256, 38 pages

Recognition of Mixture Control Chart Pattern Using Multiclass Support Vector Machine and Genetic Algorithm Based on Statistical and Shape Features

Min Zhang and Wenming Cheng
Volume 2015, Article ID 382395, 10 pages

A Novel WLAN Client Puzzle against DoS Attack Based on Pattern Matching

Ali Ordi, Mazdak Zamani, Norbik Bashah Idris, Azizah Abdul Manaf, and Mohd Shahidan Abdullah
Volume 2015, Article ID 707548, 12 pages

Dynamic Track Management in MHT for Pedestrian Tracking Using Laser Range Finder

Abdul Hadi Abd Rahman, Hairi Zamzuri, Saiful Amri Mazlan, Mohd Azizi Abdul Rahman, Yoshio Yamamoto, and Saiful Bahri Samsuri
Volume 2015, Article ID 545204, 9 pages

KD-ACP: A Software Framework for Social Computing in Emergency Management

Bin Chen, Laobing Zhang, Gang Guo, and Xiaogang Qiu
Volume 2015, Article ID 915429, 27 pages

Weight Optimization in Recurrent Neural Networks with Hybrid Metaheuristic Cuckoo Search Techniques for Data Classification

Nazri Mohd Nawawi, Abdullah Khan, M. Z. Rehman, Haruna Chiroma, and Tutut Herawan
Volume 2015, Article ID 868375, 12 pages

Model of Multilayer Knowledge Diffusion for Competence Development in an Organization

Przemysław Różewski and Jarosław Jankowski
Volume 2015, Article ID 529256, 20 pages

Optimization of Train Trip Package Operation Scheme

Lu Tong, Lei Nie, Zhenhuan He, and Huiling Fu
Volume 2015, Article ID 472591, 8 pages

Modifying Regeneration Mutation and Hybridising Clonal Selection for Evolutionary Algorithms Based Timetabling Tool

Thatchai Thepphakorn, Pupong Pongcharoen, and Chris Hicks
Volume 2015, Article ID 841748, 16 pages

Color Image Encryption Algorithm Based on TD-ERCS System and Wavelet Neural Network

Kun Zhang and Jian-bo Fang
Volume 2015, Article ID 501054, 10 pages

A Study on Many-Objective Optimization Using the Kriging-Surrogate-Based Evolutionary Algorithm Maximizing Expected Hypervolume Improvement

Chang Luo, Koji Shimoyama, and Shigeru Obayashi
Volume 2015, Article ID 162712, 15 pages

Neural Network-Based Fault-Tolerant Control of Underactuated Surface Vessels

Bong Seok Park
Volume 2015, Article ID 903759, 9 pages

Multiagent Cooperative Learning Strategies for Pursuit-Evasion Games

Jong Yih Kuo, Hsiang-Fu Yu, Kevin Fong-Rey Liu, and Fang-Wen Lee
Volume 2015, Article ID 964871, 13 pages

Stereo Matching Based on Immune Neural Network in Abdomen Reconstruction

Huan Liu, Kuangrong Hao, Yongsheng Ding, and Chunjuan Ouyang
Volume 2015, Article ID 242794, 15 pages

Landslide Occurrence Prediction Using Trainable Cascade Forward Network and Multilayer Perceptron

Mohammad Subhi Al-batah, Mutasem Sh. Alkhasawneh, Lea Tien Tay,
Umi Kalthum Ngah, Habibah Hj Lateh, and Nor Ashidi Mat Isa
Volume 2015, Article ID 512158, 9 pages

System Optimization for Temporal Correlated Cognitive Radar with EBPSK-Based MCPC Signal

Peng Chen and Lenan Wu
Volume 2015, Article ID 302083, 10 pages

A New Asymptotic Notation: Weak Theta

Andrei-Horia Mogoş, Bianca Mogoş, and Adina Magda Florea
Volume 2015, Article ID 912962, 15 pages

A Novel Tournament Selection Based Differential Evolution Variant for Continuous Optimization Problems

Qamar Abbas, Jamil Ahmad, and Hajira Jabeen
Volume 2015, Article ID 205709, 21 pages

Fuzzy Wavelet Neural Network Using a Correntropy Criterion for Nonlinear System Identification

Leandro L. S. Linhares, Aluisio I. R. Fontes, Allan M. Martins, Fábio M. U. Araújo, and Luiz F. Q. Silveira
Volume 2015, Article ID 678965, 12 pages

A Self-Adaptive Hidden Markov Model for Emotion Classification in Chinese Microblogs

Li Liu, Dashi Luo, Ming Liu, Jun Zhong, Ye Wei, and Letian Sun
Volume 2015, Article ID 987189, 8 pages

Editorial

Artificial Intelligence and Its Applications 2014

**Yudong Zhang,¹ Saeed Balochian,² Praveen Agarwal,³ Vishal Bhatnagar,⁴
and Orwa Jaber Housheya⁵**

¹*School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China*

²*Department of Electrical Engineering, Islamic Azad University, Gonabad Branch, Gonabad 96916-29, Iran*

³*Department of Mathematics, Anand International College of Engineering, Jaipur 303012, India*

⁴*Ambedkar Institute of Advanced Communication Technologies and Research, Government of NCT of Delhi, Geeta Colony, Delhi 110031, India*

⁵*Department of Chemistry, Arab American University, Jenin, State of Palestine*

Correspondence should be addressed to Yudong Zhang; zhangyudong@njnu.edu.cn

Received 22 February 2016; Accepted 23 February 2016

Copyright © 2016 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.

Artificial intelligence (AI) researches the intelligence exhibited by machines. It creates revolutionized information technology. The world famous companies like Google, Yahoo, Facebook, Baidu, and so forth have spent millions of dollars to research on developing new algorithms on AI. Nevertheless, there are a number of challenging issues in realistic applications due to fast-growing large and complex problems.

This special issue aims to bring together academia and industry experts to report on the recent developments on artificial intelligence and its applications, in every aspect of artificial intelligence technology, including machine learning, data mining, computer vision, multiagent systems, evolutionary computation, deep learning, and fuzzy logic. The primary guideline was to either demonstrate the most significant developments on the topics of AI or apply AI-related algorithms in real-life scenarios.

In the paper entitled “A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications,” Y. Zhang et al. investigated the particle swarm optimization (PSO). Their survey presented a comprehensive investigation of PSO. On one hand, they provided advances with PSO, including its modifications (including quantum-behaved PSO, bare-bones PSO, chaotic PSO, and fuzzy PSO), population topology (as fully connected, von Neumann, ring, star, random, etc.), hybridization (with genetic algorithm, simulated annealing, Tabu search, artificial immune system, ant colony algorithm, artificial bee colony, differential

evolution, harmonic search, and biogeography-based optimization), extensions (to multiobjective, constrained, discrete, and binary optimization), theoretical analysis (parameter selection and tuning and convergence analysis), and parallel implementation (in multicore, multiprocessor, GPU, and cloud computing forms). On the other hand, they offered a survey on applications of PSO to the following eight fields: electrical and electronic engineering, automation control systems, communication theory, operations research, mechanical engineering, fuel and energy, medicine, chemistry, and biology. It is hoped that this survey would be beneficial for the researchers studying PSO algorithms.

In the paper entitled “Model of Multilayer Knowledge Diffusion for Competence Development in an Organization,” P. Rózewski and J. Jankowski proposed several models with the main goal of simulating diffusion and explaining the nature of knowledge diffusion. They extended existing approaches by using multilayer diffusion model and focused on analysis of competence development process. The proposed model described competence development process in a new way through horizontal and vertical knowledge diffusion in multilayer network. In the network, agents collaborated and interchanged various kinds of knowledge through different layers and these mutual activities affected the competencies in a positive or negative way. Taking into consideration worker's cognitive and social abilities and the previous level of competence, the new competence level

can be estimated. Their model was developed to support competence management in different organizations.

In the paper entitled "Landslide Occurrence Prediction Using Trainable Cascade Forward Network and Multilayer Perceptron," M. S. Al-batah et al. introduced two models of artificial neural network, namely, multilayer perceptron (MLP) and cascade forward neural network (CFNN), to predict the landslide hazard map of Penang Island. These two models were tested and compared using eleven machine-learning algorithms: Levenberg Marquardt, Broyden Fletcher Goldfarb, resilient backpropagation, scaled conjugate gradient, conjugate gradient with Beale, conjugate gradient with Fletcher Reeves updates, conjugate gradient with Polakribiere updates, one step secant, gradient descent, gradient descent with momentum and adaptive learning rate, and gradient descent with momentum algorithm. The performance of the landslide prediction depends on the input factors beside the prediction method. In this research work, 14 input factors were used. The prediction accuracies of networks were verified using the area under curve for the receiver operating characteristics. The results indicated that the best prediction accuracy was achieved using the CFNN network with the Levenberg Marquardt learning algorithm, that is, 82.89% for the training data set and 81.62% for the testing data set.

In the paper entitled "Optimization of Train Trip Package Operation Scheme," L. Tong et al. firstly analyzed the related factors of train trip package transportation from its organizational forms and characteristics. Then an optimization model for train trip package transportation was established to provide optimum operation schemes. The proposed model was solved by the genetic algorithm. At last, the paper tested the model on the basis of the data of 8 regions. The results showed that the proposed method is feasible for solving operation scheme issues of train trip package.

In the paper entitled "A Self-Adaptive Hidden Markov Model for Emotion Classification in Chinese Microblogs," L. Liu et al. proposed a modified version of hidden Markov model (HMM) classifier, called self-adaptive HMM, the parameters of which were optimized by particle swarm optimization algorithms. Since manually labeling large-scale dataset was difficult, Liu et al. also employed the entropy to decide whether a new unlabeled tweet will be contained in the training dataset after being assigned an emotion using the proposed HMM-based approach. In the experiment, about 200,000 Chinese tweets from Sina Weibo were collected. The results showed that the *F*-score of the approach gets 76% on happiness and fear and 65% on anger, surprise, and sadness. In addition, the self-adaptive HMM classifier outperformed Naive Bayes and support vector machine on recognition of happiness, anger, and sadness.

In the paper entitled "Automatic Classification of Remote Sensing Images Using Multiple Classifier Systems," B. Yang et al. aimed to correctly identify land use types reflected in remote sensing images. Support vector machine, maximum likelihood classifier, backpropagation neural network, fuzzy *c*-means, and minimum distance classifier were combined to construct three multiple classifier systems (MCSs). Two MCSs were implemented, namely, comparative major voting

(CMV) and Bayesian average (BA). One method called WA-AHP was proposed, which introduced analytic hierarchy process into MCS. Classification results of base classifiers and MCSs were compared with the ground-truth map. Accuracy indicators were computed and receiver operating characteristic curves were illustrated, so as to evaluate the performance of MCSs. Experimental results showed that employing MCSs can increase classification accuracy significantly, compared with base classifiers. From the accuracy evaluation result and visual check, the best MCS is WA-AHP with overall accuracy of 94.2%, which overmatches BA and rivals CMV. The producer's accuracy of each land use type proves the good performance of WA-AHP. Therefore, MCS is superior to base classifiers in remote sensing image classification, and WA-AHP is an efficient MCS.

In the paper entitled "A Novel Tournament Selection Based Differential Evolution Variant for Continuous Optimization Problems," Q. Abbas et al. proposed a novel tournament based parent selection variant of differential evolution (DE) algorithm. The proposed variant enhanced searching capability and improved convergence speed of DE algorithm. This paper also presented a novel statistical comparison of existing DE mutation variants, which categorized these variants in terms of their overall performance. Experimental results showed that the proposed DE variant had significance performance over other DE mutation variants.

In the paper entitled "Manifold Learning with Self-Organizing Mapping for Feature Extraction of Nonlinear Faults in Rotating Machinery," L. Liang et al. proposed a new method for extracting the low-dimensional feature automatically with self-organization mapping manifold for the detection of rotating mechanical nonlinear faults (such as rubbing and pedestal looseness). Under the phase space reconstructed by single vibration signal, the self-organization mapping (SOM) with expectation maximization iteration algorithm was used to divide the local neighborhoods adaptively without manual intervention. After that, the local tangent-space alignment algorithm was adopted to compress the high-dimensional phase space into low-dimensional feature space. The proposed method takes advantages of the manifold learning in low-dimensional feature extraction and adaptive neighborhood construction of SOM and can extract intrinsic fault features of interest in two-dimensional projection space. To evaluate the performance of the proposed method, the Lorenz system was simulated and rotation machinery with nonlinear faults was obtained for test purposes. Compared with the holospectrum approaches, the results reveal that the proposed method is superior in identifying faults and effective for rotating machinery condition monitoring.

In the paper entitled "Modifying Regeneration Mutation and Hybridising Clonal Selection for Evolutionary Algorithms Based Timetabling Tool," T. Thepphakorn et al. outlined the development of a new evolutionary algorithms based timetabling (EAT) tool for solving course scheduling problems that include a genetic algorithm (GA) and a memetic algorithm (MA). Reproduction processes may generate infeasible solutions. Previous research used

repair processes that were applied after a population of chromosomes was generated. This research developed a new approach which (i) modified the genetic operators to prevent the creation of infeasible solutions before chromosomes were added to the population and (ii) included the clonal selection algorithm (CSA) and the elitist strategy (ES) to improve the quality of the solutions produced. This approach was adopted by both the GA and MA within the EAT. The MA was further modified to include hill climbing local search. The EAT program was tested using 14 benchmark timetabling problems from the literature using a sequential experimental design, which included a fractional factorial screening experiment. Experiments were conducted to (i) test the performance of the proposed modified algorithms, (ii) identify which factors and interactions were statistically significant, (iii) identify appropriate parameters for the GA and MA, and (iv) compare the performances of the various hybrid algorithms. The genetic algorithm with modified genetic operators produced an average improvement of over 50%.

In the paper entitled “Fuzzy Wavelet Neural Network Using a Correntropy Criterion for Nonlinear System Identification,” L. L. S. Linhares et al. investigated the fuzzy wavelet neural networks (FWNNs) that are an efficient tool to identify nonlinear systems. In these structures, features related to fuzzy logic, wavelet functions, and neural networks are combined in an architecture similar to the adaptive neurofuzzy inference systems (ANFIS). In practical applications, the experimental data set used in the identification task often contains unknown noise and outliers, which decrease the FWNN model reliability. In order to reduce the negative effects of these erroneous measurements, their work proposed the direct use of a similarity measure based on information theory in the FWNN learning procedure. The mean squared error (MSE) cost function was replaced by the maximum correntropy criterion (MCC) in the traditional error backpropagation (BP) algorithm. The input-output maps of a real nonlinear system studied were identified from an experimental data set corrupted by different outliers rates and additive white Gaussian noise. The results demonstrated the advantages of the proposed cost function using the MCC as compared to the MSE. This work also investigated the influence of the kernel size on the performance of the MCC in the BP algorithm, since it is the only free parameter of correntropy.

In the paper entitled “A Novel WLAN Client Puzzle against DoS Attack Based on Pattern Matching,” A. Ordi et al. analyzed the popularity of 802.11 based networks and pointed out that they suffered several types of DoS attack, launched by an attacker whose aim is to make an access point (AP) unavailable to legitimate users. One of the most common DoS attacks on 802.11 based networks is to deplete the resources of the AP. A serious situation like this can occur when the AP receives a burst of connection requests. Their paper addressed this common DoS attack and proposed a lightweight puzzle, based on pattern matching. Using a pattern-matching technique, this model adequately resisted resource-depletion attacks in terms of both puzzle generation and solution verification. Using a sensible series of contextual

comparisons, the outcomes were modelled by a simulator, and the security definition and proofs are verified, among other results.

In the paper entitled “Weight Optimization in Recurrent Neural Networks with Hybrid Metaheuristic Cuckoo Search Techniques for Data Classification,” N. M. Nawi et al. investigated recurrent neural network (RNN). This network can be educated with gradient descent backpropagation. However, traditional training algorithms had some drawbacks such as slow speed of convergence being not definite to find the global minimum of the error function since gradient descent may get stuck in local minima. As a solution, nature inspired metaheuristic algorithms provide derivative-free solution to optimize complex problems. This paper proposed a new metaheuristic search algorithm called Cuckoo Search (CS) based on Cuckoo bird's behavior to train Elman recurrent network (ERN) and backpropagation Elman recurrent network (BPERN) in achieving fast convergence rate and to avoid local minima problem. The proposed CSERN and CSBPERN algorithms were compared with artificial bee colony using BP algorithm and other hybrid variants algorithms. Specifically, some selected benchmark classification problems were used. The simulation results showed that the computational efficiency of ERN and BPERN training process was highly enhanced when coupled with the proposed hybrid method.

In the paper entitled “Stereo Matching Based on Immune Neural Network in Abdomen Reconstruction,” H. Liu et al. suggested stereo feature matching is a technique that finds an optimal match in two images from the same entity in the three-dimensional world. The stereo correspondence problem is formulated as an optimization task where an energy function, which represents the constraints on the solution, is to be minimized. They proposed a novel intelligent biological network (Bio-Net), which involves the human B-T cells immune system into neural network, in order to learn the robust relationship between the input feature points and the output matched points. In the experiments, the abdomen reconstructions for different-shape mannequins were performed by means of the proposed method. The final results were compared and analyzed, which demonstrate that the proposed approach greatly outperforms the single neural network and the conventional matching algorithm in precise. Particularly, with respect to time cost and efficiency, the proposed method exhibits its significant promising potential for improvement. Hence, it is entirely considered as an effective and feasible alternative option for stereo matching.

In the paper entitled “Neural Network-Based Fault-Tolerant Control of Underactuated Surface Vessels,” B. S. Park addressed the problem of trajectory tracking of underactuated surface vessels (USVs) in the presence of thruster failure. Multilayer neural networks (MNNs) were employed to estimate the unknown model parameters and external disturbances. To design a fault-tolerant controller without a fault detection scheme, they used the Nussbaum gain technique. They also introduced an additional control to resolve the difficulty arising from having fewer inputs than degrees of freedom. Further, an approach angle was proposed to track both straight

and curved path. Stability analysis and simulations were performed to demonstrate the effectiveness of the proposed scheme.

In the paper entitled “Dynamic Track Management in MHT for Pedestrian Tracking Using Laser Range Finder,” A. H. A. Rahman et al. proposed a multilevel clustering of laser range finder (LRF) data to improve the accuracy of a tracking system by adding another clustering level after the feature extraction process. A dynamic track management (DTM) was introduced in multiple hypothesis tracking (MHT) with multiple motion models to perform a track creation, association, and deletion. The experimental results from real time implementation proved that the proposed multiclustering is capable of producing a better performance with less computational complexity for a track management process. The proposed dynamic track management is able to solve the tracking problem with lower computation time when dealing with occlusion, crossed track, and track deletion.

In the paper entitled “A Study on Many-Objective Optimization Using the Kriging-Surrogate-Based Evolutionary Algorithm Maximizing Expected Hypervolume Improvement,” C. Luo et al. investigated and compared the many-objective optimization performance of the Kriging-surrogate-based evolutionary algorithm (EA), which maximizes expected hypervolume improvement (EHVI) for updating the Kriging model, with those using expected improvement (EI) and estimation (EST) updating criteria. Numerical experiments were conducted in 3- to 15-objective DTLZ1-7 problems. An exact hypervolume calculating algorithm was used for the problems with less than six objectives. On the other hand, an approximate hypervolume calculating algorithm based on Monte Carlo sampling was adopted for the problems with more objectives. The results indicate that, in the nonconstrained case, EHVI is a highly competitive updating criterion for the Kriging model and EA based many-objective optimization, especially when the test problem is complex and the number of objectives or design variables is large.

In the paper entitled “System Optimization for Temporal Correlated Cognitive Radar with EBPSK-Based MCPC Signal,” P. Chen and L. Wu proposed a novel radar working scheme to consider both target detection and estimation. At the detection stage, the generalized likelihood ratio test (GLRT) threshold was deduced, and the GLRT detection probability was given. At the estimation stage, an approach based on Kalman filtering (KF) was proposed to estimate target scattering coefficients (TSC), and the estimation performance was improved significantly by exploiting the TSC temporal correlation. Additionally, the optimal waveform was obtained to minimize the mean squared error (MSE) of KF estimation. For the practical consideration, iteration algorithms were proposed to optimize the EBPSK-based multicarrier phase-coded (MCPC) signal in terms of power allocation and coding matrix. Simulation results demonstrated that the KF estimation approach can improve the estimation performance by 25% compared with maximum a posteriori probability (MAP) method, and the KF estimation performance can be further improved by 90% by optimizing the transmitted waveform spectrum.

Moreover, by optimizing the power allocation and coding matrix of the EBPSK-based MCPC signal, the KF estimation performances were, respectively, improved by 7% and 8%.

In the paper entitled “Color Image Encryption Algorithm Based on TD-ERCS System and Wavelet Neural Network,” K. Zhang and J. Fang proposed a new image encryption algorithm based on TD-ERCS system and wavelet neural network, in order to solve the security problem of transmission image across public networks. According to the permutation process and the binary XOR operation from the chaotic series by producing TD-ERCS system and wavelet neural network, it can achieve image encryption. This encryption algorithm was a reversible algorithm, and it can achieve original image in the rule inverse process of encryption algorithm. Finally, through computer simulation, the experiment results showed that the new chaotic encryption algorithm based on TD-ERCS system and wavelet neural network is valid and has higher security.

In the paper entitled “Fault Diagnosis of Supervision and Homogenization Distance Based on Local Linear Embedding Algorithm,” G. Wang et al. developed an improved local linear embedding algorithm of homogenization distance (HLL), in view of the problems of uneven distribution of reality fault samples and dimension reduction effect of locally linear embedding (LLE) algorithm, which is easily affected by neighboring points. The method made the overall distribution of sample points tend to be homogenization and reduces the influence of neighboring points using homogenization distance instead of the traditional Euclidean distance. It was helpful to choose effective neighboring points to construct weight matrix for dimension reduction. Because the fault recognition performance improvement of HLL was limited and unstable, the paper further proposed a new local linear embedding algorithm of supervision and homogenization distance (SHLLE) by adding the supervised learning mechanism. On the basis of homogenization distance, supervised learning increased the category information of sample points, so that the same category of sample points will be gathered and the heterogeneous category of sample points will be scattered. It effectively improved the performance of fault diagnosis and maintains stability at the same time. A comparison of the methods mentioned above was made by simulation experiment with rotor system fault diagnosis, and the results showed that SHLLE algorithm has superior fault recognition performance.

In the paper entitled “A New Asymptotic Notation: Weak Theta,” A.-H. Mogoş et al. defined a new asymptotic notation, called “Weak Theta,” that used the comparison of various complexity functions with two given complexity functions. Weak Theta notation was especially useful in characterizing complexity functions whose behavior was hard to be approximated using a single complexity function. In addition, in order to highlight the main particularities of Weak Theta, they proposed and proved several theoretical results: properties of Weak Theta, criteria for comparing two complexity functions, and properties of a new set of complexity functions based on Weak Theta. Furthermore, to illustrate the usefulness of this notation,

the authors discussed an application of Weak Theta in artificial intelligence.

In the paper entitled “Locating High-Impedance Fault Section in Electric Power Systems Using Wavelet Transform, k -Means, Genetic Algorithms, and Support Vector Machine,” Y.-Y. Hong and W.-S. Huang presented a new method for locating the line (feeder) section of the high-impedance fault (HIF) with the help of limited measurements in electric power systems. The discrete wavelet transform was used to extract the features of transients caused by HIFs. A modified k -means algorithm associated with genetic algorithms was then utilized to determine the placement of measurement facilities. The signal energies attained by wavelet coefficients served as inputs to the support vector machine for locating the HIF line section. The simulation results obtained from an 18-busbar distribution system showed the applicability of the proposed method.

In the paper entitled “Recognition of Mixture Control Chart Pattern Using Multiclass Support Vector Machine and Genetic Algorithm Based on Statistical and Shape Features,” M. Zhang and W. Cheng introduce an intelligent hybrid model for recognizing the mixture control chart patterns (CCPs) that included three main aspects: feature extraction, classifier, and parameters optimization. In the feature extraction, statistical and shape features of observation data were used in the data input to get the effective data for the classifier. A multiclass support vector machine (MSVM) was applied for recognizing the mixture CCPs. Finally, genetic algorithm (GA) was utilized to optimize the MSVM classifier by searching the best values of the parameters of MSVM and kernel function. The performance of the hybrid approach was evaluated by simulation experiments, and simulation results demonstrated that the proposed approach was able to effectively recognize mixture CCPs.

In the paper entitled “A Fault-Tolerant Filtering Algorithm for SINS/DVL/MCP Integrated Navigation System,” X. Xu et al. proposed a fault-tolerant adaptive Kalman filter (FTAKF) algorithm for the integrated navigation system composed of a strapdown inertial navigation system (SINS), a Doppler velocity log (DVL), and a magnetic compass (MCP). The evolutionary artificial neural networks (EANN) were used in self-learning and training of the intelligent data fusion algorithm. The proposed algorithm can significantly outperform the traditional KF in providing estimation continuously with higher accuracy and smoothing the KF outputs when observation data were inaccurate or unavailable for a short period. The experiments of the prototype verified the effectiveness of the proposed method.

In the paper entitled “Multiagent Cooperative Learning Strategies for Pursuit-Evasion Games,” J. Y. Kuo et al. examined the pursuit-evasion problem for coordinating multiple robotic pursuers to locate and track a nonadversarial mobile evader in a dynamic environment. Two kinds of pursuit strategies were proposed, one for agents that cooperate with each other and the other for agents that operate independently. This work further employed the probabilistic theory to analyze the uncertain state information about the pursuers and the evaders and uses case-based reasoning to

equip agents with memories and learning abilities. According to the concepts of assimilation and accommodation, both positive-angle and bevel-angle strategies were developed to assist agents in adapting to their environment effectively. The case study analysis used the recursive porous agent simulation toolkit (REPAST) to implement a multiagent system and demonstrated superior performance of the proposed approaches to the pursuit-evasion game.

In the paper entitled “User Adapted Motor-Imaginary Brain-Computer Interface by means of EEG Channel Selection Based on Estimation of Distributed Algorithms,” A. Astigarraga et al. presented a personalized interface design method, for electroencephalogram (EEG) based Brain-Computer Interfaces (BCIs), based on channel selection. They described a novel two-step method in which firstly a computationally inexpensive greedy algorithm found an adequate search range; and then, an estimation of distribution algorithm (EDA) was applied in the reduced range to obtain the optimal channel subset. The use of the EDA algorithm allowed us to select the most interacting channels subset, removing the irrelevant and noisy ones, thus selecting the most discriminative subset of channels for each user improving accuracy. The method was tested on the IIIa dataset from the BCI competition III. Experimental results showed that the resulting channel subset was consistent with motor imaginary related neurophysiological principles and, on the other hand, optimized performance reducing the number of channels.

In the paper entitled “KD-ACP: A Software Framework for Social Computing in Emergency Management,” B. Chen et al. addressed the application of a computational theory and related techniques for studying emergency management in social computing. They proposed a novel software framework called KD-ACP. The framework provided a systematic and automatic platform for scientists to study the emergency management problems in three aspects: modelling the society in emergency scenario as the artificial society; investigating the emergency management problems by the repeat computational experiments; parallel execution between artificial society and the actual society managed by the decisions from computational experiments. The software framework was composed of a series of tools. These tools were categorized into three parts corresponding to “A,” “C,” and “P,” respectively. Using H1N1 epidemic in Beijing city as the case study, the modelling and data generating of Beijing city, experiments with settings of H1N1, and intervention measures and parallel execution by situation tool were implemented by KD-ACP. The results output by the software framework showed that the emergency response decisions can be tested to find a more optimal one through the computational experiments. In the end, the advantages of the KD-ACP and the future work were summarized in the conclusion.

Acknowledgments

We would like to show our appreciation to all authors for their contributions and the reviewers for their effort providing

valuable and prompt comments. We hope that this special issue offers a comprehensive and timely view of the advances in artificial intelligence and its applications, and we expect that it will offer stimulation for further research.

Yudong Zhang
Saeed Balochian
Praveen Agarwal
Vishal Bhatnagar
Orwa Jaber Housheya

Research Article

User Adapted Motor-Imaginary Brain-Computer Interface by means of EEG Channel Selection Based on Estimation of Distributed Algorithms

Aitzol Astigarraga,¹ Andoni Arruti,² Javier Muguerza,² Roberto Santana,¹ Jose I. Martin,² and Basilio Sierra¹

¹Department of Computer Science and Artificial Intelligence, University of the Basque Country UPV/EHU, Computer Science Faculty, 20018 Donostia-San Sebastian, Spain

²Department of Computer Architecture and Technology, University of the Basque Country UPV/EHU, Computer Science Faculty, 20018 Donostia-San Sebastian, Spain

Correspondence should be addressed to Andoni Arruti; andoni.arruti@ehu.es

Received 4 June 2014; Accepted 19 November 2014

Academic Editor: Yudong Zhang

Copyright © 2016 Aitzol Astigarraga 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.

Brain-Computer Interfaces (BCIs) have become a research field with interesting applications, and it can be inferred from published papers that different persons activate different parts of the brain to perform the same action. This paper presents a personalized interface design method, for electroencephalogram- (EEG-) based BCIs, based on channel selection. We describe a novel two-step method in which firstly a computationally inexpensive greedy algorithm finds an adequate search range; and, then, an Estimation of Distribution Algorithm (EDA) is applied in the reduced range to obtain the optimal channel subset. The use of the EDA allows us to select the most interacting channels subset, removing the irrelevant and noisy ones, thus selecting the most discriminative subset of channels for each user improving accuracy. The method is tested on the IIIa dataset from the BCI competition III. Experimental results show that the resulting channel subset is consistent with motor-imaginary-related neurophysiological principles and, on the other hand, optimizes performance reducing the number of channels.

1. Introduction

Recent advances in Cognitive Neuroscience and Brain Imaging technologies provide us with the ability to interact directly with the human brain, offering thus an alternative to natural communication and control. In recent years, many Brain-Computer Interfaces (BCIs) have been built [1].

The aim of a BCI system is to establish a communication method that translates human intentions, mental tasks reflected by suitable brain signals (e.g., electric, chemical, and blood flow changes), into a control signal for an output device such as a computer application or a neuroprosthesis, not requiring any muscular response. The idea is to provide a new communication method to people who are paralysed but are cognitively intact. Motor-Imaginary BCI systems are based on the fact that imagination of movement changes brain

activity in the cortex. Therefore, the recognition of patterns associated with certain movements could be used to generate control signals.

Two general BCI systems can be found in literature: invasive technologies, in which sensors are implanted directly in the brain, and noninvasive technologies, which measure brain activity using external sensors. In most BCI applications, brain signals are measured by electroencephalography (EEG) sensors, recording electrical activity from the scalp with electrodes. They are inexpensive, portable and their temporal resolution is very good.

However, there are several challenges. For example, when developing new BCI systems, the selection of specific channels to be used for each subject is a crucial aspect. However, finding optimal channel number and their positions is still a challenging task [2]. In such cases, the most common way

is to use a large number of channels for signal classification. But applying a large number of channels is time consuming and may include noisy and redundant signals that degrade the BCI system performance.

Therefore, selecting the least number of channels that yield the best or required accuracy is crucial. Channel selection can be performed manually [3], based on neurophysiological knowledge, but this approach does not always guarantee optimal results [4]. Moreover, automated channel selection algorithms can provide optimal channel positions without any prior knowledge about the task at hand. Two types of automatic channel selection approaches can be found in literature: subject-independent methods, in which the electrode subset is shared between all subjects, and subject-dependent methods, in which the channel subset is customized for each subject in order to improve the individual BCI performances. This paper focuses on subject-dependent channel selection, and its main goal is the automatic selection of a reduced number of channels (and consequently their location) adapted to each subject, maintaining, or even improving, the classification accuracy in EEG-based BCI.

Our proposed method decomposes the channel selection problem into two steps: estimation of a suitable channel range by means of a backward greedy algorithm and optimal channel selection applying an evolutionary algorithm. In the first step, we propose to use a computationally inexpensive greedy algorithm to estimate a suitable channel range—this is made in a similar way as in [5]. In the second step, the optimal channels are selected applying an evolutionary algorithm. In particular, we investigate the adequateness of an Estimation of Distribution Algorithm (EDA) [6], an evolutionary algorithm based on probabilistic modelling of the search space, to exploit the channel's relationship and pick out the subset that yields the best performance. The ability of Estimation of Distribution Algorithms to maintain the quality of the joint probability distribution among the selected features is remarked by Yin and Wu [7]. In order to evaluate the performance of the proposed approach, public data available from BCI competition has been used. The main goal of this paper is to adapt BCI-EEG interfaces to each individual by selecting those channels which better discriminate among the different actions to be performed.

The paper is structured as follows: Section 2 discusses the related work. In Section 3 the two-step channel selection method (greedy algorithm + EDA) is described. Section 4 shows the experimental setup, the data acquisition procedure, and the characteristics of the EDA implementation. Experimental results are given in Section 5 and Section 6 presents the conclusions and future work.

2. Related Work

Several channel selection methods have been proposed in the literature [8–10]. An updated overview can be found in [2, 4]. Most relevant methods include common spatial patterns (CSP) [11], Support Vector Machines (SVM) [12], and methods based on the mutual information (MI).

The classical backward greedy strategy has been also applied in many research works [8, 13]. In this method the

subset is built iteratively. It starts with a full set of electrodes and, at each step, the electrode that maximizes the accuracy of the complementary subset is removed.

More closely related to our approach, previous researches have addressed the channel selection problem using evolutionary algorithms [4, 9, 14–17]. However, the application of EDAs has been constrained to the analysis of Magnetoencephalography (MEG) data in the context of multiobjective optimization [18], an approach with important differences with the one introduced in this paper. Nevertheless, we briefly review this related work to emphasize differences and similarities with our work.

Wei et al. [17, 19] apply genetic algorithms (GAs) for the analysis of multichannel electrocorticogram (ECoG) recordings in a ECoG-based BCI. They combine the application of the CSP method for feature extraction, Fisher discriminant analysis as classification method, with the use of a GA for feature selection. The authors acknowledge the ability of the GA-based approach to reduce the number of features without losing classification accuracy.

In [9], a GA is combined with a multilayer neural network (ML-NN) to find a subset of channels that maximize the ML-NN's classification accuracy. They use EEG and ECoG recordings. A particular characteristic of this work is that channel selection is not directly accomplished by the GA by optimizing the accuracy. Instead, the fitness function used in the GA is the training error when an MLP is trained for a limited number of epochs. Channels are selected a posteriori by analysing the best solutions of the GA, which includes a nonautomatic part that cannot be analysed.

In [14], a multiobjective evolutionary algorithm (EA) is applied to channel selection. The idea of simultaneously optimizing two or more objectives, including the selection of an optimal set of channels, is very promising for classification of brain recordings. However, a difficulty is that the multiobjective search can be much more complex than single-objective optimization, and this difficulty rapidly increases with the number of objectives involved. Also in [18], multiobjective EAs are applied for channel selection in MEG recordings. In this case, each objective corresponds to maximizing the classification accuracy for a different individual. A regularized logistic regression method is used as a classifier, and a posteriori analysis of the best solutions obtained serves to estimate the relevance of each channel. There is not a sound experiment concerning the real contribution of the evolutionary algorithm.

Kee et al. [4] propose the combination of a Bayesian linear discriminant analysis classifier with a GA for automatic channel selection for a P300 BCI. The authors recognize that a “GA exhibits great potential to study the correlation and the joint effect of various channel combination as the fitness value is not biased to the performance of individual channel.” However, the limitations of classical EAs like GAs to capture and respect the dependencies between interacting variables are not discussed. EDAs are conceived to automatically identify and represent these types of dependencies.

More recently, Bhattacharyya et al. [20] suggest the use of a combination of a Learning Automata (LA) and Differential Evolution (DE) [21] for feature extraction in the analysis of

EEG data in BCI experiments. Here, the LA-DA is not used for channel selection since features are extracted from only two channels (C3 and C4). Instead, it is applied to select the features that are passed to the classifier. The selection of those two channels is based on the authors experience, but the generalization of this approach is not clear.

There are other evolutionary computation algorithms to have in mind [22]. Nevertheless, the benefits of using EDAs that model the interactions between the variables over evolutionary algorithms that do not take correlations into account have been documented in previous work [6, 23].

To summarize, previous works have proved that channel selection by means of evolutionary algorithms can improve classification accuracy across a variety of brain signals, that is, EEG, ECoG, and MEG. However, these research works have been focused on the application of GAs and other standard EAs that do not explicitly capture and represent the dependencies between the channels. As recognized by some of the authors, the potential of the wrapper approach [24] is its capacity for considering the synergies between the features during the classification process. But in order to exploit these synergies, the EAs should be able to identify the most important interacting features and use this information at the time of generating new solutions. This is exactly what EDAs do.

While filter-based methods could in principle be applied for channel selection, these methods cannot cope with intricate and higher order dependencies between the channels, as our wrapper-type EDA approach does. In addition, the analysis of the probabilistic models generated by the EDA can provide information about what the relevant interactions for the problem are. However, in this paper, we use probabilistic models only to make a more efficient generation of solutions and leave the analysis of the potential information captured by the model as a topic of future research.

3. Proposed Approach

Having to personalize BCI-EEG interfaces to each user as main goal, this paper proposes a novel two-step method for personalized optimal channel selection. Firstly, the system selects an adequate range of number of channels to be explored in the second step. This selection is done by applying a simple greedy algorithm. Thereafter, a more computationally expensive EDA is applied to find the final channel selection. The main reason to choose EDA for the second step is its ability to capture the relevant relationships between the variables of the optimization problem, and, thus, the selection of the channels will be more accurate, reducing the complexity of the BCI system.

Both the greedy algorithm and the EDA will work with a population of candidate solutions to the problem. A solution $\mathbf{x} = (x_1, \dots, x_{60})$ will be defined as a 60-tuple of binary 0, 1 values—the so-called Binary Encoding. Each position in the tuple refers to a concrete EEG channel, and the value indicates whether this channel is used (1 value) or not (0 value). Furthermore, the two algorithms require the use of a fitness function to evaluate the quality of the explored solutions. In this work we used the common spatial patterns (CSP)

method for feature extraction from the EEG raw signals, and the accuracy of a Support Vector Machine classifier as fitness function, because of its good performance in comparison with other paradigms [25] (see Section 4.2 for more details).

In the next two subsections we explain the main characteristics of the proposed greedy algorithm and the EDA.

3.1. Greedy Search Approach. For the first step we designed a simple greedy search algorithm to reduce the search space (see Algorithm 1). The aim is to ease the search to the EDA making a rough estimation of the adequate number of channels.

In this method, we start with a solution \mathbf{x}_{60} that uses the full set of electrodes (the 60 elements set to one). At each iteration l , a population D_l of solutions is generated by removing one electrode of the best solution selected in the previous iteration ($60 - l + 1$ possibilities). The solution that gives the best score when evaluated with the fitness function is chosen as the best distribution \mathbf{x}_{60-l} for that number of electrodes.

Simplicity of the greedy algorithm makes a good choice for a rough selection of the range of channels to be used in the more refined search.

3.2. Estimation of Distribution Algorithms. Estimations of distribution algorithms have successfully been developed for combinatorial optimization [6]. They combine statistical learning with population-based search in order to automatically identify and exploit certain structural properties of optimization problems.

EDAs typically work with a population of candidate solutions to the problem, starting with the population generated according to the uniform distribution over all admissible solutions.

The population is then scored using a fitness function. This fitness function gives a numerical ranking for each candidate, with the higher the number the better the solution. From this ranked population, a subset of the most promising solutions are selected by the selection operator. An example selection operator is truncation selection with threshold $\tau = 50\%$, which selects the 50% best solutions. The algorithm then constructs a probabilistic model which attempts to estimate the probability distribution of the selected solutions. Once the model is constructed, new solutions are generated by sampling the distribution encoded by this model. These new solutions are then incorporated back into the old population, possibly replacing it entirely. The process is repeated until some termination criteria are met (usually when a solution of sufficient quality is reached or when the number of iterations reaches some threshold), with each iteration of this procedure usually referred to as one generation of the EDA.

4. Experimental Setup

The performed experiment is explained in this section. First of all, we present the used dataset and how it has been obtained, then details about how the fitness function is

```

(1)  $\mathbf{x}_{60} \leftarrow$  solution with all the components (60) set to one.
(2)  $l = 1$ 
(3) do {
(4)    $D_l \leftarrow$  Generate  $60 - l + 1$  different solutions removing one channel from  $\mathbf{x}_{60-l+1}$ 
(5)   Compute the fitness function for all the solutions in  $D_l$ 
(6)    $\mathbf{x}_{60-l} \leftarrow$  Select the solution with maximum fitness function value
(7)    $l = l + 1$ 
(8) } until  $l = 60$ 

```

ALGORITHM 1: Population-based greedy search.

calculated are given, and, finally, the EDA implementation is commented.

4.1. Data Acquisition. In this work we have selected the IIIa dataset from the BCI competition III [26], because it is publicly available and has been widely used for benchmark evaluation. It contains data from 3 subjects: K3b, K6b, and L1b, collected as follows [27]:

- (1) Each subject, sitting in front of a computer, was asked to perform imaginary movements of the left hand, right hand, tongue, or foot during a specified time interval according to a cue. The order of cues was random.
- (2) 60 electrodes were placed on the subject's scalp (see Figure 1(a)) recording a signal sampled at 250 Hz and filtered between 1 and 50 Hz using a Notch filter.
- (3) As shown in Figure 1(b), each trial started with a blank screen. At $t = 2$ s, a beep was generated and a cross "+" was shown to attract the subject's attention. At $t = 3$ s an arrow pointing to the left, right, up, or down was shown for 1 s and the subject was asked to try one of four imaginary movements until the cross disappeared at $t = 7$ s. This was followed by a 2 s break, and then the next trial began.

The dataset contains 360 instances (cases) for subject K3b, 240 for K6b, and 240 for L1b. Each instance was labelled as belonging to one of the four classes. Each subject contains a balanced distribution of the classes. Two data files are available for each subject: training and testing. The number of instances in the training and testing datasets was equal for all subjects and was 180 for K3b, 120 for K6b, and 120 for L1b. The distribution of the classes was equal in both training and testing data.

The three subjects had different amounts of experience in BCI training. K3b was the most experienced, L1b had less experience, and K6b was a beginner. This has a great influence in classification results (K3b presents the highest accuracy and K6b the lowest) as can be seen in the work of AlZoubi et al. [25] or in [28].

In this work, being our goal the study of the channel selection problem, we will maintain the methodology used in one of these studies [25] as a reference.

4.2. Fitness Function. As said before, both algorithms (greedy and EDA) need to evaluate each individual in the population of possible solutions. For this, a classifier using the corresponding subset of channels is built and its mean accuracy is used as a fitness function. For comparison reasons, we have applied the same methodology used by AlZoubi et al. in [25]. These are the steps followed to compute the fitness function for a given channel selection:

- (i) *Feature Extraction.* Firstly, the CSP method is applied to the channel-reduced raw EEG data. For each class, applying CSP for that class versus the others, a reduced set of 5 projection signals is obtained. Then 3 frequency band filters (for 8–12 Hz, 21–20 Hz, and 20–30 Hz) are applied, and, finally, 7 features are extracted: max, min, and mean voltage values, voltage range, number of samples above zero volts, zero voltage crossing rate, and average signal power. This process gives 420 (4 classes \times 5 projections \times 3 filters \times 7 features) numeric features for each case of the dataset.
- (ii) *Training and Cross-Validation.* In [25] twelve classification paradigms are compared, and for this research we have chosen Support Vector Machines (SVM) [29, 30] because their results are among the best. The value of the fitness function is the mean accuracy obtained with 10-fold cross-validation applied to the training dataset. This has been calculated by the data mining software package WEKA [31] (default parameters are used).

4.3. Characteristics of the EDA Implementation. One distinguished feature of EDAs is the type of probabilistic model they use. They should be able to capture the relevant relationships between the variables of the optimization problem. These relationships are encoded in terms of statistical dependencies. Complex models are able to represent higher order dependencies between the variables but they are also more computationally costly due to the requirements of the learning and sampling procedures. As the probabilistic model of choice for the EDA, we use a tree, a model that exhibits a fine balance between the power of representation and the computational cost.

Let $\mathbf{X} = (X_1, \dots, X_n)$ denote a vector of discrete random variables. We will use $\mathbf{x} = (x_1, \dots, x_n)$ to denote

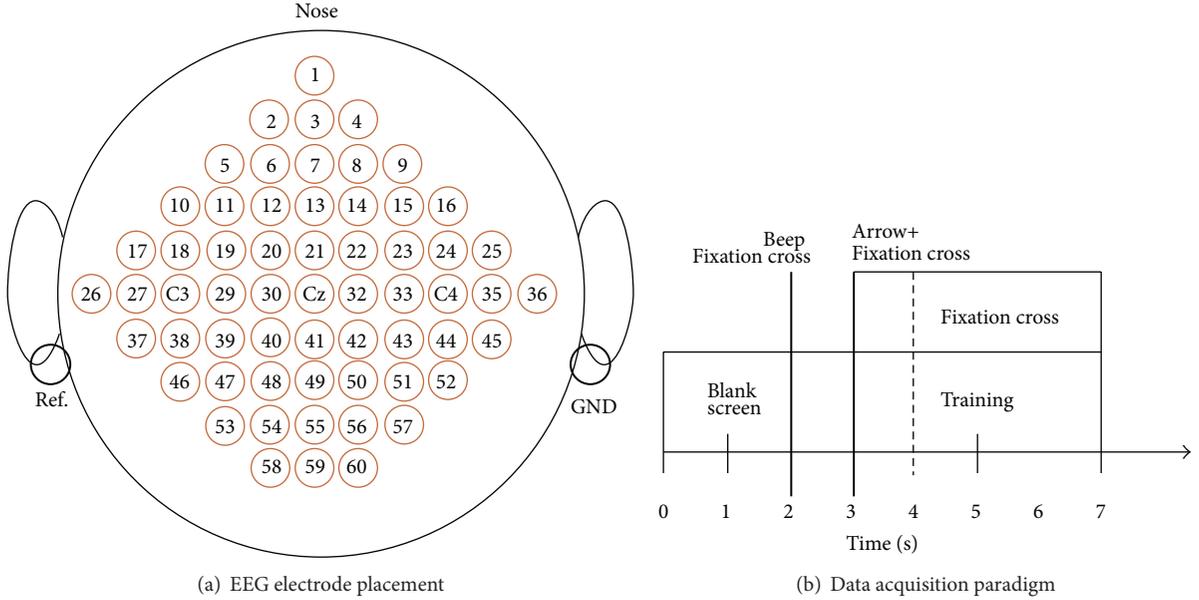


FIGURE 1: Data acquisition schemata (dataset IIIa, BCI competition III).

an assignment to the variables. A probability distribution $p_{\mathcal{T}}(\mathbf{x})$ that is conformal with a tree is defined as $p_{\mathcal{T}}(\mathbf{x}) = \prod_{i=1}^n p(x_i | \text{pa}(x_i))$, where $\text{pa}(x_i)$ is the parent of x_i in the tree, and $p(x_i | \text{pa}(x_i)) = p(x_i)$ when $\text{pa}(x_i) = \emptyset$; that is, x_i is the root of the tree. The distribution $p_{\mathcal{T}}(\mathbf{x})$ itself will be called a tree model when no confusion is possible.

Trees are able to represent bivariate marginal distributions and they can be learned from data with a computational complexity of n^2 . In order to learn the structure from a dataset, the univariate and bivariate probabilities are, respectively, calculated for every variable and pair of variables. Using the marginal probabilities, the mutual information between each pair of variables is computed. The tree structure itself is determined using the Chow-Liu algorithm [32]. It works by computing the maximum weight spanning tree from the matrix of mutual information between pairs of variables. We set a threshold on the minimal mutual information value required to connect two variables. This allows for representing disconnected trees, that is, a forest.

Algorithm 2 shows the different steps of Tree-EDA. The algorithm starts by generating the random population of solutions D_0 . In each iteration l , a set of solutions D_{l-1}^s is selected from population D_{l-1} and from this population the probabilistic model is learned.

The Tree-EDA shown in Algorithm 2 incorporates the method for learning the tree structure. To sample new solutions from a tree, probabilistic logic sampling [33] is applied to sample. In this method, the roots of each tree are initially sampled according to their univariate probabilities. The rest of the variables are sampled following the order determined by the trees structures and using the conditional probability distributions. As a selection method, Tree-EDA uses truncation selection in which the $T = 50\%$ best percentages of the population (highest objective values) are selected. The new sampled solutions are combined with the set of

best solutions (elitist solutions) selected from the previous iteration.

EDAs that used trees were originally introduced in [34] and those that employ forests were introduced in [35]. The scheme of Algorithm 2 corresponds to the algorithm introduced in [36]. It has been implemented in MATLAB using the MATEDA software [37], a highly modular implementation in which each EDA component (either added by the user or already included in the package) is implemented as an independent program.

In this paper we use as stop criterion a fixed number of generations. The Tree-EDA parameters used in our experiments are presented in Section 5.1.

4.4. Limiting the Number of Channels of EDA Population.

EDA is an evolutionary algorithm that can change the number of channels in any direction while trying to optimize the fitness function. In this work, being our goal to reduce the number of channels, we will need to fix an upper limit for number of channels at each experiment. We will achieve this with two actions:

- (i) setting a unitation constraint when generating the first population: we will generate solutions uniformly distributed but with a fix number of channels;
- (ii) repairing the sampled population at each generation: solutions with a number of channels greater than limit are truncated randomly before evaluating them.

5. Experimental Results

In this section we analyse the overall performance of the proposed approach in the channel selection problem. We evaluate the behaviour of the system in terms of the accuracy

```

(1)  $D_0 \leftarrow$  Generate  $M$  solutions randomly
(2)  $l = 1$ 
(3) do {
(4)    $D_{l-1}^s \leftarrow$  Select  $N \leq M$  solutions from  $D_{l-1}$  according to a selection method
(5)   Compute the univariate and bivariate marginal frequencies  $p_i^s(x_i | D_{l-1}^s)$  and  $p_{i,j}^s(x_i, x_j | D_{l-1}^s)$  of  $D_{l-1}^s$ 
(6)   Calculate the matrix of mutual information using bivariate and univariate marginals.
(7)   Calculate the maximum weight spanning tree from the matrix of mutual information.
(8)   Compute the parameters of the model.
(9)    $D_l \leftarrow$  Sample  $M$  solutions (the new population) from the tree and add elitist solutions.
(10)   $l = l + 1$ 
(11) } until A stop criterion is met

```

ALGORITHM 2: Tree-EDA.

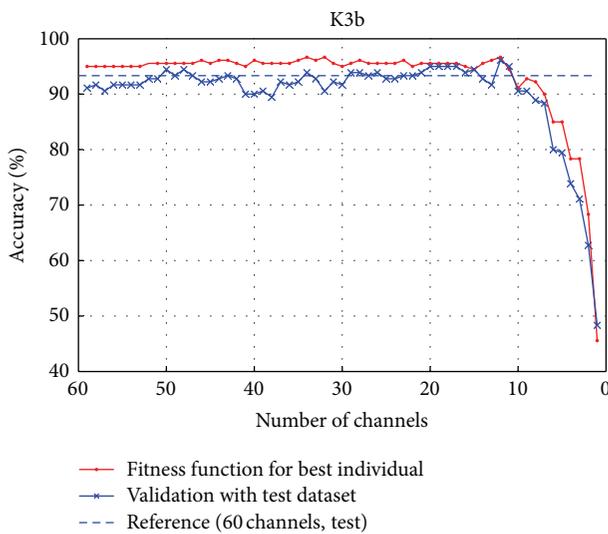


FIGURE 2: Results obtained with greedy search algorithm for subject K3b.

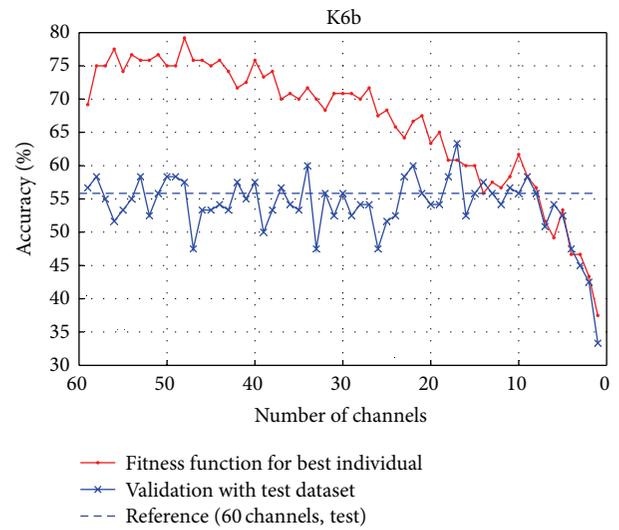


FIGURE 3: Results obtained with greedy search algorithm for subject K6b.

obtained using a standard classification paradigm. Furthermore, we compare the results of the complete system with those obtained by the greedy algorithm.

Obtained results applying the greedy algorithm to the three subjects are shown in Figures 2, 3, and 4. The figures represent the fitness function for the best solution at each iteration, the validation with testing dataset also for the best solution at each iteration; the results obtained with the testing dataset when all channels are used for classification are also shown as baseline reference.

As it can be seen, for the three subjects, the results obtained with 60 electrodes are similar to those obtained after the channel number has been decreased, and the obtained accuracy only decreases when less than 10 electrodes are used.

It can be seen, as said before, that the performance of the classifier depends strongly on the subject, very good for K3b, and poor for the untrained K6b and L1b. Surprisingly, this simple search algorithm shows clearly that, for the three subjects, the number of channels can be reduced drastically without losing accuracy.

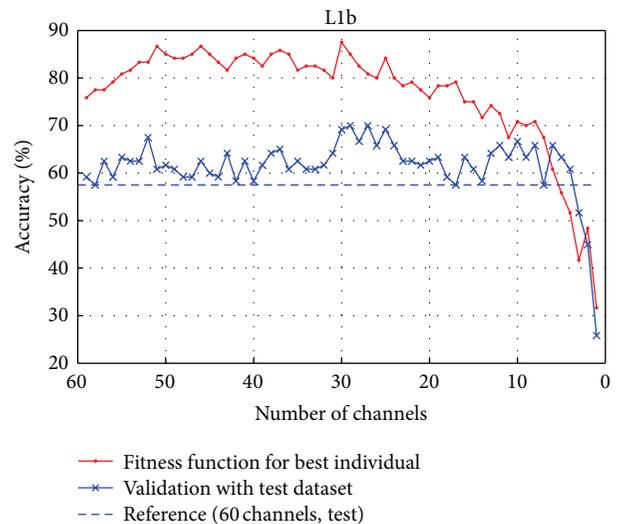


FIGURE 4: Results obtained with greedy search algorithm for subject L1b.

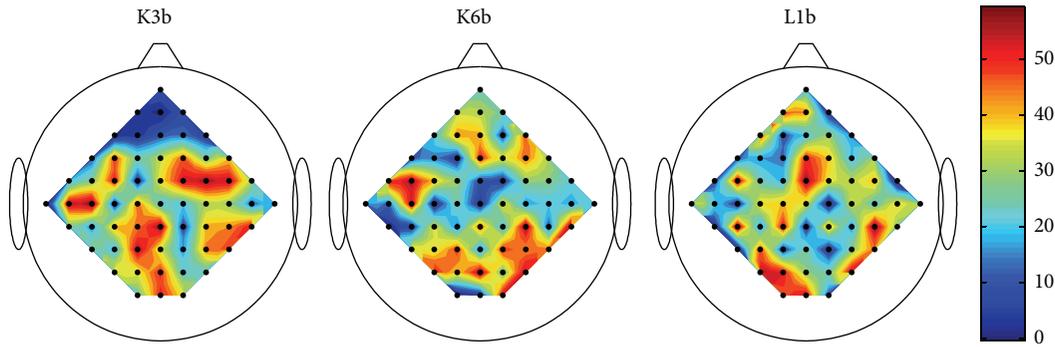


FIGURE 5: Channels position and frequency using greedy search algorithm.

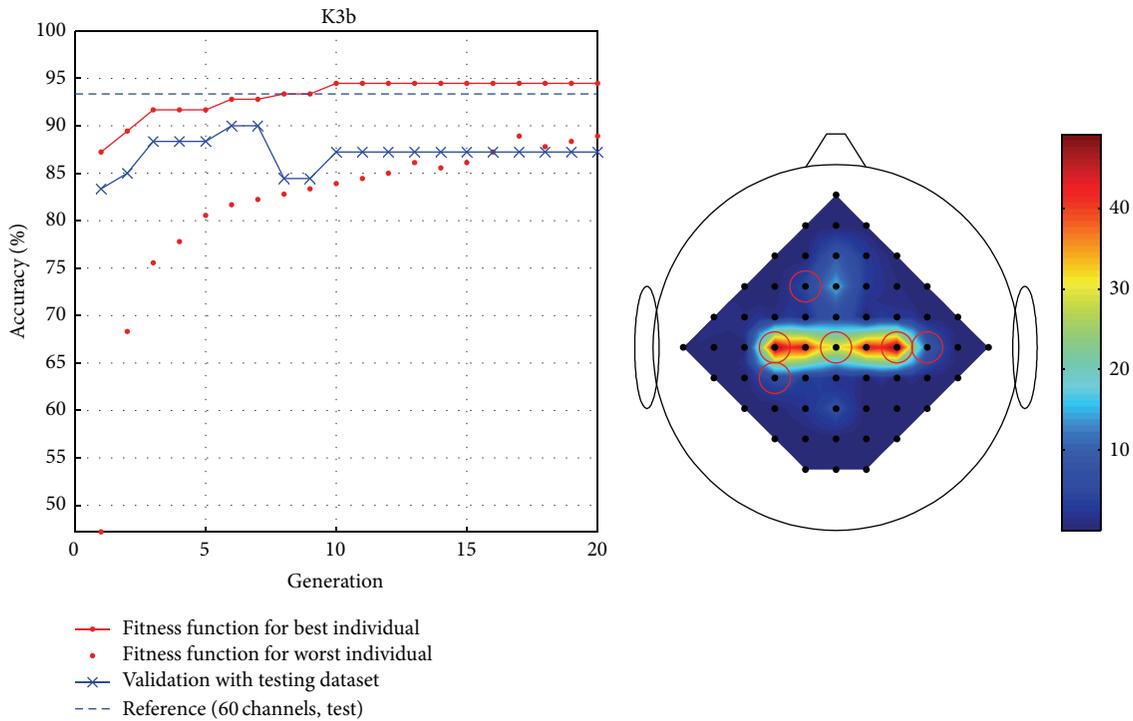


FIGURE 6: Results obtained with EDA search algorithm with a maximum of 6 channels for subject K3b.

In Figure 5 a topographical map for each subject shows the frequency of occurrence of the channels in the 60 best solutions (one for each iteration). The low values correspond to first discarded channels and the highest to the most significant for the classification task. It can be seen that the distribution is subject-dependant, thus justifying the development of an automatic method for searching the optimal subset of channels, making the adaptation of the system to each subject automatic.

5.1. EDA Approach. Seeing the results obtained with the greedy algorithm, we decided to carry out a sequence of experiments with the EDA, changing the top channel number limit in the interval from 16 to 1, wider than the minimum range (from 10 to 1) observed with the greedy algorithm. This

focuses on the work of the EDA to a range where doing a more refined search could be interesting.

With respect to EDA parameters, and based on preliminary experiments, the population size is 200 and the number of generations 20.

To show a sample of the evolution of the EDA, Figures 6, 7, and 8 show intermediate results for the 3 subjects using 6 channels. The figures show the evolution of fitness function. A validation value obtained for the best solution of each generation, with the testing dataset, is also shown, but these values are not used by EDA. A topographic map shows the presence of channels in 50 solutions (the best ten in the last 5 generations). The results show that, for the subject L1b, with only 6 channels the EDA is able to find a solution even better than using all the channels. For subjects K3b and K6b the found solutions are near to the 60-channel based solution.

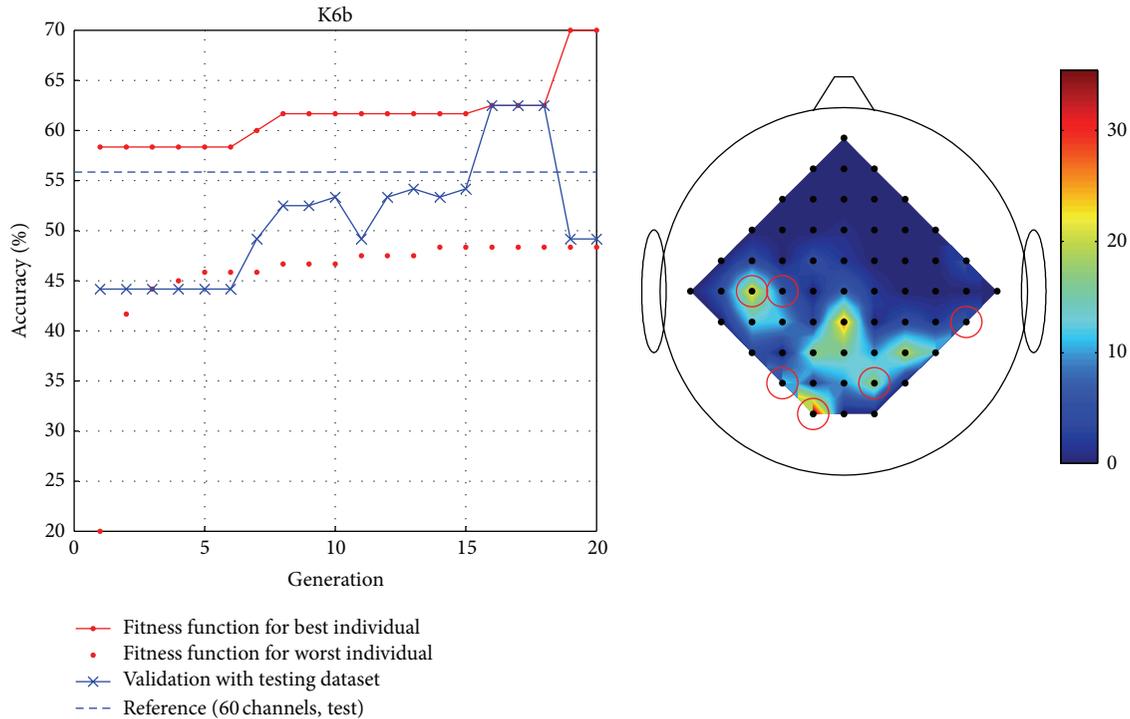


FIGURE 7: Results obtained with EDA search algorithm with a maximum of 6 channels for subject K6b.

Figures 9, 10, and 11 show results obtained with EDA in the last generation with different number of channels for the 3 subjects. Values obtained with the greedy search algorithm are also shown. As it can be seen, the EDA finds a better solution for the subjects K6b and L1b than the solution based on all the channels (with only 4 channels) and finds a similar solution with 10 channels for the K3b subject.

In Figure 12 channel position and frequency as a percentage, in best solutions (the best ten in the last 5 generations of 13 experiments), are shown for the tree subjects. This figure shows again the different distribution of the selected channels for the three subjects, making a specific selection of channels for each subject necessary.

Finally, the results obtained with the EDA have been compared with the results obtained with the greedy algorithm. We applied a Wilcoxon signed ranks test (0.05 confidence level) to determine if statistically significant differences are between the results of both algorithms. The test shows that the differences are statistically significant (p value = 0.037).

Therefore, the combination of the two algorithms in a two-step system is a good option to achieve a simpler user adapted interface maintaining or, even, improving the accuracy of the system.

5.2. Final Discussion. Experimental results show the adequacy of the proposed approach. On the first step, a greedy approach has been used in order to show that channel number reduction is appropriate in a 60-electrode EEG-BCI model. It can be seen in the obtained results that electrode number is decreased to about ten/sixteen channels without losing accuracy for the three subjects, focusing on the search

space in the second step. On the second step, the evolutionary algorithm is used to select a fixed number of electrodes (from 1 to 16). The obtained results of the EDA based approach outperform those achieved by using both the 60 channels and the greedy approach alone as well, with statistically significant differences. In addition, our experiments show that the combination of the two algorithms in a two-step system is a good option to achieve a simpler user adapted interface maintaining or, even, improving the accuracy of the system.

It has to be noticed that, among the different brain zones, selected channels appear to be consistent with MI-related neurophysiological principles [3]. In this sense, the contribution of the performed research can be seen as a step towards a personalized EEG-based BCI interface, in which a person is first trained with a general 60-electrode system, and then the relevant ones are selected to improve the human-machine communication process.

6. Conclusions and Future Works

In this paper a two-step system for channel selection by means of EDA has been presented, aiming at maintaining or even improving the classification accuracy with a few EEG channels automatically selected for each subject. The motivation behind our research work is that electrode signals received by an EEG-BCI interface are not independent among them and that selecting an optimal subset of the electrodes can improve the results when the goal is to identify motor imagery states. In this way, the fine selection

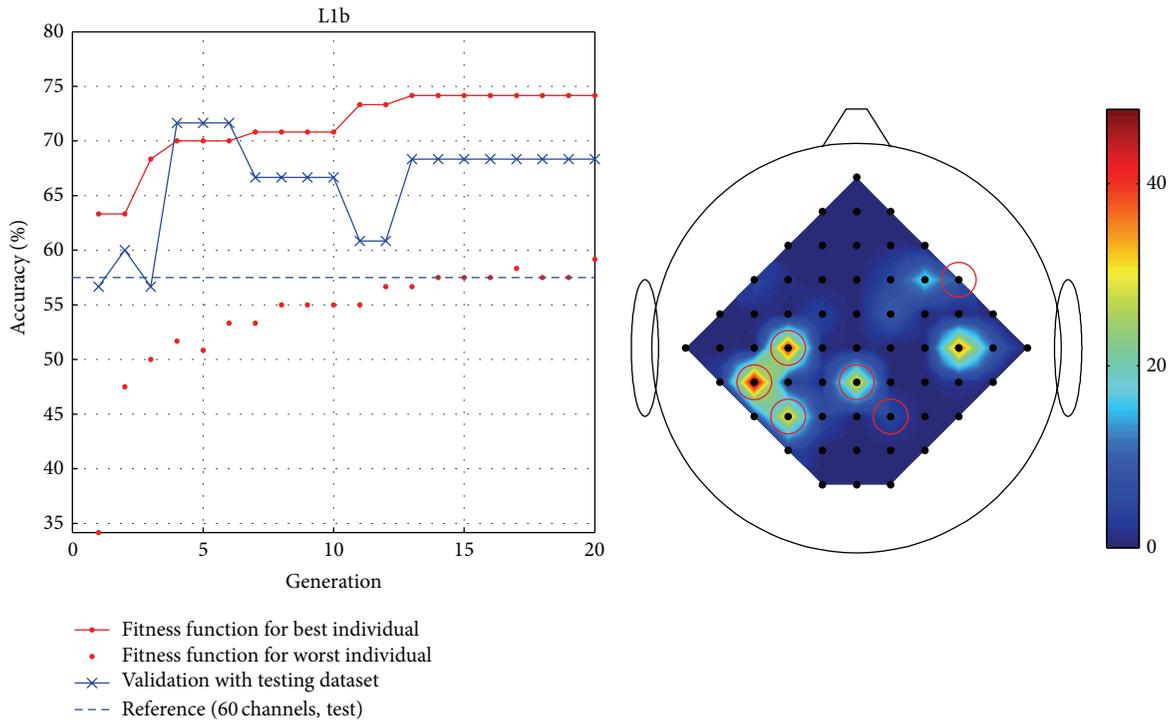


FIGURE 8: Results obtained with EDA search algorithm with a maximum of 6 channels for subject L1b.

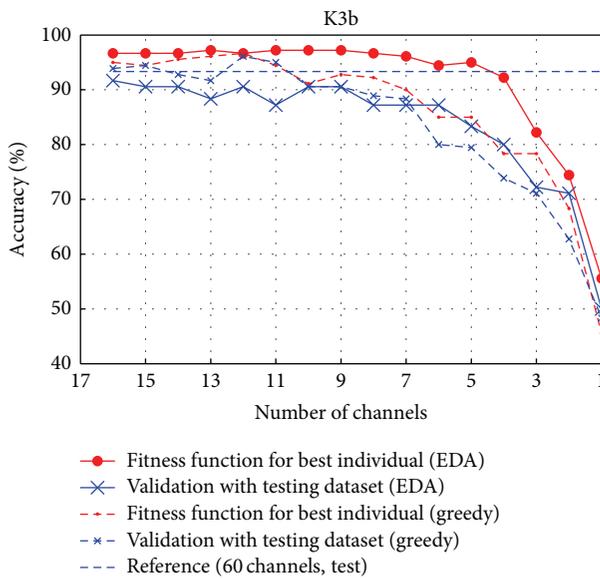


FIGURE 9: Results obtained with EDA search algorithm in the last generation with different number of channels for subject K3b.

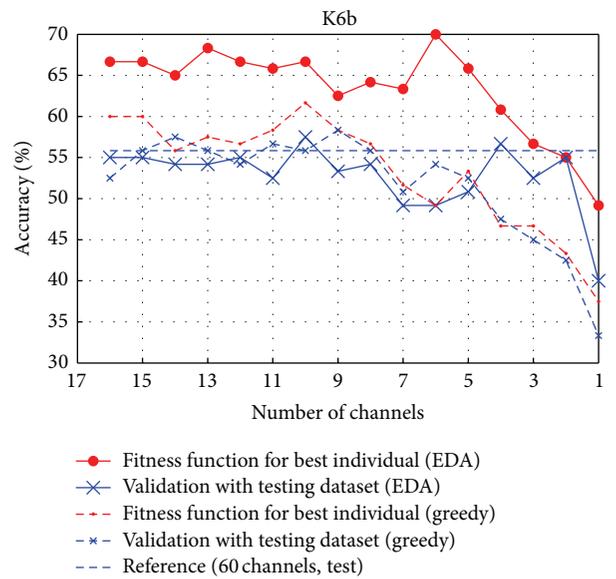


FIGURE 10: Results obtained with EDA search algorithm in the last generation with different number of channels for subject K6b.

is performed using an evolutionary computing paradigm—EDA—which looks for, and maintains, a relationship among the selected input channels, thus reducing or suppressing existing redundancies among the channels. The obtained results show that the combination of the greedy and EDA approaches, in a two-step system, is a good option to achieve a simpler and more accurate personalized system.

As future work, along with a comparison with other methods, we are planning to apply the presented approach taking more base classifiers or multiclassifier systems [38] in the channel selection process. Other joint probability models such as Bayesian Networks could be used to extract the relationships of the channels [39]. Fitness-scaling methods will be explored to improve the EDA searching process [40].

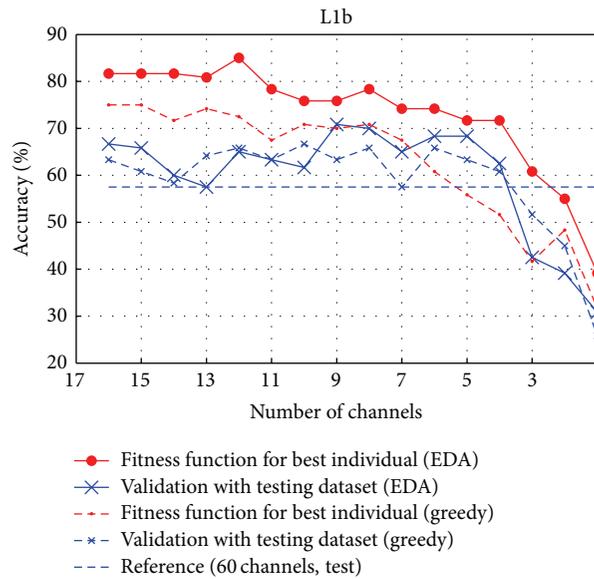


FIGURE 11: Results obtained with EDA search algorithm in the last generation with different number of channels for subject L1b.

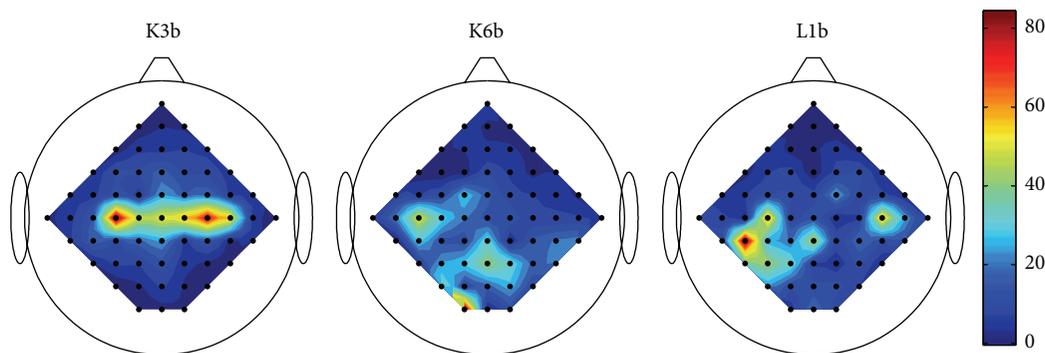


FIGURE 12: Channels position and frequency using EDA search algorithm.

On the other hand, in order to avoid high dimensionality, certain features must be selected before classification starts [23]. A more in-depth analysis of the channel signal is to be done, in order to detect parts of the brain activity that have more influence. A (temporal) clustering among the received signals [41] is also a line for future research which is being considered by the authors. Finally, the inclusion of nonintentional control state patterns—as fifth class—in the experiments could be a first step towards an asynchronous system.

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 partially supported by the Basque Government under Grants IT395-10, IT313-10, and IT609-13, the University of the Basque Country

UPV/EHU under Grant UFI11/45, and the Ministry of Science and Technology of Spain under Grant TIN-41272P.

References

- [1] C. Guger, B. Z. Allison, and G. Edlinger, *Brain-Computer Interface Research: A State-of-the-Art Summary 2*, Springer, New York, NY, USA, 2014.
- [2] M. Arvaneh, C. Guan, K. K. Ang, and C. Quek, "Optimizing the channel selection and classification accuracy in EEG-based BCI," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 6, pp. 1865–1873, 2011.
- [3] M. Thulasidas, C. Guan, and J. Wu, "Robust classification of EEG signal for brain-computer interface," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 1, pp. 24–29, 2006.
- [4] C. Y. Kee, R. K. Chetty, B. H. Khoo, and S. Ponnambalam, "Genetic algorithm and bayesian linear discriminant analysis based channel Selection method for P300 BCI," in *Trends in Intelligent Robotics, Automation, and Manufacturing*, vol. 330 of *Communications in Computer and Information Science*, pp. 226–235, Springer, Berlin, Germany, 2012.

- [5] T. Shibanoki, K. Shima, T. Tsuji, A. Otsuka, and T. Chin, "A quasi-optimal channel selection method for bioelectric signal classification using a partial Kullback-Leibler information measure," *IEEE Transactions on Bio-Medical Engineering*, vol. 60, no. 3, pp. 853–861, 2013.
- [6] P. Larrañaga and J. A. Lozano, *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*, Kluwer Academic Press, 2001.
- [7] P.-Y. Yin and H.-L. Wu, "Cyber-EDA: estimation of distribution algorithms with adaptive memory programming," *Mathematical Problems in Engineering*, vol. 2013, Article ID 132697, 11 pages, 2013.
- [8] A. Barachant and S. Bonnet, "Channel selection procedure using riemannian distance for BCI applications," in *Proceedings of the 5th International IEEE/EMBS Conference on Neural Engineering (NER '11)*, pp. 348–351, Cancun, Mexico, May 2011.
- [9] J. Yang, H. Singh, E. L. Hines et al., "Channel selection and classification of electroencephalogram signals: an artificial neural network and genetic algorithm-based approach," *Artificial Intelligence in Medicine*, vol. 55, no. 2, pp. 117–126, 2012.
- [10] T. N. Lal, M. Schröder, T. Hinterberger et al., "Support vector channel selection in BCI," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 1003–1010, 2004.
- [11] J. Müller-Gerking, G. Pfurtscheller, and H. Flyvbjerg, "Designing optimal spatial filters for single-trial EEG classification in a movement task," *Clinical Neurophysiology*, vol. 110, no. 5, pp. 787–798, 1999.
- [12] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Statistics for Engineering and Information Science, Springer, New York, NY, USA, 2nd edition, 2000.
- [13] M. Xu, H. Qi, L. Ma et al., "Channel selection based on phase measurement in P300-based brain-computer interface," *PLoS ONE*, vol. 8, no. 4, Article ID e60608, 2013.
- [14] B. A. S. Hasan, J. Q. Gan, and Q. Zhang, "Multi-objective evolutionary methods for channel selection in brain-computer interfaces: some preliminary experimental results," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '10)*, pp. 1–6, Barcelona, Spain, July 2010.
- [15] J.-Y. Kim, S.-M. Park, K.-E. Ko, and K.-B. Sim, "Optimal EEG channel selection for motor imagery BCI system using BPSO and GA," in *Robot Intelligence Technology and Applications 2012*, vol. 208 of *Advances in Intelligent Systems and Computing*, pp. 231–239, Springer, Berlin, Germany, 2013.
- [16] M. Schröder, M. Bogdan, T. Hinterberger, and N. Birbaumer, "Automated EEG feature selection for brain computer interfaces," in *Proceedings of the 1st International IEEE EMBS Conference on Neural Engineering*, pp. 626–629, IEEE, March 2003.
- [17] Q. Wei, Z. Lu, K. Chen, and Y. Ma, "Channel selection for optimizing feature extraction in an electrocorticogram-based brain-computer interface," *Journal of Clinical Neurophysiology*, vol. 27, no. 5, pp. 321–327, 2010.
- [18] R. Santana, C. Bielza, and P. Larrañaga, "Regularized logistic regression and multiobjective variable selection for classifying MEG data," *Biological Cybernetics*, vol. 106, no. 6–7, pp. 389–405, 2012.
- [19] Q. Wei and W. Tu, "Channel selection by genetic algorithms for classifying single-trial ECoG during motor imagery," in *Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '08)*, pp. 624–627, IEEE, Vancouver, Canada, August 2008.
- [20] S. Bhattacharyya, A. Sengupta, T. Chakraborti, A. Konar, and D. N. Tibarewala, "Automatic feature selection of motor imagery EEG signals using differential evolution and learning automata," *Medical & Biological Engineering & Computing*, vol. 52, no. 2, pp. 131–139, 2014.
- [21] K. Price, R. Storn, and J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, Natural Computing Series, Springer, Berlin, Germany, 2005.
- [22] X. Zhang, L. Jiao, A. Paul, Y. Yuan, Z. Wei, and Q. Song, "Semisupervised particle swarm optimization for classification," *Mathematical Problems in Engineering*, vol. 2014, Article ID 832135, 11 pages, 2014.
- [23] I. Inza, P. Larrañaga, R. Etxeberria, and B. Sierra, "Feature subset selection by Bayesian network-based optimization," *Artificial Intelligence*, vol. 123, no. 1–2, pp. 157–184, 2000.
- [24] Y. Zhang, S. Wang, P. Phillips, and G. Ji, "Binary PSO with mutation operator for feature selection using decision tree applied to spam detection," *Knowledge-Based Systems*, vol. 64, pp. 22–31, 2014.
- [25] O. AlZoubi, I. Koprinska, and R. A. Calvo, "Classification of brain-computer interface data," in *Proceedings of the 7th Australasian Data Mining Conference (AusDM '08)*, J. F. Roddick, J. Li, P. Christen, and P. J. Kennedy, Eds., vol. 87 of *CRPIT*, pp. 123–131, Australian Computer Society, Adelaide, Australia, November 2008, <http://dblp.uni-trier.de/db/conf/ausdm/ausdm2008.html#AlZoubiKC08>
- [26] B. Blankertz, K.-R. Müller, D. J. Krusienski et al., "The BCI competition III: validating alternative approaches to actual BCI problems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 153–159, 2006.
- [27] A. Schlögl, F. Lee, H. Bischof, and G. Pfurtscheller, "Characterization of four-class motor imagery EEG data for the BCI-competition 2005," *Journal of Neural Engineering*, vol. 2, no. 4, pp. 14–22, 2005.
- [28] F. Lee, R. Scherer, R. Leeb, C. Neuper, H. Bischof, and G. Pfurtscheller, "A comparative analysis of multi-class EEG classification for brain-computer interface," in *Proceedings of the 10th Computer Vision Winter Workshop (CVWW '05)*, Zell an der Pram, Austria, February 2005.
- [29] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [30] D. Meyer, F. Leisch, and K. Hornik, "The support vector machine under test," *Neurocomputing*, vol. 55, no. 1–2, pp. 169–186, 2003.
- [31] M. Hall, E. Frank, G. Holmes, B. P. P. Reutemann, and I. Witten, "The weka data mining software: an update," *SIGKDD Exploration Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.
- [32] C. K. Chow and C. N. Liu, "Approximating discrete probability distributions with dependence trees," *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 462–467, 1968.
- [33] M. Henrion, "Propagating uncertainty in bayesian networks by probabilistic logic sampling," in *Proceedings of the 2nd Annual Conference on Uncertainty in Artificial Intelligence*, J. F. Lemmer and L. N. Kanal, Eds., pp. 149–164, Elsevier, 1988.
- [34] S. Baluja and S. Davies, "Using optimal dependency-trees for combinatorial optimization: learning the structure of the search space," in *Proceedings of the 14th International Conference on Machine Learning (ICML '97)*, D. H. Fisher, Ed., pp. 30–38, Morgan Kaufmann, San Francisco, Calif, USA, July 1997.
- [35] M. Pelikan and H. Mühlenbein, "The bivariate marginal distribution algorithm," in *Advances in Soft Computing—Engineering*

- Design and Manufacturing*, R. Roy, T. Furuhashi, and P. Chawdhry, Eds., pp. 521–535, Springer, London, UK, 1999.
- [36] R. Santana, A. Ochoa, and M. R. Soto, “The mixture of trees factorized distribution algorithm,” in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '01)*, L. Spector, E. Goodman, A. Wu et al., Eds., pp. 543–550, Morgan Kaufmann, San Francisco, Calif, USA, July 2001.
- [37] R. Santana, C. Bielza, P. Larrañaga et al., “Mateda-2.0: a MATLAB package for the implementation and analysis of estimation of distribution algorithms,” *Journal of Statistical Software*, vol. 35, no. 7, pp. 1–30, 2010, <http://www.jstatsoft.org/v35/i07>.
- [38] L. Susperregi, A. Arruti, E. Jauregi et al., “Fusing multiple image transformations and a thermal sensor with kinect to improve person detection ability,” *Engineering Applications of Artificial Intelligence*, vol. 26, no. 8, pp. 1980–1991, 2013.
- [39] B. Sierra, E. Lazkano, E. Jauregi, and I. Irigoien, “Histogram distance-based Bayesian Network structure learning: a supervised classification specific approach,” *Decision Support Systems*, vol. 48, no. 1, pp. 180–190, 2009.
- [40] 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.
- [41] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona, “An extensive comparative study of cluster validity indices,” *Pattern Recognition*, vol. 46, no. 1, pp. 243–256, 2013.

Research Article

A Fault-Tolerant Filtering Algorithm for SINS/DVL/MCP Integrated Navigation System

Xiaosu Xu,¹ Peijuan Li,² and Jian-juan Liu³

¹Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, School of Instrument Science & Engineering, Southeast University, Nanjing 210096, China

²Industrial Center, Nanjing Institute of Technology, Nanjing 211167, China

³Henan University of Technology, Zhengzhou 450007, China

Correspondence should be addressed to Xiaosu Xu; xxs@seu.edu.cn

Received 3 July 2014; Revised 13 April 2015; Accepted 15 April 2015

Academic Editor: Filippo Ubertini

Copyright © 2015 Xiaosu Xu 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.

The Kalman filter (KF), which recursively generates a relatively optimal estimate of underlying system state based upon a series of observed measurements, has been widely used in integrated navigation system. Due to its dependence on the accuracy of system model and reliability of observation data, the precision of KF will degrade or even diverge, when using inaccurate model or trustless data set. In this paper, a fault-tolerant adaptive Kalman filter (FTAKF) algorithm for the integrated navigation system composed of a strapdown inertial navigation system (SINS), a Doppler velocity log (DVL), and a magnetic compass (MCP) is proposed. The evolutionary artificial neural networks (EANN) are used in self-learning and training of the intelligent data fusion algorithm. The proposed algorithm can significantly outperform the traditional KF in providing estimation continuously with higher accuracy and smoothing the KF outputs when observation data are inaccurate or unavailable for a short period. The experiments of the prototype verify the effectiveness of the proposed method.

1. Introduction

Integration technology has become one of the key issues in navigation because of its capability of overcoming the drawbacks of single strapdown inertial navigation system (SINS), such as error accumulation and error swing over a period of time. The most commonly used navigation systems are radio and satellite based system, including global positioning system (GPS) [1] and Doppler velocity log (DVL) [2, 3]; their performance is fundamentally limited by propagation of signals [4, 5]. The integrated navigation system proposed in this work is an economic navigating algorithm of high precision, in which the SINS is integrated with a DVL and a magnetic compass (MCP).

The Kalman filter (KF) is an effective optimal estimation algorithm and has been widely used in integrated navigation system since 1970s [6–9]. On one hand, in order to achieve a better performance with a KF, an accurate system model

and reliable observation data are indispensable. Moreover, the KF fails to estimate the state vector exactly when the observations noise increases [10–12]. However, in real engineering applications, it is usually difficult to satisfy the above conditions. For example, undersea DVLs output will be interfered or broken owing to the cliffy sea bottom and other vessel reflections [13, 14]. In order to solve this problem, the following method is often adopted: if the observation data from DVL or MCP are temporarily inaccurate or inaccessible, the integrated navigation system can be compromised to work under pure inertial mode. But, it is evident that the measuring accuracy will deteriorate remarkably and the system error will accumulate or swing over time in pure inertial mode. After the system returns to the integration mode when the observation data are recovered, the position error will definitely not be compensated although the errors of velocities and attitude angles can be reduced. Therefore, a fault-tolerant filtering algorithm that can maintain

the accuracy of an integrated navigation system through inexact or unavailable observation data is important and urgent [15–17].

Evolutionary programming, put forward by Fogel [18], is essential for random searching in that it boosts the survival of the fittest and organic evolution mechanism and searches for the optimal point in feasible space via population search strategy [19, 20]. Moreover, the artificial neural network (ANN), optimized by evolutionary programming, not only overcomes the obstacle of conventional ANN which may easily plunge into local minimum and requires long training time, but also avoids the problem in genetic algorithms caused by binary-coded and cross operation. Thanks to its superiorities, ANN has been applied in many fields such as financial forecasting [21], breast cancer diagnosis [22], and electricity supply industry [23], and so forth [24].

In this paper, a fault-tolerant integrated navigation system with an adaptive Kalman filter (AKF) based on evolutionary artificial neural networks (EANN) is proposed, in which the EANN is used in self-learning and training of the intelligent data fusion algorithm. The proposed fault-tolerant adaptive Kalman filter (FTAKF) algorithm is able to maintain a smooth output under the condition when observation data are inaccurate or unavailable. The prototype experiment indicates that the algorithm can efficiently outperform the traditional KF with a higher accuracy when observation data are inaccurate or unavailable.

This paper is organized as follows. In Section 2, the inertial error model is introduced and a corresponding mathematical model in state and measurement space is described. In Section 3, we propose the AKF algorithm and compare its characteristics and properties with those of traditional KF algorithm. A fast ANN based on evolutionary programming is proposed in Section 4. In Sections 5 and 6, applications of algorithm in SINS/DVL/MCP are presented and several tests are conducted to evaluate the performance of the intelligent navigation system with FTAKF. Conclusions are made in Section 7.

2. Mathematical Model for Filter

2.1. Error Model of Integrated Navigation System

(1) *Inertial Error Model.* Here we define the navigation coordinates frame as an east-north-vertical (ENV) geography coordinates frame. The attitude angle, velocity, and position errors are given in the following. Attitude angle error is determined by

$$\dot{\boldsymbol{\phi}}^n = \delta\boldsymbol{\omega}_{ie}^n + \delta\boldsymbol{\omega}_{en}^n - (\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \boldsymbol{\phi}^n + \mathbf{C}_b^n \boldsymbol{\varepsilon}^b, \quad (1)$$

where $\boldsymbol{\phi}^n = [\phi_E, \phi_N, \phi_U]^T$ is the orientation error vector of the calculated platform represented in the navigation coordinates frame, $\boldsymbol{\omega}_{ie}^n = [0, \omega_{ie} \cos L, \omega_{ie} \sin L]^T$ represents the rotation projection of the earth onto the ENV axes with $\omega_{ie} = 7.292115 \times 10^{-5}$ rad/s to be the angular velocity of the rotation

of the earth, and $\boldsymbol{\omega}_{en}^n = [-V_N/(R_n + h), V_E/(R_e + h), (V_E/(R_e + h)) \tan L]^T$ denotes the angular velocity of the rotation of a navigation coordinates frame relative to the earth with V_E , V_N , and V_U that represent the linear velocities, λ , L , and h that represent the position coordinates (longitude latitude and height, resp.), and R_n , R_e that represent the radii of the curvatures along the meridian and parallel, respectively (Table 2). Moreover, $\boldsymbol{\varepsilon}^b = [\varepsilon_x, \varepsilon_y, \varepsilon_z]$ is the gyro-drifts in the body coordinates frame with $\dot{\boldsymbol{\varepsilon}}^b = \mathbf{0}$. The last term \mathbf{C}_b^n represents the attitude matrix, which can be expressed by the attitude angles ψ , γ , and θ .

Given the definition $[s_\psi, c_\psi, s_\gamma, c_\gamma, s_\theta, c_\theta]^T = [\sin \psi, \cos \psi, \sin \gamma, \cos \gamma, \sin \theta, \cos \theta]^T$, \mathbf{C}_b^n can be written as

$$\mathbf{C}_b^n = \begin{bmatrix} c_\gamma c_\psi + s_\gamma s_\psi s_\theta & s_\psi c_\theta & s_\gamma c_\psi - c_\gamma s_\psi s_\theta \\ -c_\gamma s_\psi + s_\gamma c_\psi s_\theta & c_\psi c_\theta & -s_\gamma s_\psi - c_\gamma c_\psi s_\theta \\ -s_\gamma c_\theta & s_\theta & c_\gamma c_\theta \end{bmatrix}. \quad (2)$$

Velocity error is defined as

$$\delta \dot{\mathbf{V}}^n = \mathbf{f}^n \times \boldsymbol{\phi}^n - (2\delta\boldsymbol{\omega}_{ie}^n + \delta\boldsymbol{\omega}_{en}^n) \times \mathbf{V}^n - (2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \delta \mathbf{V}^n + \mathbf{C}_b^n \mathbf{V}^b, \quad (3)$$

where $\delta \mathbf{V}^n = [\delta V_E, \delta V_N, \delta V_U]^T$ viewed as the linear velocity error vector is determined by the specific force vector $\mathbf{f}^n = [f_E, f_N, f_U]^T$ and the bias vector $\mathbf{V}^b = [V_x, V_y, V_z]$ of accelerometer in the body coordinates frame, along with $\delta\boldsymbol{\omega}_{ie}^n$ and $\delta\boldsymbol{\omega}_{en}^n$ that represent the slow variations of $\boldsymbol{\omega}_{ie}^n$ and $\boldsymbol{\omega}_{en}^n$, respectively. Here, $\dot{\mathbf{V}}^b = \mathbf{0}$.

Position error is given by

$$\begin{aligned} \delta \dot{L} &= \frac{\delta V_N}{R_n + h} - \frac{V_N}{(R_n + h)^2} \delta h, \\ \delta \dot{\lambda} &= \frac{\delta V_E}{R_e + h} \sec L + \frac{V_E}{R_e + h} \sec L \tan L \delta L \\ &\quad - \frac{V_E}{(R_e + h)^2} \sec L \delta h, \\ \delta \dot{h} &= \delta V_U, \end{aligned} \quad (4)$$

where $\delta\lambda$, δL , and δh represent the position errors.

(2) *DVL Error Model.* The velocity of DVL in the body coordinates frame $\tilde{\mathbf{V}}_d$ can be described as follows:

$$\tilde{\mathbf{V}}_d = (\mathbf{I} + \delta \mathbf{K}) \mathbf{V}_d + \delta \mathbf{V}_d. \quad (5)$$

Here, $\delta \mathbf{K} = [\delta K_{dx}, \delta K_{dy}, \delta K_{dz}]^T$ is the scale factor error assumed to be random constant and $\delta \dot{\mathbf{K}} = \mathbf{0}$ and $\delta \mathbf{V}_d = [\delta V_{dx}, \delta V_{dy}, \delta V_{dz}]^T$ represents the linear velocity errors of

DVL, which can be characterized by first-order Markov process as

$$\begin{aligned}\delta\dot{V}_{dx} &= -\frac{1}{\tau_{dx}}\delta V_{dx} + w_{dx}, \\ \delta\dot{V}_{dy} &= -\frac{1}{\tau_{dy}}\delta V_{dy} + w_{dy}, \\ \delta\dot{V}_{dz} &= -\frac{1}{\tau_{dz}}\delta V_{dz} + w_{dz},\end{aligned}\quad (6)$$

where $\boldsymbol{\tau}_d = [\tau_{dx}, \tau_{dy}, \tau_{dz}]^T$ is the time correlation coefficient of this Markov process and $\mathbf{w}_d = [w_{dx}, w_{dy}, w_{dz}]^T$ is the input noise of DVL.

(3) *MCP Error Model*. Heading attitude angle error of MCP $\delta\psi_m$ can be represented as first-order Markov process with

$$\delta\dot{\psi}_m = -\frac{1}{\tau_m}\delta\psi_m + w_m, \quad (7)$$

where τ_m is the time correlation coefficient and w_m is the input noise of MCP.

2.2. *State Space*. The error models given in (1), (3)-(4), and (6)-(7) have a standard form given by

$$\dot{\mathbf{X}} = \mathbf{F} \cdot \mathbf{X} + \mathbf{\Gamma} \cdot \mathbf{W}, \quad (8)$$

where matrices \mathbf{F} and $\mathbf{\Gamma}$ can be established from the above error model and \mathbf{W} is the input (system) noise:

$$\begin{aligned}\mathbf{W} &= [w_{gx}, w_{gy}, w_{gz}, w_{ax}, w_{ay}, w_{az}, w_{dx}, w_{dy}, w_{dz}, w_m], \\ &\quad (9)\end{aligned}$$

where w_{gx} , w_{gy} , and w_{gz} are the white noise of the gyros and w_{ax} , w_{ay} , and w_{az} are the white noise of the accelerometers. The state vector \mathbf{X} of the integrated navigation system is defined as

$$\begin{aligned}\mathbf{X} &= [\delta V_E, \delta V_N, \delta V_U, \phi_E, \phi_N, \phi_U, \delta L, \delta \lambda, \delta h, \nabla_x, \nabla_y, \\ &\quad \nabla_z, \varepsilon_x, \varepsilon_y, \varepsilon_z, \delta V_{dx}, \delta V_{dy}, \delta V_{dz}, \delta K_{dx}, \delta K_{dy}, \delta K_{dz}, \\ &\quad \delta\psi_m].\end{aligned}\quad (10)$$

2.3. *Measurement Space*. Supposing that the error of the velocity of DVL in the navigation coordinates frame \mathbf{V}_d^n is mainly caused by platform misalignment, the error of $\tilde{\mathbf{V}}_d^n$ can be described as

$$\begin{aligned}\tilde{\mathbf{V}}_d^n &= \mathbf{V}_d^n + \delta\mathbf{V}_d^n = \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d + \delta\mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d \\ &= \mathbf{V}_d^n + \delta\mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d,\end{aligned}\quad (11)$$

where \mathbf{C}_d^b is a constant matrix which translates the velocity of DVL \mathbf{V}_d from the instrumental frame to the body coordinates

frame. The perturbation of the attitude matrix $\delta\mathbf{C}_b^n$ is defined as

$$\delta\mathbf{C}_b^n = (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n, \quad (12)$$

where p is the calculating navigation coordinate frame. Substituting (12) into (11), it yields

$$\tilde{\mathbf{V}}_d^n = \mathbf{V}_d^n + (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d. \quad (13)$$

After differentiating the velocity of SINS and DVL, the measurement model can be written as

$$\begin{aligned}\Delta\mathbf{V}^n &= \tilde{\mathbf{V}}^n - \tilde{\mathbf{V}}_d^n + \boldsymbol{\eta}_d \\ &= \mathbf{V}^n + \delta\mathbf{V}^n - \mathbf{V}_d^n - (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d + \boldsymbol{\eta}_d \\ &= \delta\mathbf{V}^n - (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d + \boldsymbol{\eta}_d,\end{aligned}\quad (14)$$

where the measurement velocity vector $\Delta\mathbf{V}^n = [\Delta V_E, \Delta V_N, \Delta V_U]$ and $\delta\mathbf{V}_d^n = (\mathbf{I} - \mathbf{C}_n^p) \mathbf{C}_b^n \mathbf{C}_d^b \mathbf{V}_d$ can be calculated from

$$\begin{aligned}\delta V_{dE} &= \phi_U V_N - \phi_N V_U + \mathbf{C}_b^n(1, 1) \delta V_{dx} \\ &\quad + \mathbf{C}_b^n(1, 2) \delta V_{dy} + \mathbf{C}_b^n(1, 3) \delta V_{dz} \\ &\quad + \mathbf{C}_b^n(1, 1) V_E \delta K_{dx} + \mathbf{C}_b^n(1, 2) V_N \delta K_{dy} \\ &\quad + \mathbf{C}_b^n(1, 3) V_U \delta K_{dz}, \\ \delta V_{dN} &= \phi_E V_U - \phi_U V_E + \mathbf{C}_b^n(2, 1) \delta V_{dx} \\ &\quad + \mathbf{C}_b^n(2, 2) \delta V_{dy} + \mathbf{C}_b^n(2, 3) \delta V_{dz} \\ &\quad + \mathbf{C}_b^n(2, 1) V_E \delta K_{dx} + \mathbf{C}_b^n(2, 2) V_N \delta K_{dy} \\ &\quad + \mathbf{C}_b^n(2, 3) V_U \delta K_{dz}, \\ \delta V_{dU} &= \phi_N V_E - \phi_E V_N + \mathbf{C}_b^n(3, 1) \delta V_{dx} \\ &\quad + \mathbf{C}_b^n(3, 2) \delta V_{dy} + \mathbf{C}_b^n(3, 3) \delta V_{dz} \\ &\quad + \mathbf{C}_b^n(3, 1) V_E \delta K_{dx} + \mathbf{C}_b^n(3, 2) V_N \delta K_{dy} \\ &\quad + \mathbf{C}_b^n(3, 3) V_U \delta K_{dz}.\end{aligned}\quad (15)$$

Similarly, after differentiating the heading angle of SINS and MCP, the measurement model is given as

$$\begin{aligned}\Delta\psi &= \tilde{\psi} - \tilde{\psi}_m + \eta_m = \psi + \delta\psi - \psi_m - \delta\psi_m + \eta_m \\ &= \delta\psi - \delta\psi_m + \eta_m,\end{aligned}\quad (16)$$

where

$$\begin{aligned}\delta\psi &= -\frac{\mathbf{C}_b^n(1, 2) \cdot \mathbf{C}_b^n(3, 2)}{[\mathbf{C}_b^n(1, 2)]^2 + [\mathbf{C}_b^n(2, 2)]^2} \phi_E \\ &\quad - \frac{\mathbf{C}_b^n(2, 2) \cdot \mathbf{C}_b^n(3, 2)}{[\mathbf{C}_b^n(1, 2)]^2 + [\mathbf{C}_b^n(2, 2)]^2} \phi_N + \phi_U.\end{aligned}\quad (17)$$

Based on the previous analysis, the measurement model is defined as

$$\mathbf{Z}_k = \mathbf{H}_k \mathbf{X}_k + \boldsymbol{\eta}_k, \quad (18)$$

where $\mathbf{Z} = [\Delta V_E, \Delta V_N, \Delta V_U, \Delta \psi]^T$ is the observation vector, \mathbf{H}_k is the observation matrix at the moment of k which can be established from (15) and (17), and vector $\boldsymbol{\eta}_k$ represents the Gaussian measurement noise with zero mean value and covariance matrix \mathbf{R} .

3. AKF Algorithm to Integrated Navigation System

Given the differential equations, we can have a continuous state transition matrix \mathbf{F} for error states. The continuous transition matrix can be discretized for KF application by using the first-order Taylor series approximation. The corresponding discrete transition matrix can be computed as

$$\Phi = \mathbf{I} + \mathbf{F}\Delta T, \quad (19)$$

where ΔT is chosen as 0.01 s. Thus, the state equations and observation equation can be written as

$$\begin{aligned} \mathbf{X}_k &= \Phi_{k,k-1} \mathbf{X}_{k-1} + \Gamma_{k-1} \mathbf{W}_{k-1}, \\ \mathbf{Z}_k &= \mathbf{H}_k \mathbf{X}_k + \boldsymbol{\eta}_k, \end{aligned} \quad (20)$$

where \mathbf{X}_k is the 22-dimensional state vector at the moment of k , \mathbf{Z}_k is the 4-dimensional observation vector at the moment of k , $\Phi_{k,k-1}$ is state-transfer-matrix from sample time $k-1$ to k , Γ_{k-1} is the system noise matrix at the moment of $k-1$, and \mathbf{W}_{k-1} is system noise. Here, $\{\mathbf{W}_k\}$ and $\{\boldsymbol{\eta}_k\}$ are uncorrelated white noise sequences with zero mean. So we can have $E\{\mathbf{W}_k \mathbf{W}_j^T\} = \mathbf{Q}_k \delta_{kj}$ and $E\{\boldsymbol{\eta}_k \boldsymbol{\eta}_j^T\} = \mathbf{R}_k \delta_{kj}$. \mathbf{Q}_k is the variance matrix of system noise and \mathbf{R}_k is the variance matrix of observation noise.

The equations of AKF based on the new information sequence are written as

$$\begin{aligned} \widehat{\mathbf{X}}_{k/k-1} &= \Phi_{k,k-1} \widehat{\mathbf{X}}_{k-1}, \\ \mathbf{V}_k &= \mathbf{Z}_k - \mathbf{H}_k \widehat{\mathbf{X}}_{k/k-1}, \\ \mathbf{P}_{k/k-1} &= \mathbf{K}_{k-1} \mathbf{V}_k \mathbf{V}_k^T \mathbf{K}_{k-1}^T + \mathbf{P}_{k-1}, \\ \mathbf{K}_k &= \mathbf{P}_{k/k-1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k/k-1} \mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \\ \widehat{\mathbf{X}}_k &= \widehat{\mathbf{X}}_{k/k-1} + \mathbf{K}_k \mathbf{V}_k, \\ \mathbf{P}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k/k-1}. \end{aligned} \quad (21)$$

Compared with traditional KF equations, there are two differences. Firstly, \mathbf{Q}_k does not appear in above filter equations. That is to say, the statistical properties of the system noise are not needed in AKF, which is a highlight of this algorithm. Secondly, the initial value \mathbf{K}_0 is needed, and it can be obtained by applying the traditional KF in the first loop. The AKF scheme of the integrated navigation system is shown in Figure 1.

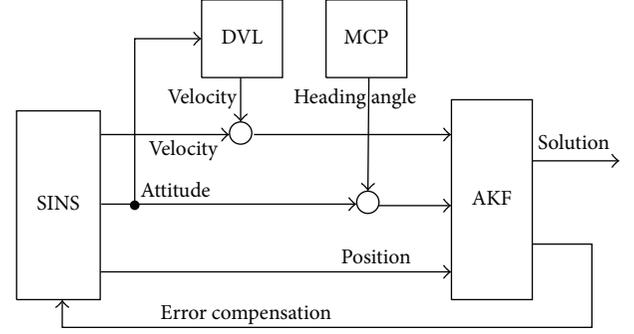


FIGURE 1: AKF scheme of SINS/DVL/MCP.

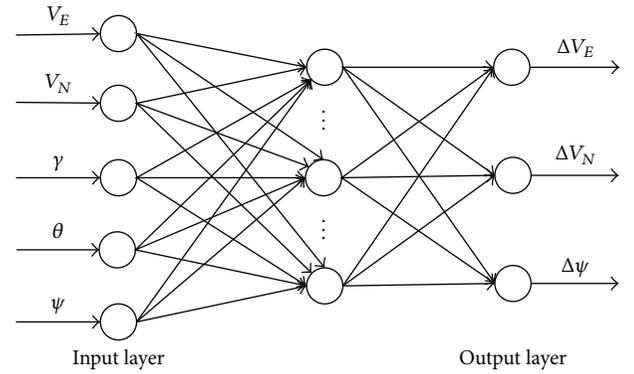


FIGURE 2: Artificial neural networks model.

4. Evolutionary Artificial Neural Networks

In recent years, ANN has been widely used in various fields, especially the backpropagation (BP) training [25]. However, drawbacks like poor convergence and failure in obtaining the global optimum are the bottleneck in the practical applications [26]. Articles [27–29] use genetic algorithms to optimize the parameters of neural networks. But there exist some imperfections. In this work, evolutionary programming is used to optimize the structure and connection weights in neural networks.

4.1. Artificial Neural Networks Model. ANN theory has proved that three-layer feedforward networks containing only one hidden layer can be a nonlinear mapping of any system with any precision [30]. According to this, the ANN model used in this paper consists of an input layer, a hidden layer, and an output layer. The network structure shown in Figure 2 treats the output navigation parameters of SINS as the input of the networks and the measurement errors as the output of the networks. Considering the limitation of underwater experimental environment, tests are conducted on a land vehicle, and the experimental area is flat and the speed of the vertical direction V_U can be regarded as zero. Hence, there are 5 nodes in input layer while there are 3 in output layer.

Based on the assumption that the number of hidden layer nodes is m , we have $\mathbf{x}' \in \mathbf{R}^m$, $\mathbf{x}' = (x'_0, x'_1, \dots, x'_{m-1})^T$, and the connection weights between input and hidden layers are ω_{ij} ($i = 0, 1, \dots, 4$; $j = 0, 1, \dots, m-1$), with thresholds to be Θ_j . The connection weights between output and hidden layers are represented as ω_{jk} ($k = 0, 1, 2$), with thresholds to be Θ_k . Then the output of each layer can be written as

$$\begin{aligned} x'_j &= f_1 \left(\sum_{i=0}^4 \omega_{ij} x_i - \Theta_j \right), \\ y_k &= f_2 \left(\sum_{j=0}^{m-1} \omega_{jk} x'_j - \Theta_k \right), \end{aligned} \quad (22)$$

where f_1 and f_2 are transfer functions.

In order to study the nonlinear properties of biological neurons, usually f_1 is chosen as a sigmoid function $f_1(x) = 1/(1 + e^{-x})$ and f_2 is chosen as a linear function $f_2(x) = x$. During optimizing, the threshold value is treated as import power with value of -1 .

4.2. Artificial Neural Networks Optimized by Evolutionary Programming. Darwin's evolutionary programming is a new search algorithm which is largely dependent on the mutation operator, while the genetic algorithms rely greatly on the crossover operator. When using the evolutionary programming as an optimization algorithm, users need not to be constrained to a particular code structure (representation) or a mutation strategy (evolution operator). Therefore, the flexibility and convenience of evolutionary programming make it surpass the conventional genetic algorithm. The steps for ANN optimization with the network structure and parameters using evolutionary programming are described as follows.

(1) *Initialization.* Here the population of neural networks is set by an initial quantity and the number of hidden layer neurons and the connection weights are initialized randomly within a certain range. Population can be expressed as $\mathbf{P} = \{X_1, X_2, \dots, X_{2N}\}$, where X_i ($i = 1, 2, \dots, 2N$) is the weight distribution of the network and defined as $X_i = [W_1, W_2, \dots, W_M]$ with W_k ($k = 1, 2, \dots, M$) to be the weight between every two connecting nodes of the network.

(2) *Performance Calculation and Categorization.* The training procedure of neural networks uses the overall mean square deviation as a metric. Training will be terminated if the networks' overall mean square deviation is less than a preset value. Here, the set error is defined as the individual survival environment, which is the goal of every individual evolutionary. For each individual, the fitness function is defined as

$$\begin{aligned} f &= \frac{1}{1 + E}, \\ E &= \frac{1}{2} \sum_{l=1}^p \sum_{k=1}^3 (d_k - o_k)^2, \end{aligned} \quad (23)$$

where p is the number of the imported samples and d_k and o_k are the desired and actual outputs of the node k of the output layer. Each individual is sorted in a descending order according to fitness.

(3) *Termination Condition Check.* When the best individual meets the requirement or number of loops exceeds the limitation of evolutionary generation, turn to *Step (5)*; otherwise continue.

(4) *New Population Generation.* Firstly, n individuals in the current generation are picked as part of the next generation, while removing the rest. Secondly, n new individuals are generated from individuals selected by mutation. These $2n$ individuals gather to form the new population. Here Gaussian variation is used as the mutation operator. Moreover, according to the characteristics of BP neural networks, the actual error of network is introduced as the enlightening knowledge in the formula; namely,

$$\begin{aligned} W_k(t+1) &= W_k(t) + \Delta W_k, \\ \Delta W_k &= L \sqrt{\frac{1}{F_{X_i}}} N(0, 1), \\ L &= C \exp \frac{n}{n_{\max}}, \end{aligned} \quad (24)$$

where $W_k(t+1)$ and $W_k(t)$ are the weights of the $(t+1)$ th generation and (t) th generation, respectively, with subscript to be $k = 1, 2, \dots, M$, $N(0, 1)$ represents a standard normal distribution, L and C are definition coefficients, n_{\max} is the maximum training cycles of the network, and n is the number of the individual survival generations. When the new population is generated, jump to *Step (2)*.

(5) *Termination Cycle.* Terminate the evolution procedure and then train the optimal neural networks using BP algorithm until all requirements are met.

5. Applications of Algorithm in SINS/DVL/MCP

On account of the previous algorithm, the FTAKF method based on EANN technology can be applied to the SINS/DVL/MCP integrated system.

The integrated navigation system with an EANN based AKF is shown in Figure 3.

When the observation data are diagnosed as available and reliable, EANN works at training state and uses SINS velocity and attitude information as input while the measurement errors are treated as the desired output. In this case switch 1 is connected and switch 2 is off. When the observation data is unavailable or unreliable, EANN works at forecasting state, with switch 1 to be off and switch 2 to be connected.

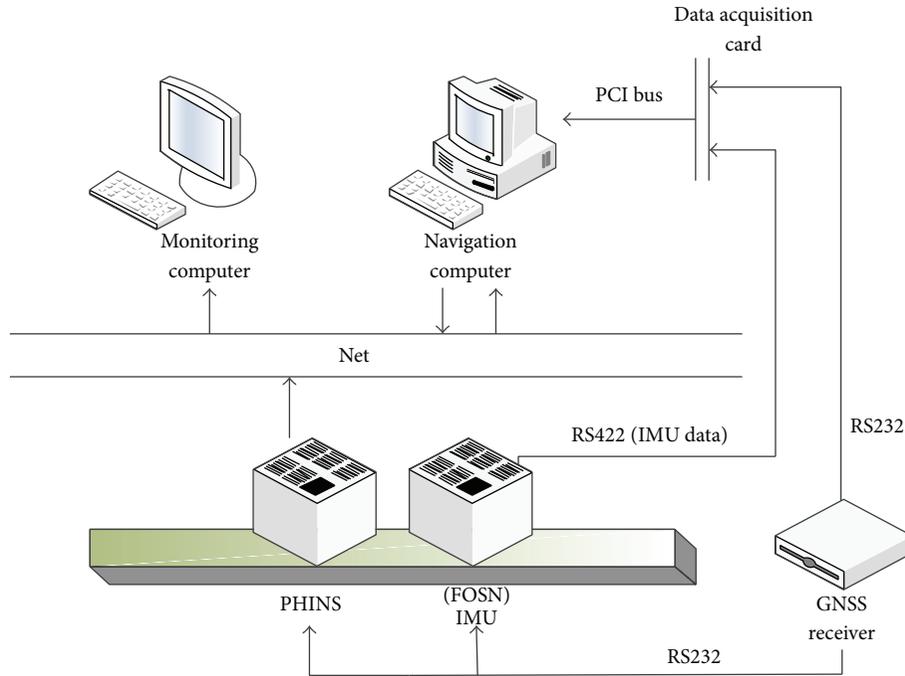


FIGURE 4: Vehicle test structure.



FIGURE 5: Experimental equipment: tested system and comparison unit.



FIGURE 6: Working scene.

TABLE 2: Statistical results of horizontal position errors.

	Longitude error (unit: m)	Latitude error (unit: m)
Experiment 1	141	798
Experiment 2	1.5	21

Experiment 1 (for a traditional KF algorithm). The experiment consists of three phases.

Phase 1. From start time as the yellow arrow on the map to 1000 seconds, this phase marked by yellow line on the map, the tested system works in the integration mode, and the observation data are available.

Phase 2. From 1000 seconds to 1600 seconds marked by black line on the map, the tested system works in the pure inertial mode, and the observation data are unavailable.

Phase 3. From 1600 seconds to 2800 seconds marked by light green line on the map, the tested system recovers from the pure inertial mode and works in the integration mode, while the observation data are renewed again.

Figures 7–9 show the system’s navigation errors of Experiment 1. Figure 10 shows the navigation route of Experiment 1.

According to Figures 7 to 9, during Phase 2, the horizontal attitude errors are bigger than the attitude errors when the system is working in the integrated mode. For lack of outside velocity and yaw information, the yaw error, velocity error, and the position errors are increased fast as time grows when

(2) *Experiment Route.* The experiment was conducted on the campus of Southeast University.

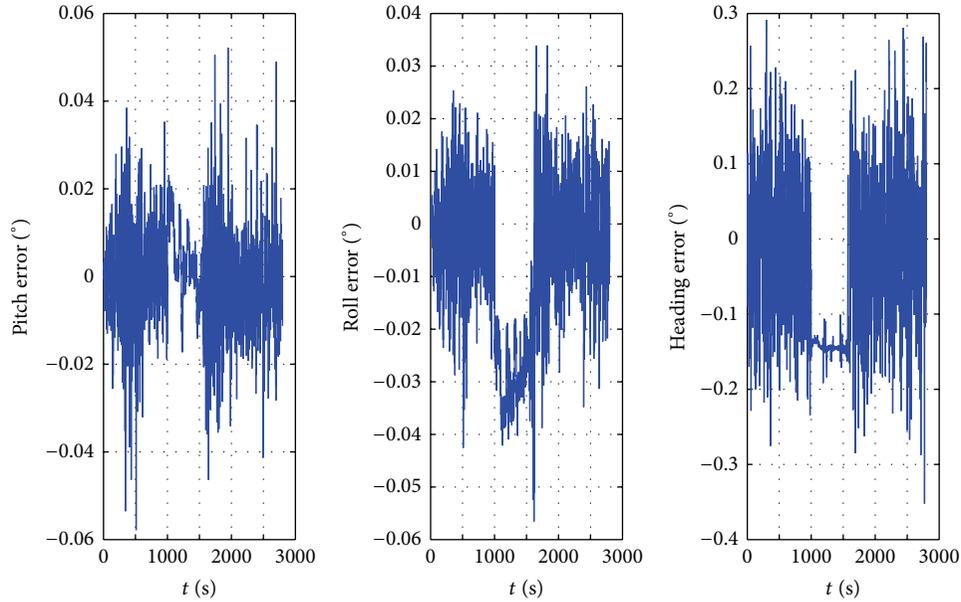


FIGURE 7: Experiment 1, attitude errors.

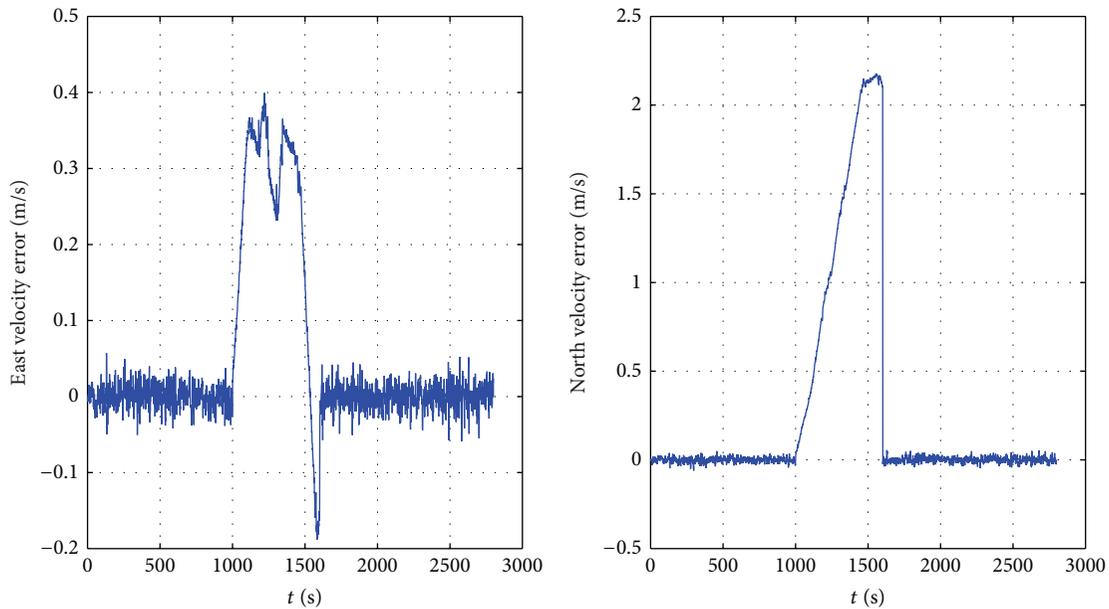


FIGURE 8: Experiment 1, velocity errors.

the system works in the pure inertial mode, and the precision of the system depends highly on IMU's accuracies.

From 1600 seconds, the observation data are available again and the system returns to the integration mode. The attitude and velocity errors can be lessened due to correction effects of KF, while the position errors cannot be eliminated.

(3) Experiment Results

Experiment 2 (for the FTAKF algorithm based on EANN). The experiment also consists of three phases like Experiment 1. In order to verify the effects of evolutionary

neural networks, the observation data will be unavailable during Phase 2 from 1000 to 1600 seconds; the tested system will work in the fault-tolerant filtering mode instead of the pure inertial mode.

Experiment results shown in Figures 11–13 indicate that when the observation data are unavailable, the evolutionary neural network, which has completed training process, works on the forecasting mode and provides KF with required inputs. Thus, the KF can keep operating in a continuous manner and the system precision can be maintained stably. Figure 14 shows the navigation route of Experiment 2.

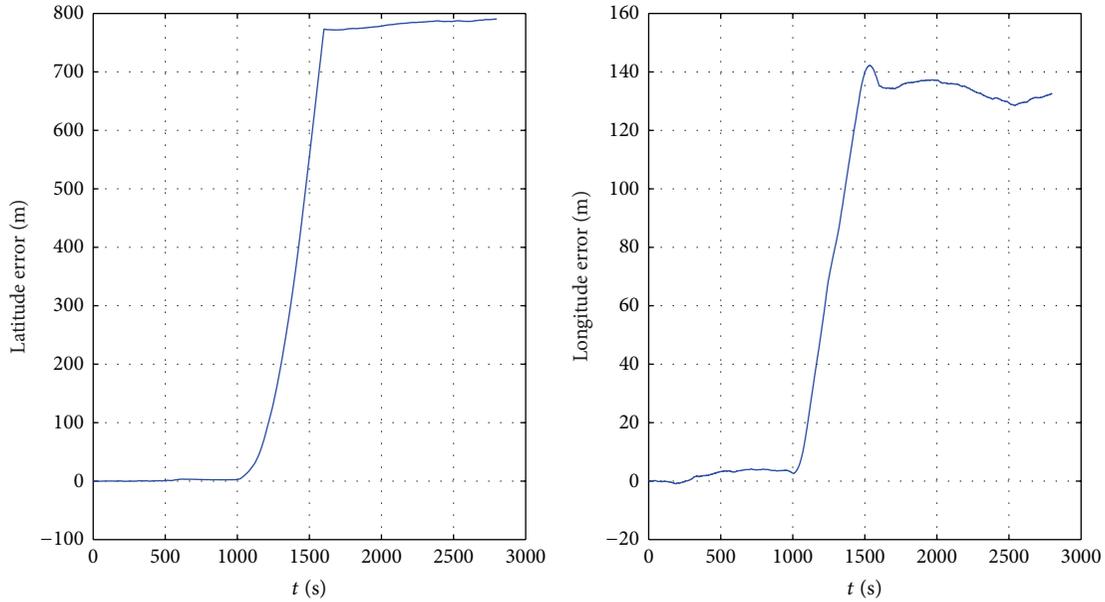
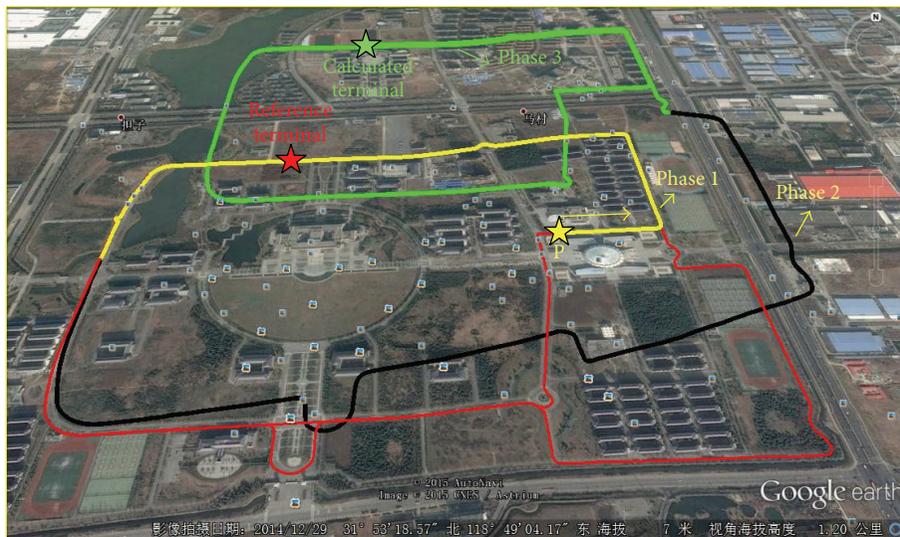


FIGURE 9: Experiment 1, position errors.



- Phase 1
- Phase 2
- Phase 3
- Reference route
- ★ Start point
- ★ Calculated terminal
- ★ Reference terminal

FIGURE 10: Experimental route map with traditional KF algorithm.

7. Conclusion

In this paper, aimed at addressing the problems of inaccurate model and unreliable observation data for KF in practical applications of AUV (autonomous underwater vehicle), a FTAKF algorithm for integrated navigation system based on EANN is proposed and tested. The conclusions are listed as follows:

(1) The performance of a traditional KF is vulnerable due to its dependence on precision of the system model,

such as state equation, observation equation, and noise statistic characters. In order to reduce the dependence on the precision of the system model, an AKF algorithm is designed.

(2) The FTAKF algorithm based on EANN is capable of maintaining a smooth filter output even under the severe condition when observation data are inaccurate or unavailable and of enhancing the precision and reliability of integrated navigation system.

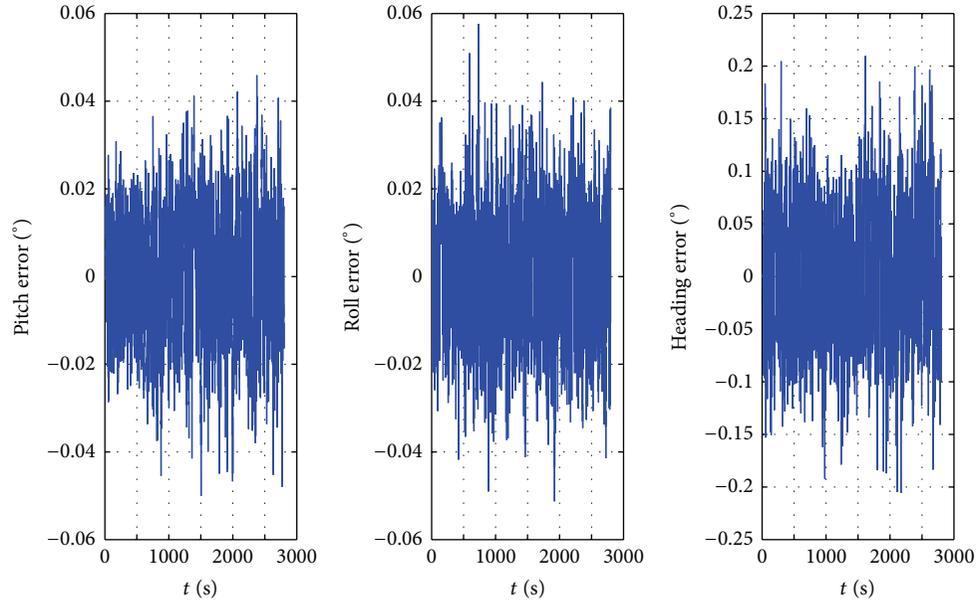


FIGURE 11: Experiment 2, attitude errors.

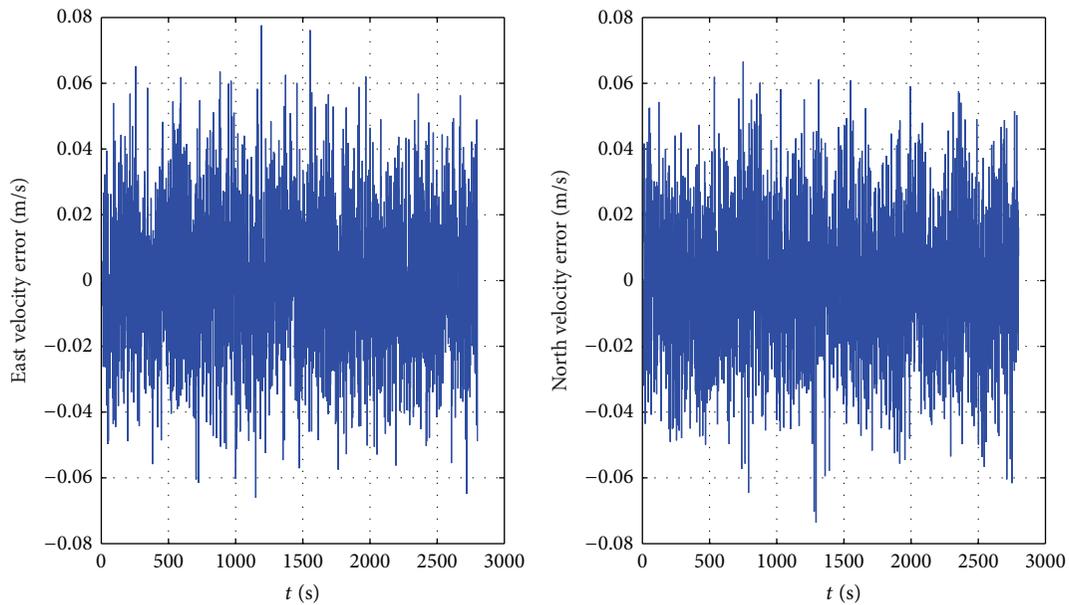


FIGURE 12: Experiment 2, velocity errors.

- (3) The results of experiments show that the FTAKF algorithm based on EANN proposed in this paper is effective and practical for engineering application.
- (4) It should be noted that, because of the utilization of the evolutionary neural network, the prerequisite for this system is that the training time of neural networks must be longer than the period that observation data is unavailable.

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 the Chinese National Natural Science Foundation (51175082, 60874092) and the Project

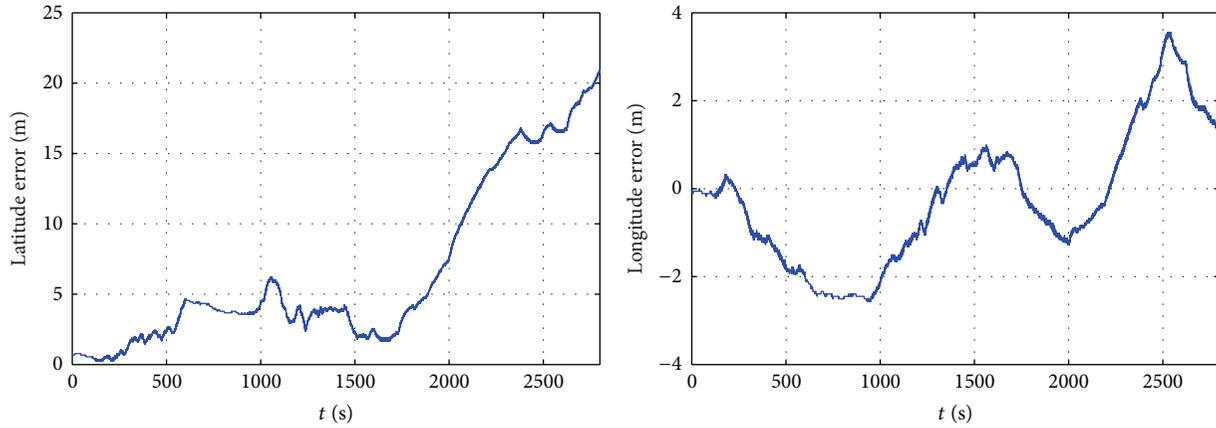


FIGURE 13: Experiment 2, position errors.

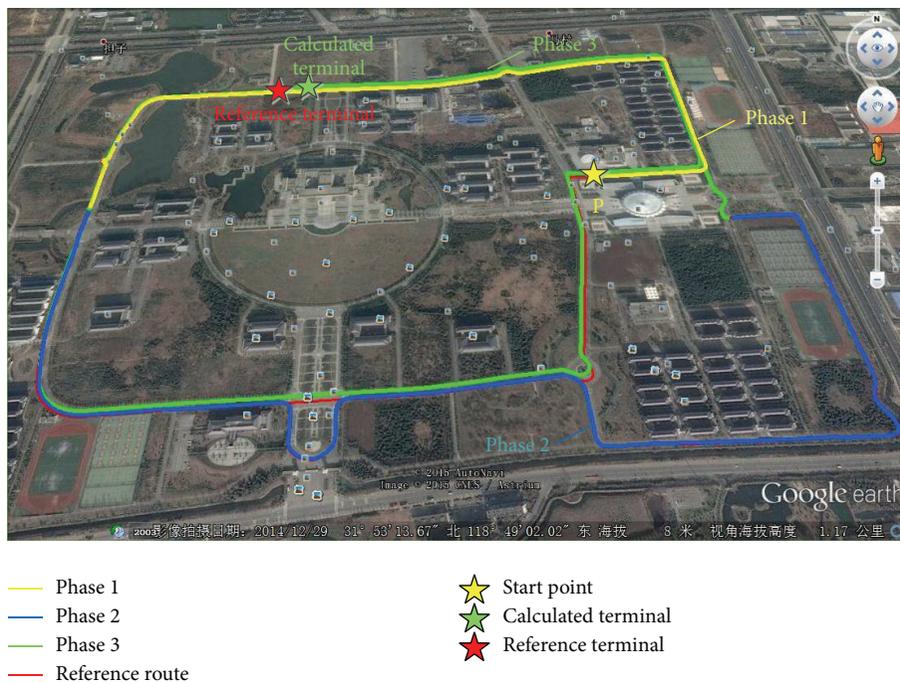


FIGURE 14: Experimental route map with FTAKF algorithm.

Supported by the Foundation of Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, China (201404), for the support of this work.

References

[1] R. McEwen, H. Thomas, D. Weber, and F. Psota, "Performance of an AUV navigation system at arctic latitudes," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 443–454, 2005.

[2] P. Ridao, D. Ribas, E. Hernandez, and A. Rusu, "USBL/DVL navigation through delayed position fixes," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '11)*, pp. 2344–2349, Shanghai, China, May 2011.

[3] B. Jalving, K. Gade, K. Svartveit, A. Willumsen, and R. Sørhagen, "DVL velocity aiding in the HUGIN 1000 integrated inertial navigation system," *Modeling, Identification and Control*, vol. 25, no. 4, pp. 223–235, 2004.

[4] T. Zhang and X. Xu, "A new method of seamless land navigation for GPS/INS integrated system," *Measurement*, vol. 45, no. 4, pp. 691–701, 2012.

[5] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: a review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, 2014.

[6] M. J. Dove, "Kalman filter techniques in marine integrated navigation systems," *Journal of Navigation*, vol. 30, no. 1, pp. 135–145, 1977.

[7] S. A. Broatch and A. J. Henley, "An integrated navigation system manager using federated Kalman filtering," in *Proceedings of the*

- IEEE National Aerospace and Electronics Conference (NAECON '91)*, pp. 422–426, Dayton, Ohio, USA, May 1991.
- [8] X.-Y. Chen, J. Yu, and X.-F. Zhu, “Theoretical analysis and application of Kalman filters for ultra-tight global position system/inertial navigation system integration,” *Transactions of the Institute of Measurement and Control*, vol. 34, no. 5, pp. 648–662, 2012.
- [9] T. Xu, “Adaptive Kalman filter for INS/GPS integrated navigation system,” in *Proceedings of the 2nd International Conference on Measurement, Instrumentation and Automation*, vol. 336–338, pp. 332–335, Guilin, China, April 2013.
- [10] T. Abdelazim, W. Abdel-Hamid, N. El-Sheimy, and E. H. Shin, “Experimental results of an adaptive fuzzy network Kalman filtering integration for low cost navigation applications,” in *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society: Fuzzy Sets in the Heart of the Canadian Rockies (NAFIPS '04)*, pp. 844–849, Banff, Canada, June 2004.
- [11] H. Yanling, Z. Yi, and S. Feng, “An anomaly recognition algorithm for DVL,” in *Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA '11)*, pp. 2423–2427, IEEE, Beijing, China, August 2011.
- [12] Y. Zhang, Y. Ding, and N. Li, “A tightly integrated SINS/DVL navigation method for autonomous underwater vehicle,” in *Proceedings of the 5th International Conference on Computational and Information Sciences (ICCIS '13)*, pp. 1107–1110, Shiyuan, China, June 2013.
- [13] K.-H. Tang, M.-M. Jiang, and J. Weng, “Design of SINS/Phased Array DVL integrated navigation system for underwater vehicle based on adaptive filtering,” *Journal of Chinese Inertial Technology*, vol. 21, no. 1, pp. 65–70, 2013 (Chinese).
- [14] Ø. Hegrehaes and O. Hallingstad, “Model-aided INS with sea current estimation for robust underwater navigation,” *IEEE Journal of Oceanic Engineering*, vol. 36, no. 2, pp. 316–337, 2011.
- [15] H. Miller and D. A. Hiltz, “Fault tolerant integrated inertial navigation/global positioning systems for next generation spacecraft,” in *Proceedings of the IEEE/AIAA 10th Digital Avionics Systems Conference*, pp. 207–212, Los Angeles, Calif, USA, October 1991.
- [16] T. Zhang, X.-S. Xu, Y. Li, and S.-P. Gong, “AUV Fault-tolerant technology based on inertial navigation and underwater acoustics assisted navigation system,” *Journal of Chinese Inertial Technology*, vol. 21, no. 4, pp. 512–516, 2013 (Chinese).
- [17] M. Ushaq and F. J. Cheng, “A fault tolerant integrated navigation scheme realized through online tuning of weighting factors for Federated Kalman Filter,” *Applied Mechanics and Materials*, vol. 446–447, pp. 1078–1085, 2014.
- [18] D. B. Fogel, “Introduction to simulated evolutionary optimization,” *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 3–14, 1994.
- [19] A. F. Tappenden and J. Miller, “A novel evolutionary approach for adaptive random testing,” *IEEE Transactions on Reliability*, vol. 58, no. 4, pp. 619–633, 2009.
- [20] M. L. M. Víctor, C. Vicent, L. Luis, and F. Anta Antonio, “A model of self-avoiding random walks for searching complex networks,” *Networks*, vol. 60, no. 2, pp. 71–85, 2012.
- [21] J. Du and R. Rada, “Dilemmas in knowledge-based evolutionary computation for financial investing,” *Intelligent Decision Technologies*, vol. 7, no. 2, pp. 123–136, 2013.
- [22] M. Mahsal Khan, A. Masood Ahmad, G. Muhammad Khan, and J. F. Miller, “Fast learning neural networks using Cartesian genetic programming,” *Neurocomputing*, vol. 121, pp. 274–289, 2013.
- [23] C. Xilin, “Neural network combined with evolutionary algorithm for Knowledge management in Electricity Supply Industry,” in *Proceedings of the IEEE International Symposium on Test and Measurement (ICTM '09)*, vol. 2, pp. 355–358, Hong Kong, 2009.
- [24] N. Mehdiyev and D. Enke, “Interest rate prediction: a neuro-hybrid approach with data preprocessing,” *International Journal of General Systems*, vol. 43, no. 5, pp. 535–550, 2014.
- [25] F. Zha, B.-Q. Hu, and J. Liu, “Prediction of gyro motor’s state based on grey theory and BP neural network,” *Journal of Chinese Inertial Technology*, vol. 18, no. 1, pp. 120–123, 2010 (Chinese).
- [26] G. Jing, W. Du, and Y. Guo, “Studies on prediction of separation percent in electro dialysis process via BP neural networks and improved BP algorithms,” *Desalination*, vol. 291, pp. 78–93, 2012.
- [27] Z. Haodong and L. Hongchan, “Optimized BP neural network model based on niche genetic algorithm,” in *Proceedings of the 2nd International Conference on Green Communications and Networks*, vol. 223 of *Lecture Notes in Electrical Engineering*, pp. 219–226, Springer, 2013.
- [28] L. Tian, Y. Luo, and Y. Wang, “Prediction model of TIG welding seam size based on BP neural network optimized by genetic algorithm,” *Journal of Shanghai Jiaotong University*, vol. 47, no. 11, pp. 1690–1696, 1701, 2013 (Chinese).
- [29] H. Yang, R. S. Fan, and D. H. Xu, “Power information system risk assessment method based on genetic algorithms and neural network,” *Applied Mechanics and Materials*, vol. 530–531, pp. 429–433, 2014.
- [30] H. Wei, *Theory and Method of the Neural Network Structure Design*, National Defence of Industry Press, Beijing, China, 2005.
- [31] P.-J. Li, X.-S. Xu, L.-H. Wang, and Y. Li, “Design of intelligent fault-tolerant to passive underwater integrated navigation system,” *Journal of Chinese Inertial Technology*, vol. 21, no. 2, pp. 221–225, 2013 (Chinese).

Research Article

Manifold Learning with Self-Organizing Mapping for Feature Extraction of Nonlinear Faults in Rotating Machinery

Lin Liang,^{1,2} Fei Liu,¹ Maolin Li,³ and Guanghua Xu^{1,4}

¹School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China

²Key Laboratory of Education Ministry for Modern Design and Rotor-Bearing System, Xi'an Jiaotong University, Xi'an 710049, China

³Engineering Workshop, Xi'an Jiaotong University, Xi'an 710049, China

⁴State Key Laboratory for Manufacturing Systems Engineering, Xi'an Jiaotong University, Xi'an 710049, China

Correspondence should be addressed to Lin Liang; lianglin@mail.xjtu.edu.cn

Received 19 September 2014; Revised 27 December 2014; Accepted 4 January 2015

Academic Editor: Saeed Balochian

Copyright © 2015 Lin 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.

A new method for extracting the low-dimensional feature automatically with self-organization mapping manifold is proposed for the detection of rotating mechanical nonlinear faults (such as rubbing, pedestal looseness). Under the phase space reconstructed by single vibration signal, the self-organization mapping (SOM) with expectation maximization iteration algorithm is used to divide the local neighborhoods adaptively without manual intervention. After that, the local tangent space alignment algorithm is adopted to compress the high-dimensional phase space into low-dimensional feature space. The proposed method takes advantages of the manifold learning in low-dimensional feature extraction and adaptive neighborhood construction of SOM and can extract intrinsic fault features of interest in two dimensional projection space. To evaluate the performance of the proposed method, the Lorenz system was simulated and rotation machinery with nonlinear faults was obtained for test purposes. Compared with the holospectrum approaches, the results reveal that the proposed method is superior in identifying faults and effective for rotating machinery condition monitoring.

1. Introduction

Rotating machinery covers a wide range of mechanical equipment and is of importance in industrial applications. Therefore, faults in rotating machinery may severely affect operations in industry and even safety. To minimize the number of breakdowns as well as to increase the reliability, rotating machinery condition should be monitored for symptoms and incipient fault detection. By this, the life of machinery could be prolonged and the catastrophic consequences of unplanned failure could be avoided. Traditionally, to monitor the conditions and diagnose the faults of rotating machinery, vibration signals are most selected due to its easy-to-measure characteristics and analysis [1–4]. The process technologies of vibration signals in common use are frequency spectrum, axis center orbit, time-frequency analysis, and so on. However, for the nonlinear faults of rotating machinery, such as rubbing,

pedestal looseness fluid, and so forth, there are still some problems, for example, the amplitude and phase information of vibration signals are separated from each other, and the correlation of the vibration signals on the vertical and horizontal direction cannot be obtained. Hence, it is difficult to recognize the panorama of the rotor's vibration.

To overcome the shortcomings of the traditional methods, the holospectrum was put forth for synthesizing the information of the phase, amplitude, and frequency [5]. As an effective fault diagnosis technology for the rotation machinery, it takes advantage of the improved Fourier transform algorithm to analyze the vibration displacement signals from one measuring section which can provide much more information about rotor vibration behavior. However, it needs artificial experience for better result. In case of holospectrum, it is unavoidable to select proper harmonic components for correct judgment. For example, for misalignment fault, the

orbit of axis center of rotor is synthesized by 1x, 2x, and 4x frequency components, while the axis center orbit for oil whirl is built by 0.45x frequency component. That is to say, that the method failed to deal with the vibration data without manual intervention.

Due to instantaneous variations in friction, damping, and load, the mechanical systems are often characterized by nonlinear behaviors. Therefore, nonlinear analysis methods provide a good choice to extract defect-related features hidden in the measured signals, which may not be effectively identified using the conventional methods. Many nonlinear methods, such as correlation dimension, Lyapunov exponent, and approximate entropy [6], have been investigated. These methods are suitable to reveal the variations of the dynamical system where it is in the noise-free or low noise conditions. In fact, the vibration signals obtained from the mechanical system are inevitably contaminated by noise. Thus the above methods are conducted by averaging all points in the embedding space, and this may lose significant information about the time domain.

As a new dimension reduction technique, manifold learning methods have emerged in nonlinear research fields to identify meaningful low-dimensional structures hidden in high-dimensional observations, such as locally linear embedding [7], isometric feature mapping [8], and local tangent space alignment [9]. These methods have been applied in computer vision, document analysis, and fault diagnosis [10–12]. Yang et al. [13] proposed a method for nonlinear time series noise reduction based on principal manifold learning applied to the analysis of gearbox vibration signal with tooth broken. Li et al. [14] proposed the multiple manifolds analysis approach to extract manifold information from the bearing vibration signals with different faults. As for rotor systems, Jiang et al. [15] recently proposed the supervised manifold learning algorithm for effective feature extraction. Based on the survey of methods above, it is found that the manifold learning is an effective method for feature extraction. However, the features are extracted usually in uniform distribution of sample data, ignoring the influence of neighborhood size.

Obviously, neighborhood of high dimension constructed with vibration signal can not ensure uniform distribution. Same neighborhoods size can falsely estimate the relationships between the neighbors; it is therefore worthy of considering variable number of neighbors that are adaptively chosen. In order to distinguish the nonlinear fault of rotating machinery with vibration signals, a new low-dimensional embedding extraction method based on the local tangent space alignment combined with self-organization mapping is proposed. The main advantages of the approach, compared with other nonlinear analysis methods, are as follows: vibration signals are embedded into a high-dimensional space, which is more effective to discover the essential characteristics of the dynamical system, and it can distinguish the type of faults with less manual intervention. In a word, the new approach extracts the low-dimensional embedding from the manifolds to reflect the states of the mechanical system rather than extract a feature by averaging all points with the time waveform.

The organization of the rest paper is given as follows: a brief introduction of manifold learning with self-organizing mapping is given in Section 2. In Section 3, the details of feature extraction scheme are proposed. And Section 4 applies the method in detection for nonlinear fault of rotating machine. Finally, conclusions are drawn in Section 5.

2. Manifold Learning with Self-Organizing Mapping

2.1. Adaptive Selection of Neighborhood. Obviously, large neighborhoods cause confusions when dealing with the highly twisted manifold. In contrast, small neighborhoods can falsely estimate the relationships between the neighbors. Thinking to added noise, the distribution of samples in feature space is usually nonuniform. Thus, the fixed sizes of neighborhoods cannot satisfy the changing manifold structures. It is inevitable that the neighborhood size should be selected adaptively with the principle that all of subspaces should be connected to construct the topology structure of manifold. Meanwhile, there should be enough overlaps between adjacent neighbors, in order to transmit the local information.

From the view of network, self-organizing mapping (SOM) has the ability to divide nodes adaptively. Using competing-layer neurons to match the center of local neighbors of manifold structures, node grids are organized to cover the topological structures. Then with the learning of SOM, the local neighbors of high-dimensional manifolds are divided adaptively.

2.2. Self-Organizing Mapping with EM. A SOM is a type of artificial neural network that is trained using unsupervised learning to produce a low-dimensional mapping space, discretized representation of the input space of training samples, and a self-organizing mapping consists of components called nodes. Associated with each node is a weight vector of the same dimension as the input data vectors, and a position in the mapping space.

Let p_r^i denote the probability that input x_i is assigned to the node with weight w_r . It is constrained by $\sum_r p_r^i = 1$ and $p_r^i > 0$. There is a neighborhood function $h_{r,s}$ that corresponds to the control strength between node r and node s . Usually, it is a decreasing function of the distance between nodes r and s . Given the data \mathbf{X} , the optimal goal is to find the probability assignments \mathbf{P} and weights \mathbf{W} that minimizes

$$F_q(\mathbf{P}, \mathbf{W}) = \sum_i \sum_r p_r^i \sum_s h_{r,s} \mathbf{D}(x_i, \mathbf{w}_s), \quad (1)$$

where $d(r, s)$ is the distance between nodes of network and \mathbf{D} is the distance between inputs \mathbf{X} and weights \mathbf{W} .

The closer the distance between the nodes, the smaller $h_{r,s}$ value. So the logarithm likelihood function is defined as follows:

$$F_\varepsilon(P) = \sum_i \sum_r p_r^i \log \left[\frac{p_r^i}{q_r} \right], \quad (2)$$

where q_r is priori probability distribution and its initial value is usually set to uniform distribution. Then plusing (1) into (2), the free energy function is now the following:

$$F(\mathbf{P}, \mathbf{W}) = F_q(\mathbf{P}, \mathbf{W}) + F_\varepsilon(\mathbf{P}). \quad (3)$$

To minimize $F(\mathbf{P}, \mathbf{W})$ of constraint conditions, the implementation process of SOM can be viewed as in the condition of known data \mathbf{X} , seeking \mathbf{P} and \mathbf{W} posterior distribution density function. Obviously, it is suitable to select neighborhood with EM iteration [16]. Therefore, the neighborhood selection algorithm is as follows.

- (1) Calculate the neighborhood matrix of topology network in initial output layer $\mathbf{H} = [h_{s1}, \dots, h_{sr}]$ and normalize it. Neighborhood function is given by

$$h_{r,s} = \exp\left(\frac{-d(r,s)^2}{2\sigma^2}\right), \quad (4)$$

where $d(r,s)$ is the distance between nodes of network. Set the initial value: $s = 1, 2, \dots, r$ and $\sigma = r$; thus the sum of the relative entropy in initial competitive layer network is defined as

$$e_i = \sum_{p=1}^q -h_{ip} * \log h_{ip}. \quad (5)$$

The initial weight matrix \mathbf{W} is given by $\mathbf{W} = \text{diag}((\sum_q \mathbf{Q}^{-1}) * \mathbf{Q} * \mathbf{X})$, where $\mathbf{Q} = [q_1, \dots, q_r]$ and q_i is a random value between 1 and r .

- (2) The location coordinate of topology node is set to the element of weight \mathbf{W} . Then function \mathbf{F} is calculated as

$$\mathbf{F} = -\frac{(\sum_1^r \mathbf{D} \cdot \mathbf{Q}')}{2\sigma} + e'_i(\mathbf{W}e) - \mathbf{D} * \frac{\log(\sigma)}{2} \quad (6)$$

$$= [f_1, f_2, \dots, f_N],$$

where $\sigma = \sum_{i=1}^N \min(d_i)$ and e is the sum of relative entropy in competition layer network. \mathbf{D} is the distance between inputs \mathbf{X} and weights \mathbf{W} , where $\mathbf{D} = \|\mathbf{X} - \mathbf{W}\| = [d_1, d_2, \dots, d_q]$. The global optimal unit is $d_j = \min(d_1, d_2, \dots, d_r)$ in which j is set to the winner neurons index for input x_i .

- (3) With the iterate minimal, \mathbf{F} is found by

$$f_{i\text{new}} = \begin{cases} f_i & i = j \\ -\frac{(\sum_1^r D(i) \cdot h(i))}{2\sigma} + e'_i(i) - \mathbf{D} * \frac{\log(\sigma)}{2} & i \neq j. \end{cases} \quad (7)$$

The nodes which are greater than f_i can be found in $f_{i\text{new}}$ and are labeled with v . So the node distribution is adjusted with

$$q_{\text{new}} = v. \quad (8)$$

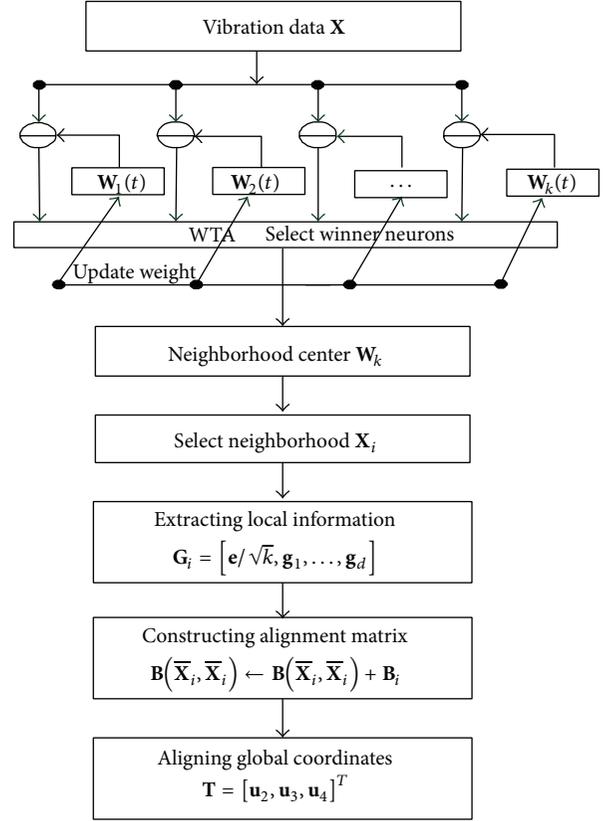


FIGURE 1: Schematic diagram of the proposed manifold learning with SOM method.

- (4) Set $h_{\text{new}} = h * \text{rate}$, in which rate is learning ratio. Then new e , \mathbf{Q} , \mathbf{W} , \mathbf{F} , and σ are calculated, and return to Step (3) until the elements in the \mathbf{H} are big enough. Finally, with the above iteration, the weight \mathbf{W} is calculated and then fixed in the maximization for the new value.

2.3. *Manifold Learning with SOM.* Manifold learning aims at discovering the intrinsic structure of nonlinear data. The process of the manifold learning with SOM is shown in Figure 1, and the implementation procedures are detailed as follows.

- (1) Given a set of inputs \mathbf{X} , the SOM network is adopted to optimize the weights \mathbf{W} . Including multifrequency components or noise in vibration signals, obviously, the performance of the trajectory in phase space reconstruction is complex. Therefore, to balance the calculation and efficiency, network size is usually set to a larger scale. Meanwhile, to keep the consistency optimal results, initial weight of \mathbf{W} can be set to unit matrix, and the learning ratio of \mathbf{H} is also set to 1.1 for a gradual learning process.
- (2) Selecting neighborhood adaptively: each element of \mathbf{W} is used to set the center node of local neighbors. To ensure enough overlap, the radius of neighbor is equal to the half of maximal distance between center nodes. According to radius of neighbor, the local neighbors

\mathbf{X}_i are selected, where $i = 1, 2, \dots, k$ and k is the number of topology grids.

- (3) Extracting local information: compute the d largest eigenvectors g_1, \dots, g_d of the correlation matrix $(\mathbf{X}_i - \bar{\mathbf{X}}_i \mathbf{e}^T)^T (\mathbf{X}_i - \bar{\mathbf{X}}_i \mathbf{e}^T)$, and set $\mathbf{G}_i = [\mathbf{e}/\sqrt{k}, \mathbf{g}_1, \dots, \mathbf{g}_d]$, where $\bar{\mathbf{X}}_i$ is the mean of \mathbf{X}_i .
- (4) Constructing alignment matrix: form the matrix \mathbf{B} by locally summing $\mathbf{B}(\bar{\mathbf{X}}_i, \bar{\mathbf{X}}_i) \leftarrow \mathbf{B}(\bar{\mathbf{X}}_i, \bar{\mathbf{X}}_i) + \mathbf{B}_i$ if a direct eigensolver will be used. Otherwise implement a routine that computes matrix-vector multiplication \mathbf{B} for an arbitrary vector u .
- (5) Aligning global coordinates: compute the $d + 1$ smallest eigenvectors of \mathbf{B} and pick up the eigenvector matrix $(u_2, u_3, \dots, u_{d+1})$ corresponding to the 2nd to $d + 1$ st smallest eigenvalues, and set the global coordinates $\mathbf{T} = (u_2, u_3, \dots, u_{d+1})^T$. With the global coordinates, the feature can be reflected in low-dimension spaces \mathbf{T} .

From the viewpoint of geometry, the vibration data of the same operation state of rotor system has the same geometric property in space distribution or topological structure, its mapping points in the low-dimension embedding space can be distributed in embedded manifolds or in its neighbor. However the embedding dimension d of manifold has relation with

$$d = m_{\min}, \quad (9)$$

where m_{\min} is the minimum embedding of dynamics system and is estimated with Cao algorithm [17].

Based on the adaptive neighborhood selection, the low-dimensional embedding can be extracted effectively. Since node neighborhood is divided adaptively by the grid of network for competition mechanism, SOM can overcome the limitation of fixed neighborhood algorithms.

3. Schematic Diagram of Feature Extraction

3.1. Manifold Reconstruction from Vibration Signal. The state space is constructed by a set of basis vectors which are composed of the dynamic variables of a system. But most commonly, not all the dynamic variables of the system are accessible for measure, an alternative form known as embedded phase space is convenient for research of the dynamics of the system. Suppose measurements obtained through sampling can be defined by x_n . In order to reconstruct a manifold from a time series, time delay embedding [18] is employed. For a time series $\mathbf{X} = [x_1, x_2, \dots, x_N]$, the delay vectors in the embedded phase space are given as

$$\mathbf{Y}_i = [x_i, x_{i+\tau}, \dots, x_{i+(m-1)\tau}], \quad (10)$$

where $i = 1, 2, \dots, N$, m is the embedding dimension and τ is the delay time.

Roughly speaking, m -dimensional delay embedding space is equivalent to the original observed state space of the dynamical system. Taking the steady-state response of

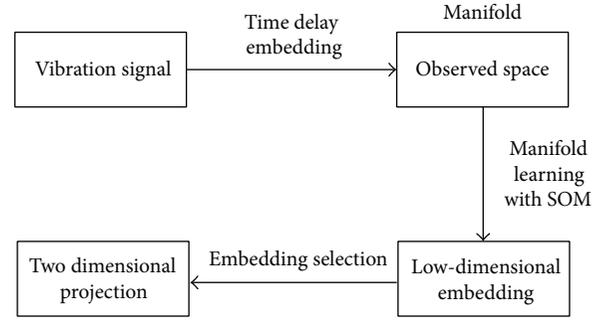


FIGURE 2: Schematic diagram of feature extraction strategy.

the variable x and performing the time delay embedding transformation lead to trajectories in the embedding space that comprises a geometric manifold object. Takens has shown that [19], using delay coordinates and the embedding procedure, a manifold can be reconstructed by embedding a time series into a high-dimensional space, in which the topological structure and nonlinear characteristics hidden in the one-dimensional time series can be easily extracted. Thus, reconstructing a manifold in a high-dimensional space is the key to the proposed approach in this paper.

However, the selection of time delay and embedding dimension in the phase space reconstruction is a question. Except for uniform time delay, the nonuniform time delay is also used to build the phase space [20]. But it is well known that even nonuniform time delay also cannot guarantee that the time delay vectors distribution is uniform. So, in order to simplify the problem, the Hankel matrix is adopted in this paper to represent the high-dimensional space, where $\tau = 1$ and embedding dimension m is large enough. Namely, if $m \gg m_{\min}$ where m_{\min} is the minimum embedding, it is possible to reconstruct the underlying dynamic structure of the entire system from the behavior of the single variable.

3.2. Schematic Diagram. For feature extraction in rotating machinery fault diagnosis, the manifold learning with SOM is adopted to explore the geometric distribution properties embedded in the high-dimensional space. On the basis of the principles above, a new approach of feature extraction method based on adaptive manifold learning is proposed. First, high-dimensional observation space is built with phase space reconstruction, and then map the space phase data into a feature space, and estimate the intrinsic distribution of samples to gain the embedding manifold structure. Finally, the feature is represented by two dimensional projections for the sake of intuitive analyses of equipment operating status. The schematic diagram of the feature extraction method based on manifold learning is shown in Figure 2.

In application, it is should be noted that the holospectrum is drawn from the different harmonic components depending on manual intervention. Instead of by frequency component selection, signal is directly used to construct the dynamic trajectory with phase space reconstruction, and then through adaptive neighborhood selection strategy, embedding low-dimensional manifold can be extracted, therefore, reducing the dependence on human experience.

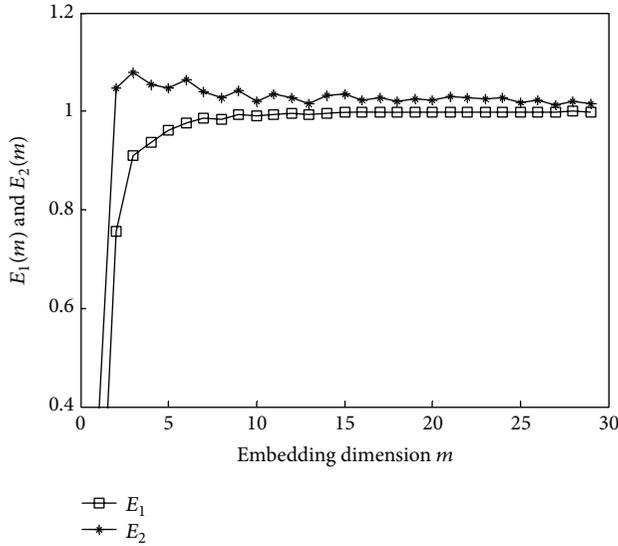


FIGURE 3: $E_1(m)$ and $E_2(m)$ curve of the Lorenz system.

4. Experiments Verifications

To verify the capability of feature extraction of the proposed method, the nonlinear Lorenz system was adopted for test and is described as

$$\begin{aligned} \frac{dx}{dt} &= \alpha(y - x), \\ \frac{dy}{dt} &= (\gamma - z)x - y, \\ \frac{dz}{dt} &= xy - \beta z, \end{aligned} \quad (11)$$

where the parameters $\alpha = 16$, $\beta = 4$, $\gamma = 45.92$, and the 8 dB white noise was also mixed. According to the evaluation index proposed in Cao algorithm, the $E_1(m)$ and $E_2(m)$ curve were calculated, respectively, where the time delay $\tau = 1$. The result is shown in Figure 3.

In manifold learning, the minimum embedding dimension is set to 3. The two dimensional projection of the phase space is shown in Figure 4, where embedding dimension m of Hankel matrix is set to 30. Due to the mixed noise, the dynamics trajectory is difficult to be identified. The two dimensional projection of Lorenz system extracted by the proposed manifold learning technique is shown in Figure 5. It is found that the embedding manifold structure is similar to the Lorenz system.

With the EM iteration, the neighborhood sizes learned from the SOM are shown in Figure 6. It is easy to figure out that the neighborhood sizes of nodes fluctuate are between 4 and 78. For the comparison, the adaptive neighborhood selection algorithm is also used to extract the low-dimension embedding using the neighborhood contraction and expansion [21]. The two dimensional projection is displayed in Figure 7. Comparing to Figure 5, it is noted that projection of the reconstructed phase space has a certain deform, because the coefficient selection of contraction and expansion of

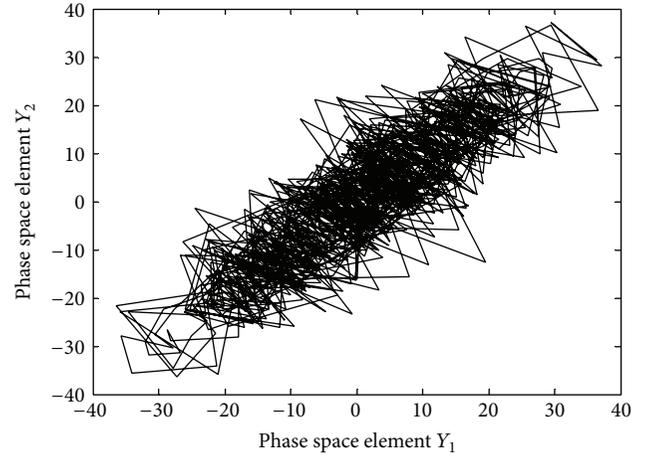


FIGURE 4: Phase space of Lorenz system with added noise.

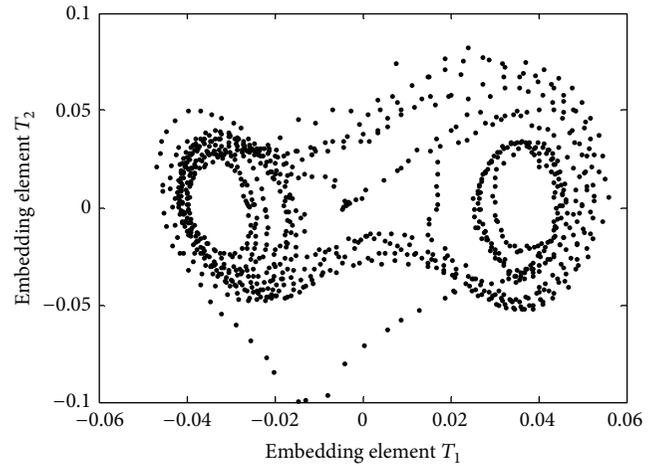


FIGURE 5: Embedding projection of manifold learning with SOM.

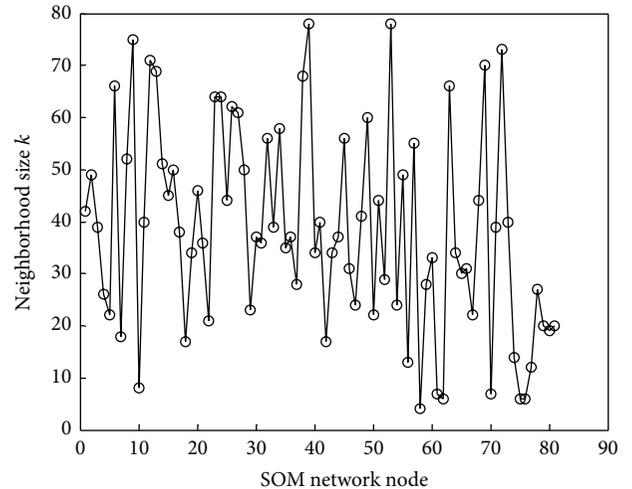


FIGURE 6: Neighborhood size learned from SOM.

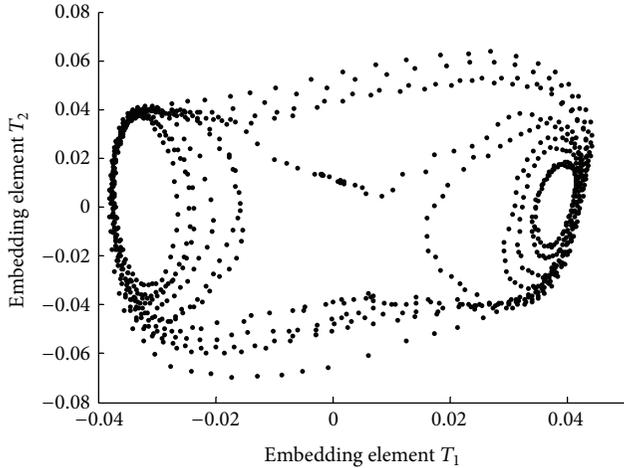


FIGURE 7: Embedding projection of manifold learning with adaptive neighborhood selection.

neighborhood depend on artificial experience. Due to the fewer parameters selection, it clearly shows SOM technique has the better ability in extracting features.

5. Application

The proposed method is applied to feature extraction of rotating mechanical nonlinear faults. Firstly, the fluid excitation failure in a N2 compressor high-pressure cylinder of petrochemical plant is adopted to extract feature. Usually, for the normal operation of rotation machinery, due to the laminar flow state of fluid medium through flow of rotor, the vibration of machinery is smaller. However, inappropriate adjustment of process parameters can lead to the steady turbulent flow phenomenon, resulting in the impact on the rotor. The compressor rotating speed is 11 416 r/min. Displacement transducers were used to acquire vibration signals of the rotor at the corresponding measurement points on coupling end. The sampling frequency is 2 000 Hz, and vibration waveform and frequency spectrum are shown in Figures 8 and 9, respectively. In Figure 9, except for the rotation frequency 190.3 Hz, there are large numbers of low frequency noise components. Therefore, the original axis center orbit plotted in Figure 10 is submerged. In order to eliminate the noise, the band-pass filter with phase preserving is used, and the corresponding filtering axis center orbit is also shown in Figure 11. According to the mechanism of fluid excitation fault, the typical performance of axis center orbit is unstable ellipse. However, due to its worse performance of filtering axis center orbit for fluid excitation fault, it is not easy to identify the fault pattern accurately.

As a comparison, the proposed method is adopted to extract embedding manifold from the collected vibration data. The $E_1(m)$ and $E_2(m)$ curve were calculated and shown in Figure 12, and the minimum embedding dimension is set to 6. In corresponding 6 embedding vectors extracted with $m = 30$, there are three kinds of frequency spectrum structure shown in Figures 13–15. It is clear that, comparing with

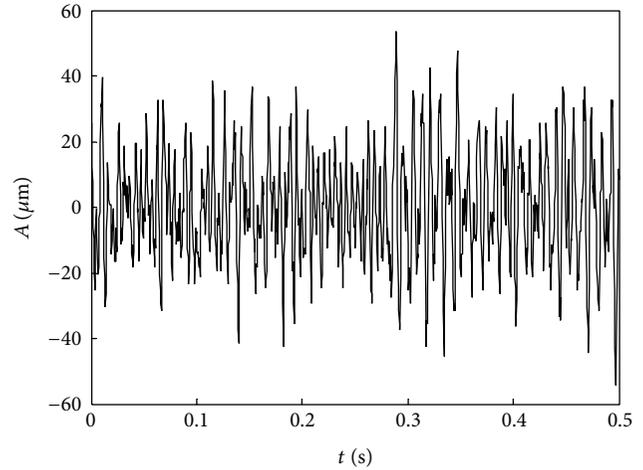


FIGURE 8: Vibration waveform of fluid excitation fault.

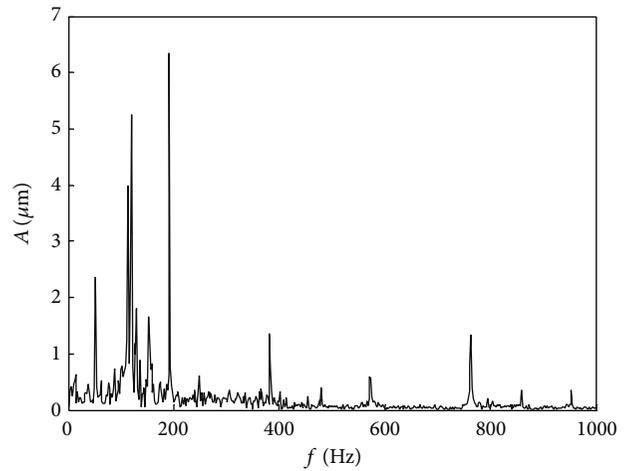


FIGURE 9: Frequency spectrum of fluid excitation fault.

Figure 9, the rotation frequency and low frequency noise components are separated effectively. Therefore the projection space of embedding manifold corresponding rotation frequency is displayed in Figure 16.

It is very convenient to identify that, due to fluid excitation through flow of rotor, the projection trajectory are an unstable ellipse trace characteristic that is different from other faults. So, according to the special curve of the embedded manifold, the fault can be identified effectively. Obviously, the axis center orbit shown in Figure 11 has a certain limitation to identify this fault.

In order to verify the capability of the proposed method, different neighborhood sizes were adopted by LTSA to extract the low-dimensional embedding, where the neighborhood size k is set to 10 and 30, respectively. The experimental results with different k are shown in Figures 17 and 18. It is clear that the projection trajectory of embedding manifold with fixed neighborhood size cannot reflect the feature structure of fluid excitation fault, and the projection results are not as good as proposed method displayed in Figure 16.

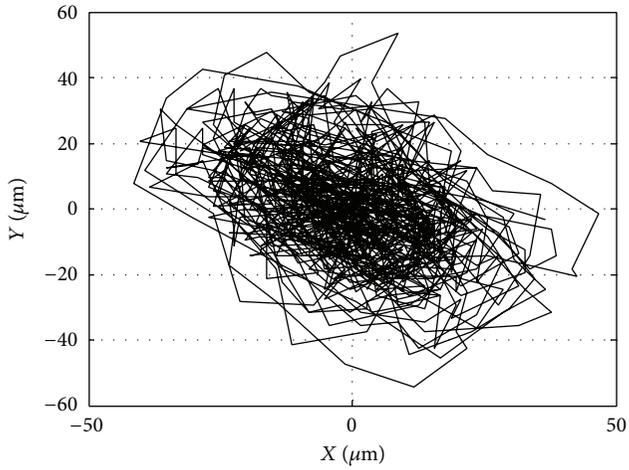


FIGURE 10: Raw axis center orbit of fluid excitation fault.

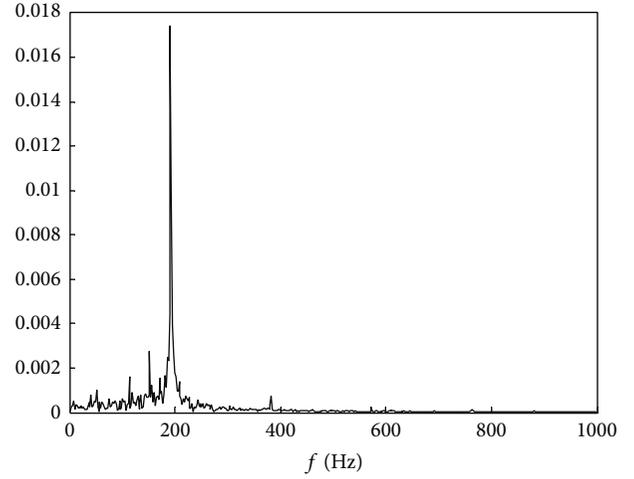


FIGURE 13: Frequency spectrum of the first embedding.

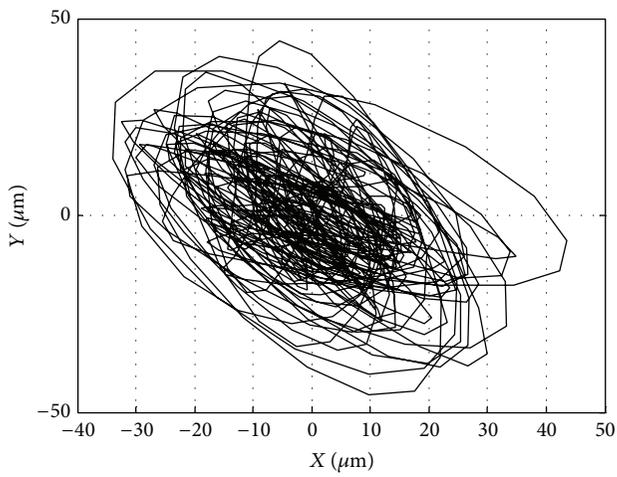


FIGURE 11: Filtering axis center orbit of fluid excitation fault.

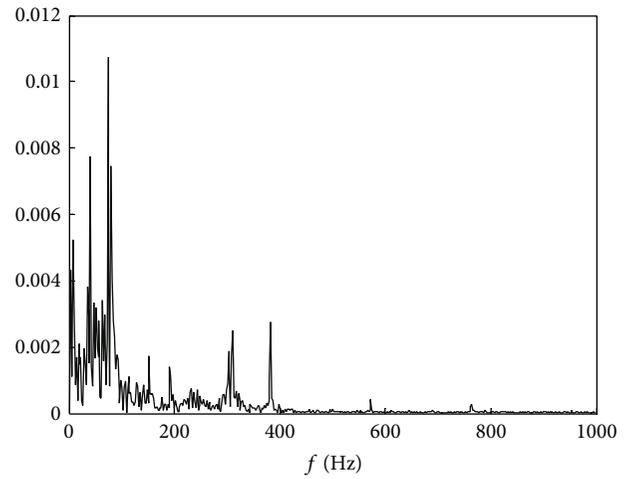


FIGURE 14: Frequency spectrum of the third embedding.

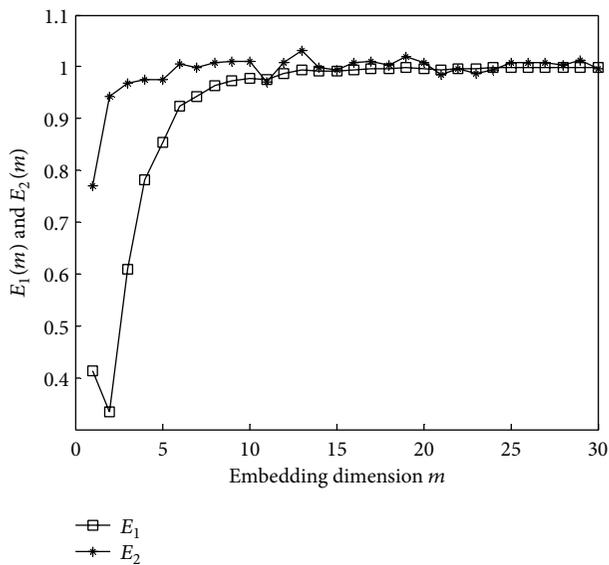


FIGURE 12: $E_1(m)$ and $E_2(m)$ curve of fluid excitation fault.

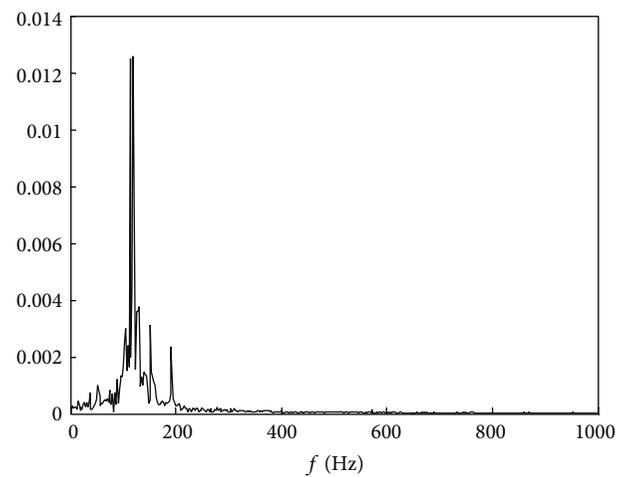


FIGURE 15: Frequency spectrum of the fifth embedding.

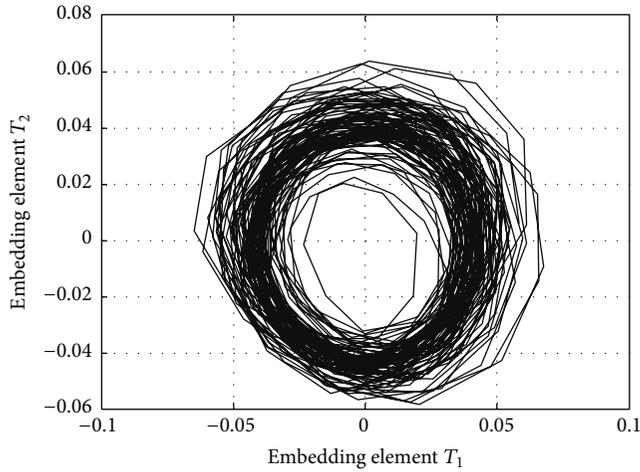


FIGURE 16: Embedding projection of manifold learning with SOM.

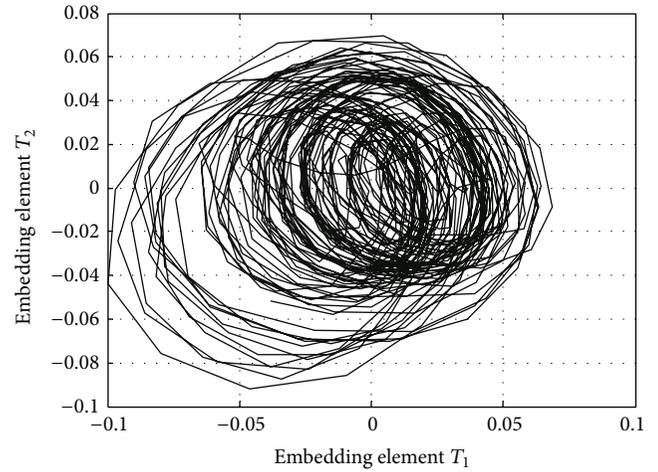
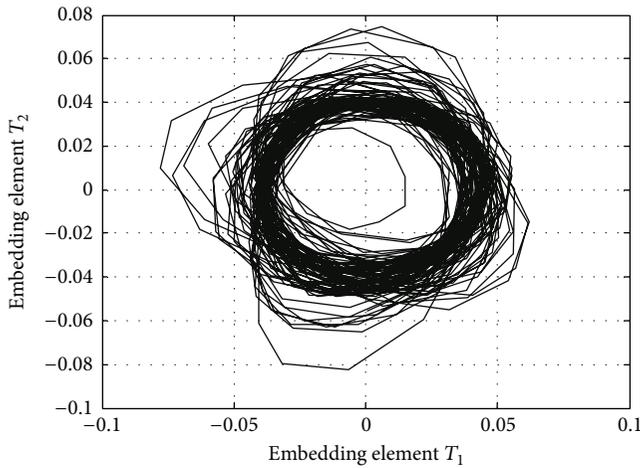
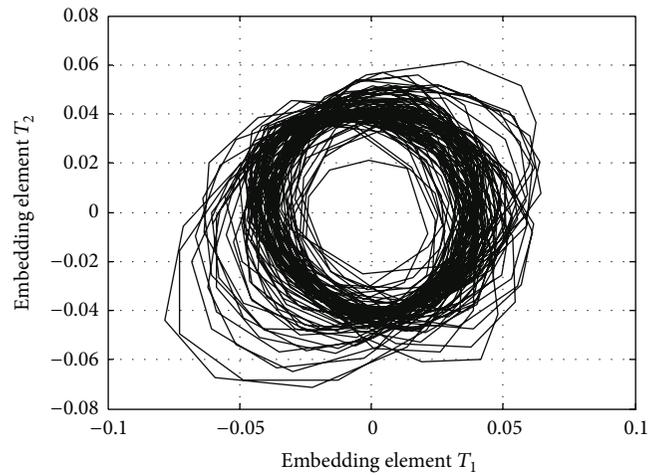
FIGURE 18: Embedding projection with LTSA in $k = 30$.FIGURE 17: Embedding projection with LTSA in $k = 10$.

FIGURE 19: Projection of embedding manifold with adaptive neighborhood selection.

Obviously, the SOM neighborhood selection is more effective than the fixed neighborhood size. In addition, the other adaptive neighborhoods selection in manifold learning is also used to extract the low-dimension embedding, and the two dimensional projection is displayed in Figure 19. It is noted that although projection effect of the low-dimension embedding is better than fixed neighborhood size, it is not good as the proposed method.

Next, the rotor-stator rub fault of turbines in refinery plant is also adopted to be analyzed. It should be noted that, due to the motion interference between stator and rotor, the axis orbit of rub fault is extremely complex and often shows a sudden change in axis center orbit. However, considering the influence of noise, only in the case of large fault degree, the characteristics can be observed effectively. Figures 20 and 21 are the waveform and frequency spectrum of rotor-stator rub, and corresponding purified trace is shown in Figure 22 where turbine rotating speed is 446 r/min and the sampling frequency is 200 Hz. Generally, each frequency

component in frequency spectrum corresponds to a sinusoidal component. So different faults can be represented with eight major harmonic frequency components and four low frequency components, where these frequency components are usually extracted for diagnosis purpose. Thus, the rotation frequency and its harmonic components can be selected to reconstruct time domain waveform via inverse Fourier transform, and the purified axis center orbit also is drawn from the reconstructed signals in measuring section. The main purpose is to eliminate the noise of the original axis center orbit. As seen from Figure 22, it is difficult to identify fault through waveform or axis center orbit.

According to the evaluation, the minimum embedding dimension is set to 4. In corresponding four embedding vectors, there are two kinds of frequency spectrum structure shown in Figure 23. It is clear that the rotation frequency and its harmonic components are extracted in different embedding. So the projection of embedding manifold corresponding rotation frequency and high frequency is displayed

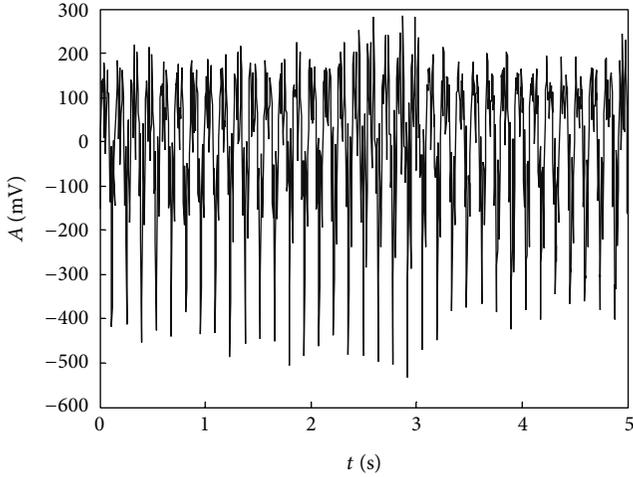


FIGURE 20: Waveform of rotor-stator rub.

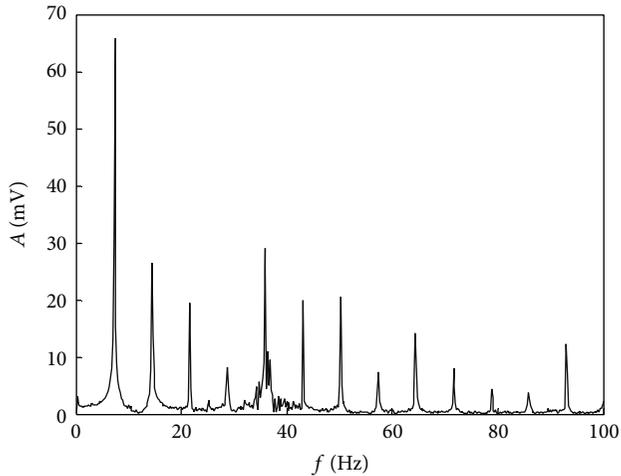


FIGURE 21: Frequency spectrum of rotor-stator rub.

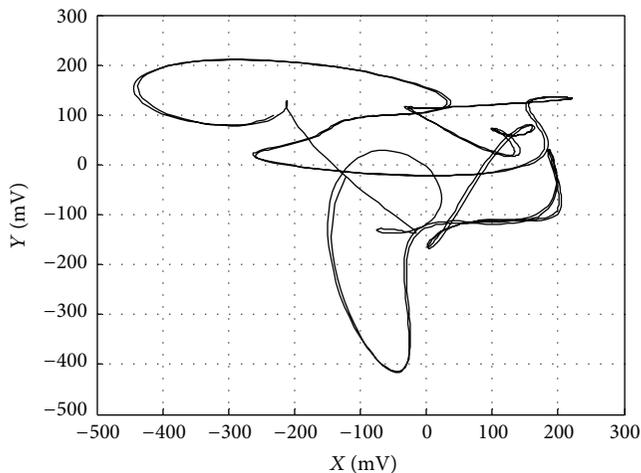
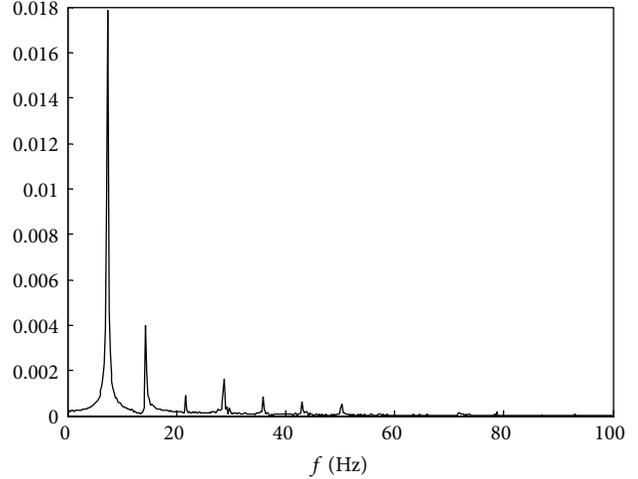
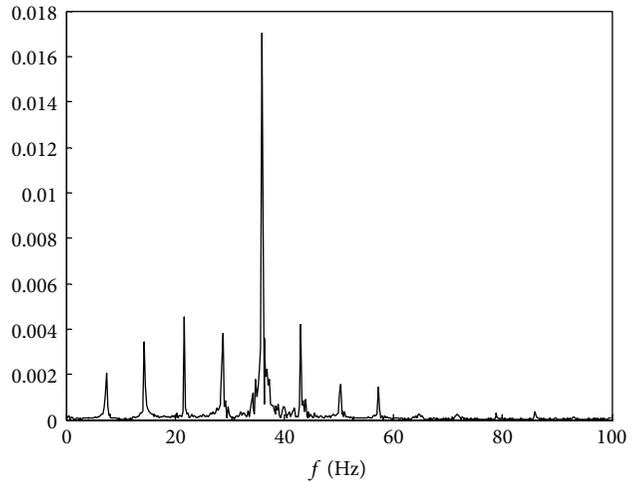


FIGURE 22: Purified axis center orbit.



(a) Frequency spectrum of the first embedding



(b) Frequency spectrum of the third embedding

FIGURE 23: Frequency spectrum of embedding manifold extracted with proposed method.

in Figure 24. It can be obviously seen that, due to the influence of the high frequency components, there are some sudden changes in two projection trajectory. Comparing the purified axis center orbit, in the projection shown in Figure 24, the change of trajectory very clearly reflects the characteristics of the rub fault. Obviously, as a typical nonlinear fault, it can be identified in the low embedding space effectively.

In order to further test the effectiveness of the proposed method, the fixed neighborhood size was adopted by LTSA to extract the low-dimensional embedding, where k is set to 20. The experimental result is shown in Figures 25 and 26 and is the corresponding frequency spectrum of embedding manifold. Obviously, due to the small neighborhood size, the high frequency components are filtered in embedding as shown in Figure 26; therefore, the sudden change behavior cannot be reflected in the trajectory of low-dimension embedding. It is clear that the method with fixed neighborhood size cannot extract effective low-dimensional embedding; thus identification of fault is seriously affected by the worst projection orbit.

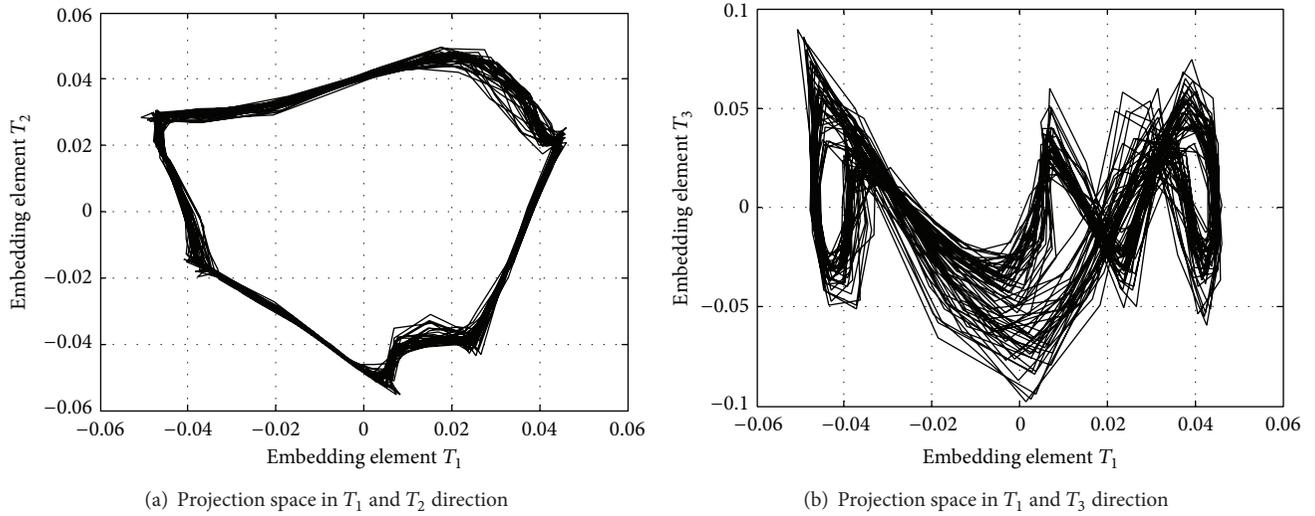


FIGURE 24: Projection of embedding manifold of rotor-stator rub.

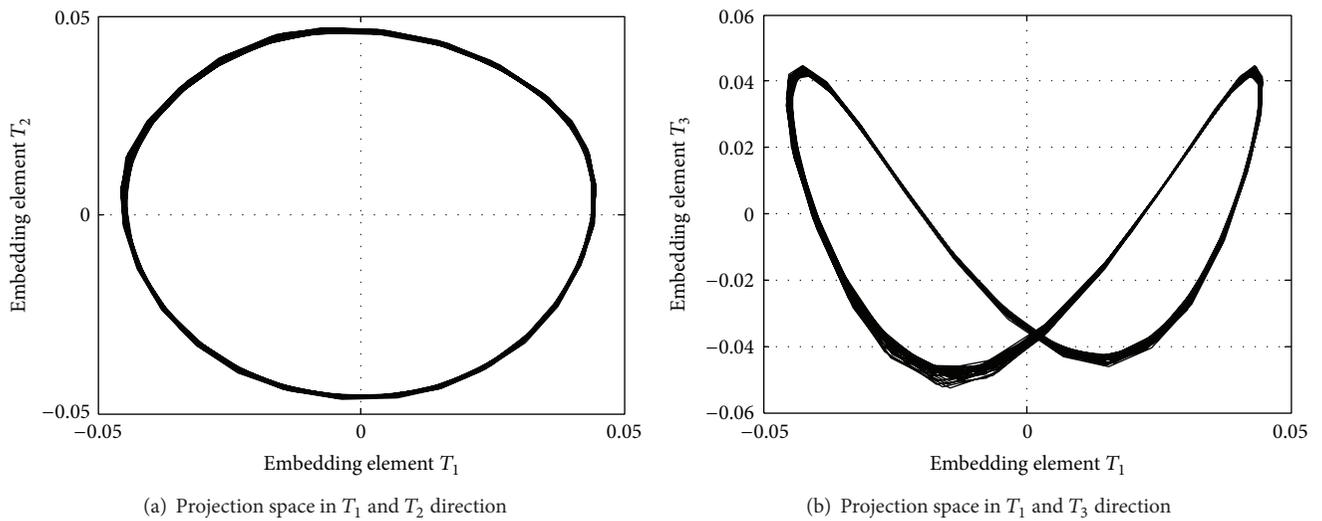
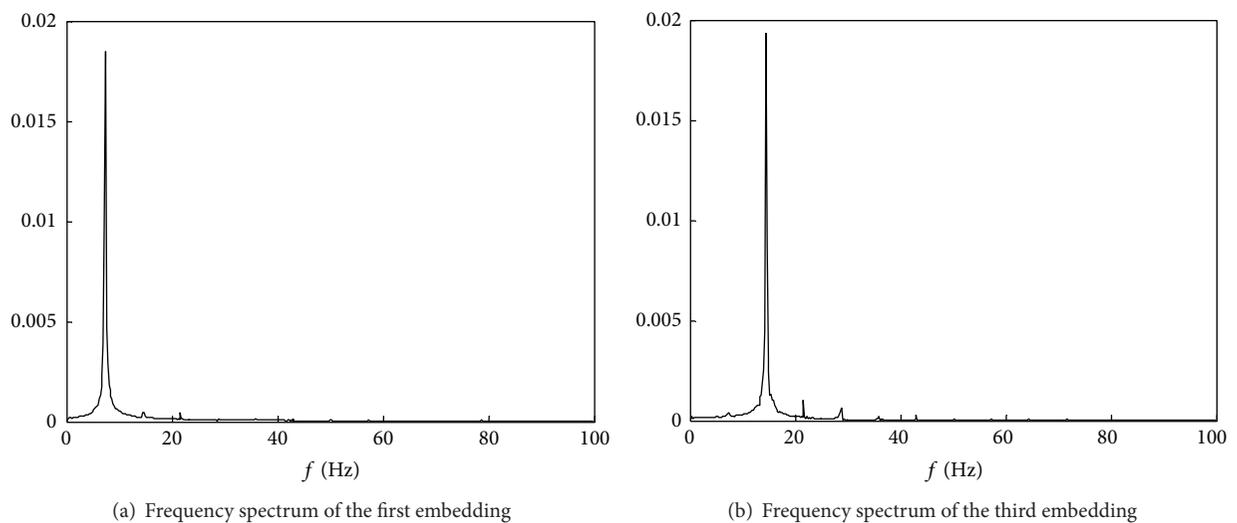
FIGURE 25: Projection of embedding manifold with neighborhood size $k = 20$.

FIGURE 26: Frequency spectrum of embedding manifold extracted with fixed neighborhood size.

6. Conclusions

In this paper, a novel method for feature extraction of rotating machinery based on nonlinear manifold learning with neighborhood selection adaptive is proposed. In order to detect the nonlinear faults with less manual intervention, a single-signal phase space reconstruction is adopted to construct the high-dimensional manifold, and the embedding is extracted by manifold learning with SOM.

The proposed method is applied to nonlinear system simulation and vibration data of rotating machinery collected with different nonlinear faults. The experimental results illustrate that the new method is superior to the holospectrum methods in fault diagnosis. Even though the behavior of faults is nonlinear and complicated, the manifold method with the adaptive neighborhood selection has demonstrated its reliability in extracting fault features that are not accessible by axis center orbit.

It should be noted that the existing feature extraction approaches in literatures also use holospectrum and axis center orbits, but they are unavoidable to select proper harmonic components for correct judgment. However, the proposed method can distinguish the faults with less manual intervention. Therefore, it has higher accuracy in fault diagnosis than the traditional methods.

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 Shaanxi overall innovation project of science and technology with Grant no. 2013KTCQ01-06.

References

- [1] Q. Hu, Z. J. He, Z. S. Zhang, and Y. Y. Zi, "Fault diagnosis of rotating machinery based on improved wavelet package transform and SVMs ensemble," *Mechanical Systems and Signal Processing*, vol. 21, no. 2, pp. 688–705, 2007.
- [2] Z. J. He, J. Y. Zhao, Y. B. He, and Q. F. Meng, "Wavelet transform and multiresolution signal decomposition for machinery monitoring and diagnosis," in *Proceedings of the IEEE International Conference on Industrial Technology (ICIT '96)*, pp. 724–727, Shanghai, China, December 1996.
- [3] Z. X. Li, X. P. Yan, C. Q. Yuan, J. B. Zhao, and Z. X. Peng, "A new method of nonlinear feature extraction for multi-fault diagnosis of rotor systems," *Noise and Vibration Worldwide*, vol. 41, no. 10, pp. 29–37, 2010.
- [4] X. Xiong, S. X. Yang, and C. B. Gan, "A new procedure for extracting fault feature of multi-frequency signal from rotating machinery," *Mechanical Systems and Signal Processing*, vol. 32, pp. 306–319, 2012.
- [5] L. S. Qu and G. H. Xu, "One decade of holospectral technique: review and prospect," in *Proceedings of the ASME Design Engineering Technical Conferences (AETC '99)*, pp. 12–15, Las Vegas, Nev, USA, 1999.
- [6] B. C. Wang, Z. H. Ren, and B. C. Wen, "Fault diagnoses method of rotating machines based on nonlinear multi-parameters," *Journal of Mechanical Engineering*, vol. 48, no. 5, pp. 63–69, 2012.
- [7] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [8] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [9] Z. Y. Zhang and H. Y. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *Journal of Shanghai University*, vol. 8, no. 4, pp. 406–424, 2004.
- [10] J. Wang and Q. B. He, "Exchanged ridge demodulation of time-scale manifold for enhanced fault diagnosis of rotating machinery," *Journal of Sound and Vibration*, vol. 333, no. 11, pp. 2450–2464, 2014.
- [11] Q. B. He, "Vibration signal classification by wavelet packet energy flow manifold learning," *Journal of Sound and Vibration*, vol. 332, no. 7, pp. 1881–1894, 2013.
- [12] B. W. Li and Y. Zhang, "Supervised locally linear embedding projection (SLLEP) for machinery fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 25, no. 8, pp. 3125–3134, 2011.
- [13] J. H. Yang, J. W. Xu, D. B. Yang, and M. Li, "Noise reduction method for nonlinear time series based on principal manifold learning and its application to fault diagnosis," *Chinese Journal of Mechanical Engineering*, vol. 42, no. 8, pp. 154–158, 2006.
- [14] M. Li, J. W. Xu, J. H. Yang, D. B. Yang, and D. D. Wang, "Multiple manifolds analysis and its application to fault diagnosis," *Mechanical Systems and Signal Processing*, vol. 23, no. 8, pp. 2500–2509, 2009.
- [15] Q. S. Jiang, M. P. Jia, J. Z. Hu, and F. Y. Xu, "Method of fault pattern recognition based on laplacian eigenmaps," *Journal of System Simulation*, vol. 20, no. 20, pp. 5710–5713, 2008.
- [16] S. K. Ng, T. Krishnan, and G. J. McLachlan, "The EM algorithm," in *Handbook of Computational Statistics*, part 2, pp. 139–172, Springer, Berlin, Germany, 2012.
- [17] L. Y. Cao, "Practical method for determining the minimum embedding dimension of a scalar time series," *Physica D: Nonlinear Phenomena*, vol. 110, no. 1-2, pp. 43–50, 1997.
- [18] D. Chelidze and M. Liu, "Multidimensional damage identification based on phase space warping: an experimental study," *Nonlinear Dynamics*, vol. 46, no. 1-2, pp. 61–72, 2006.
- [19] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical Systems and Turbulence, Warwick 1980*, vol. 898 of *Lecture Notes in Mathematics*, pp. 366–381, Springer, Berlin, Germany, 1981.
- [20] M. Ragulskis and K. Lukoseviciute, "Non-uniform attractor embedding for time series forecasting by fuzzy inference systems," *Neurocomputing*, vol. 72, no. 10–12, pp. 2618–2626, 2009.
- [21] Y. B. Zhan, J. P. Yin, X. W. Liu, and G. M. Zhang, "Adaptive neighborhood selection based on local linearity for manifold learning," *Journal of Computer Research and Development*, vol. 48, no. 2, pp. 576–583, 2011.

Research Article

Automatic Classification of Remote Sensing Images Using Multiple Classifier Systems

Bin Yang,^{1,2} Chunxiang Cao,¹ Ying Xing,³ and Xiaowen Li¹

¹State Key Laboratory of Remote Sensing Science, Institute of Remote Sensing and Digital Earth, Chinese Academy of Science, Beijing 100101, China

²University of Chinese Academy of Sciences, Beijing 100049, China

³State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Chunxiang Cao; caocx@radi.ac.cn

Received 20 September 2014; Accepted 19 November 2014

Academic Editor: Vishal Bhatnaga

Copyright © 2015 Bin Yang 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.

It is a challenge to obtain accurate result in remote sensing images classification, which is affected by many factors. In this paper, aiming at correctly identifying land use types reflected in remote sensing images, support vector machine, maximum likelihood classifier, backpropagation neural network, fuzzy c-means, and minimum distance classifier were combined to construct three multiple classifier systems (MCSs). Two MCSs were implemented, namely, comparative major voting (CMV) and Bayesian average (BA). One method called WA-AHP was proposed, which introduced analytic hierarchy process into MCS. Classification results of base classifiers and MCSs were compared with the ground truth map. Accuracy indicators were computed and receiver operating characteristic curves were illustrated, so as to evaluate the performance of MCSs. Experimental results show that employing MCSs can increase classification accuracy significantly, compared with base classifiers. From the accuracy evaluation result and visual check, the best MCS is WA-AHP with overall accuracy of 94.2%, which overmatches BA and rivals CMV in this paper. The producer's accuracy of each land use type proves the good performance of WA-AHP. Therefore, we can draw the conclusion that MCS is superior to base classifiers in remote sensing image classification, and WA-AHP is an efficient MCS.

1. Introduction

With the development of remote sensing technology, it has been widely applied in many different fields such as land use land cover (LULC) monitoring, investigation of forest resources, disaster monitoring, and urban planning [1, 2], where the identification of land use types by image classification technology plays a very important role. There are a large number of land use types with irregular distribution on the surface of earth. Remote sensing images reflect the complicated information of the earth's surface, so it is a challenge to make accurate classification of remote sensing images.

Many artificial intelligence (AI) methods have been utilized in remote sensing images classification including neural networks [3–5], support vector machine (SVM) [6–8], and

maximum likelihood classifier (MLC) [9, 10]. Although the results of these classifiers are generally positive, they still have their own limitations. According to a large number of studies, there is no individual algorithm that performs perfectly in classification. Different classifiers have different accuracies for the same class, while the same classifier has different accuracies for different classes, due to the complementary advantages of different classifiers. As a result, multiple classifier systems (MCSs) naturally become a good choice. Combining strategies for multiple classifiers has been widely investigated, the aim of which is to determine an efficient combination method that makes full use of the complementary advantages of each classifier and tackles the drawbacks of individual classifiers, to improve the accuracy of classification. In the early period, MCS theory was developed in pattern recognition such as signal processing, handwriting

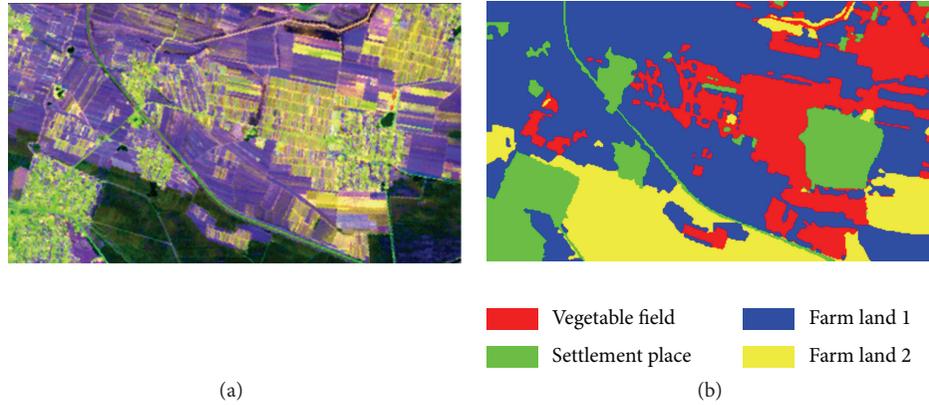


FIGURE 1: The studied area: (a) remote sensing image of SPOT-5; (b) ground truth map.

recognition, face recognition, and fingerprint identification [11–14]. In recent years, MCS has been gradually introduced to remote sensing images processing [1, 7]. There are many methods combining multiple classifiers in previous work. According to the types of the output of the base classifiers, MCS methods can be grouped into three categories, namely, abstract level, ranked level, and measurement level. When it comes to remote sensing images classification, MCSs are usually implemented in abstract level and measurement level. Xu proposed standard methods of MCS that combined multiple classifiers in both levels in the pattern recognition field [11, 12]. Voting method and Bayesian average (BA) method are the classical methods in abstract level and measurement level, respectively.

For the purpose of improving the accuracy of identifying LULC types, we applied some traditional AI methods in remote sensing images classification. Three MCS methods integrating support vector machine (SVM), maximum likelihood classifier (MLC), back propagation neural network (BPNN), fuzzy *c*-means (FCM), and minimum distance classifier (MDC) were implemented, one of which was proposed as a novel method called weighted average based on AHP (WA-AHP), making full use of the advantages of AHP and MCS. In this paper, we compared the voting method, BA, and WA-AHP. Confusion matrix and several indicators were computed to evaluate classification accuracy. Meanwhile, we illustrated the receiver operating characteristic (ROC) curve, from which area under curve (AUC) was derived, so as to evaluate the performance of the classifiers including base classifiers and MCSs. Finally, we discussed the application of MCSs in remote sensing images classification.

2. Data Sets

We chose a plot of the SPOT-5 satellite image in Liaoyang city of China as shown in Figure 1(a). According to visual interpretation and field measured data, we made the ground truth map as shown in Figure 1(b). In this region covered by the remote sensing images, there were four main types of land use, namely, vegetable field, settlement place, farm

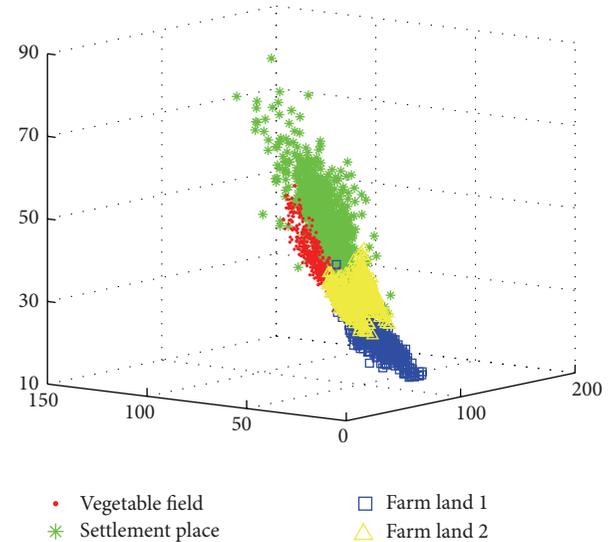


FIGURE 2: Location of the data for each class in feature space.

land 1, and farm land 2. Meanwhile, as shown in Figure 2, we analyzed the spectral character of each type. The coordinate axes represent the bands of the SPOT-5 image. It can be seen that the four types have different spectral characters.

3. Methodology

We aimed at improving the classification accuracy for remote sensing images by combining multiple classifiers. As shown in Figure 3, the framework consists of three major steps. Firstly, five base classifiers were selected and implemented, respectively. Secondly, based on the output of the five base classifiers, the classification results by comparative major voting (CMV), Bayesian average, and WA-AHP were obtained, respectively. Finally, the performance of all classification methods was evaluated including base classifiers and MCSs.

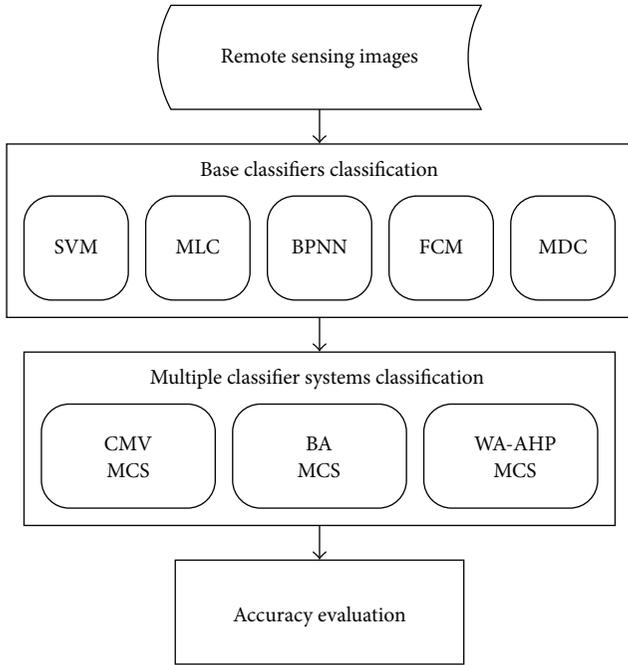


FIGURE 3: Framework of remote sensing images classification using MCS.

3.1. Base Classifiers

3.1.1. Support Vector Machine. SVM is a machine learning method which uses a certain distance between samples as the criterion of classification, based on the principle of structural risk minimization. This method has been found to be efficient for pattern recognition and recently for satellite image classification [7, 8, 15]. In SVM classification, the selection of the kernel function is an important step, and Gaussian radial basis function was selected in this paper.

3.1.2. Maximum Likelihood Classifier. MLC is a well-known method for determining a class based on Bayesian formula [9, 10]. In theory, there is an assumption in MLC of the training samples with normal distribution. During the process of classification, MLC builds the probability density functions for each class. All unclassified pixels are assigned membership based on the relative likelihood (probability) of that pixel occurring within each class. The probability is calculated as follows. First, it is supposed that there are G predefined classes. Then, according to Bayesian formula, the posterior probability of X is labeled to class k , and $P(G_k/X)$ is defined by the following equation:

$$P\left(\frac{G_k}{X}\right) = \frac{P(X/G_k)P(G_k)}{P(X)}, \quad (1)$$

where $P(G_k)$ is the prior probability of class k , $P(X/G_k)$ is the conditional probability of X from G_k , and $P(X)$ is the probability of X and is the same for each class. When the statistical probability of the given pixel is calculated, pixel X will be assigned to the class with the highest probability.

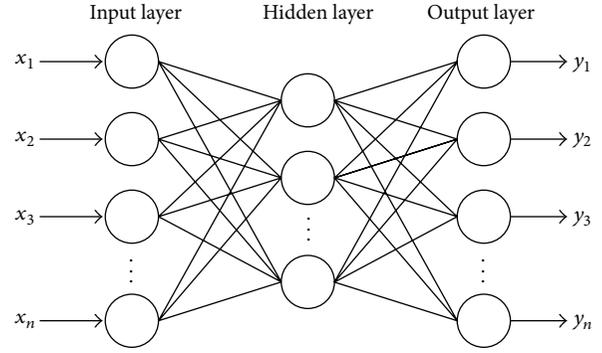


FIGURE 4: The structure of BPNN.

3.1.3. Back Propagation Neural Network. BPNN is widely used in pattern classification. BPNN with three layers of structure can solve arbitrary classification problem in theory. The three layers include input layer, hidden layer, and output layer [4, 5]. A simple sketch map of BPNN is shown in Figure 4.

BPNN needs some input sample sets and the known correct outputs of each case to learn. The learning process of BPNN consists of forward propagation and back propagation. In the process of forward propagation, information input from the input layer is propagated through the hidden layer to the output layer. The state of neurons in one layer affects the state of neurons only in the next layer. BPNN defines an object error function using mean squared error (MSE) as shown in (2), where Y_i is the actual output and \bar{Y}_i is the predicted output. If there is a difference between Y_i and \bar{Y}_i , then the neural network turns to the process of error back propagation. The error value is then propagated backwards through the network, and small changes are made to the weights of the neurons in each layer. The whole process is repeated until the error value drops below a predetermined threshold; then the network has learned the problem well:

$$MSE = \frac{1}{N} \sum_{i=1}^N (Y_i - \bar{Y}_i)^2. \quad (2)$$

3.1.4. Fuzzy c-Means. FCM is a method of clustering which allows one piece of data to belong to two or more clusters. This method is frequently used in pattern recognition [16]. It is based on minimization of the objective function J_m in (3), where C is the number of cluster centers, N is the number of data, x_i is the feature vector for the unknown input, c_j is the center of cluster j , u_{ij} is the degree of membership of x_i belonging to cluster j , and m is a real number greater than 1:

$$J_m = \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \|x_i - c_j\|^2, \quad 1 \leq m < +\infty. \quad (3)$$

FCM is carried out by iterative optimization of the objective function J_m through the update of membership u_{ij} and the cluster centers c_j in (4). This iteration will stop when $\max_{i,j} \{|u_{ij}^{k+1} - u_{ij}^k|\} < \varepsilon$, where ε is a termination threshold between 0 and 1, and k is the iteration step. Objective function

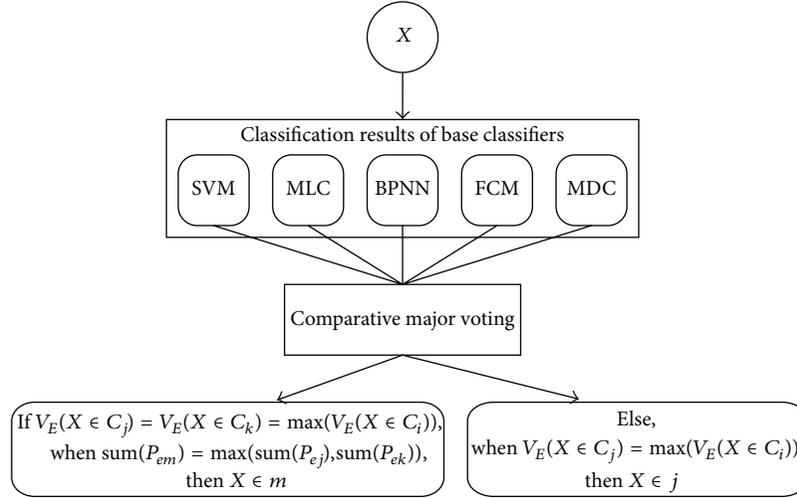


FIGURE 5: Flowchart of CMV strategy.

J_m will converge to a local minimum when this procedure is finished:

$$u_{ij} = \frac{1}{\sum_{k=1}^N (\|x_i - c_j\| / \|x_i - c_k\|)^{2/(m-1)}}, \quad (4)$$

$$c_j = \frac{\sum_{i=1}^C u_{ij}^m x_i}{\sum_{i=1}^C u_{ij}^m}.$$

3.1.5. Minimum Distance Classifier. MDC is a method which classifies an unknown class according to the distance between the unknown class and the centers of clusters, as shown in (5), where x_i is the i th feature vector for the unknown input and c_j is the center of the cluster j . This method aims to assign the class with the minimum distance to the unknown class pixel:

$$d_j = |x_i - c_j|. \quad (5)$$

3.2. Multiple Classifier Systems

3.2.1. Abstract Level MCS-Comparative Major Voting. The voting method derives from the hypothesis that the decision of a group is superior to that of the individuals. It includes majority voting rule and conservative voting rule. In the principle of majority voting, if one pixel is identified as the same class by most base classifiers, the pixel is labelled to this class. And for conservative voting rule, unless one pixel is identified as the same class by all the base classifiers, it is labeled to the class. Otherwise, it cannot be classified. The rule of class determination is shown in (6), where V_E denotes the voting value which is the number of classifiers with consistent classification results and $M + 1$ means the unclassified class.

When α is 0.5 and 1, it is the majority voting and conservative voting, respectively:

$$E(X) = \begin{cases} j, & \text{when } V_E(X \in C_j) = \max(V_E(X \in C_i)), \\ & V_E(X \in C_j) \geq \alpha \times K \\ M + 1, & \text{otherwise.} \end{cases} \quad (6)$$

However, this voting strategy has a drawback that when different classes obtain the same voting value for one pixel, the pixel cannot be labeled, which may greatly affect the classification accuracy. So we proposed a new approach called CMV to combine the base classifiers. The general flowchart of CMV is given in Figure 5. In this rule, two principles are followed: the decision of the majority is superior to that of the individual; and if several different classes get the same voting value for one pixel, the class of this pixel is determined by comparing the summations of probability of the concerned classes outputted by all the base classifiers. The pixels were labeled to the class with the highest summation.

3.2.2. Measurement Level MCS-Bayesian Average. Most classifiers can output posterior probability, which represents the probability that input pixel X belongs to class C_i , as shown in (7). Specifically, we supposed that there are M classes and K base classifiers which can output a vector $P_e(X)$, and $P_e(X) = [P_e(1), \dots, P_e(M)]^T$. Here, $P_e(C_i)$ denotes the posterior probability that pixel X belongs to C_i through classifier e . One approach called Bayesian average calculates the average value of posterior probabilities. The final classification criterion is

shown in (8), which means that pixel X is labeled by the class of the largest posterior probability:

$$P_E(X \in C_i) = \frac{1}{K} \sum_{k=1}^K P_k(X \in C_i), \quad i = 1, \dots, M, \quad (7)$$

$E(X)$

$$= \begin{cases} j, & \text{when } P_E(X \in C_j) = \max(P_E(X \in C_i)) \\ M + 1, & \text{otherwise.} \end{cases} \quad (8)$$

Some base classifiers are not capable of outputting posterior probability, in which case we adopted (9), where $d_k(C_i | X)$ denotes the distance between pixel X and the center of cluster C_i through the k th classifier:

$$P_k(X \in C_i) = \frac{1/d_k(C_i | X)}{\sum_{i=1}^K 1/d_k(C_i | X)}. \quad (9)$$

3.2.3. Weighted Average Based on AHP

(1) *Analytic Hierarchy Process*. Analytic hierarchy process (AHP) [17, 18] is a structured and expert grading-based technique for organizing and analyzing complex decisions. AHP uses experience and knowledge to order the indicators in the criteria layer and constructs judgment matrix to calculate the weight of each indicator. It was developed by Thomas L. Saaty in the 1970s and has been extensively studied and refined since then. It has particular application in group decision making and weights determination. It is widely used in many fields such as government, business, industry, healthcare, and education, however, rarely in remote sensing image classification.

(2) *WA-AHP*. BA is a simple method to calculate the average posterior probability. There is an obvious drawback with it that it does not consider the accuracy difference between all the base classifiers. So we take the advantages of both AHP and BA and integrate them into a novel method as shown by (10), where w_k determined by AHP is the weight of each classifier and K is the number of base classifiers:

$$P_E(X \in C_i) = \frac{1}{K} \sum_{k=1}^K w_k P_k(X \in C_i), \quad i = 1, \dots, M. \quad (10)$$

To be specific, after comparing the classification accuracy of base classifiers, we calculated the classifier weight of each base classifier through AHP. The higher the classification accuracy is, the larger weight the classification has. In this paper, we selected producer's accuracy to measure the classification accuracy of each base classifier for each class. By consistency check, the consistency ratio (CR) is less than 0.10, meaning that the judgment matrix has a reasonable consistency. The final classification rule is like the one used by BA. Pixel X will be classified into the class with the largest posterior probability in all the predetermined classes.

3.3. Postprocessing

3.3.1. *Smooth Processing*. Because of the complexity of the earth surface, there are some isolated pixels inconsistent with neighboring pixels in the classification results. Meanwhile, there are also many noise spots in the results. So, in order to obtain better classification results, we used mean filtering to smooth the classification results [19].

3.3.2. *Evaluation of Classification Accuracy*. The classification algorithms mentioned in this paper are compared using the following evaluation methods.

(1) *ROC Curve*. ROC curve is widely used in data mining, for it is an intuitive method that visually utilizes curves to evaluate the performance of classifiers. However, it is suitable for classification of two classes, so it is rarely applied in the classification of remote sensing images. In this paper, we adopted the following strategies to introduce ROC curve into accuracy evaluation of remote sensing image classification. There are many classes in the remote sensing images, and we suppose that all of them are divided into two classes, namely, target class and nontarget class. Target class is the one to be evaluated, while the remaining classes are considered to be of nontarget class. Then the classification of multiple classes was converted to binary classification, and consequently ROC curve can be obtained. Usually, a higher AUC which represents the area under the ROC curve denotes a better performance [20].

(2) *Indicators Derived from Confusion Matrix*. We used ground truth map and classification results to calculate confusion matrix, with which the classification accuracy is evaluated. Confusion matrix is denoted by C in the following parts. Each element C_{ij} in the confusion matrix is the number of records pertaining to class i that have been automatically classified in the class j . So the diagonal elements correspond to the numbers of records that have been correctly classified. Indicators such as overall accuracy (P_c), kappa coefficient (Kappa), producer's accuracy (P_{Aj}), and user's accuracy (P_{Ui}) are derived from confusion matrix as shown by (11), where C_{i+} and C_{+i} represent the sums of the elements in the i th row and the i th column and N is the number of samples. Kappa ranges from 0 to 1. Higher values of indicators indicate more accurate results. Thus,

$$\begin{aligned} P_c &= \sum_{k=1}^r \frac{C_{kk}}{N}, \\ \text{Kappa} &= \frac{N \times \sum_{i=1}^r C_{ii} - \sum_{i=1}^r C_{i+} C_{+i}}{N^2 - \sum_{i=1}^r C_{i+} C_{+i}}, \\ P_{Aj} &= \frac{C_{jj}}{C_{j+}}, \\ P_{Ui} &= \frac{C_{ii}}{C_{i+}}. \end{aligned} \quad (11)$$

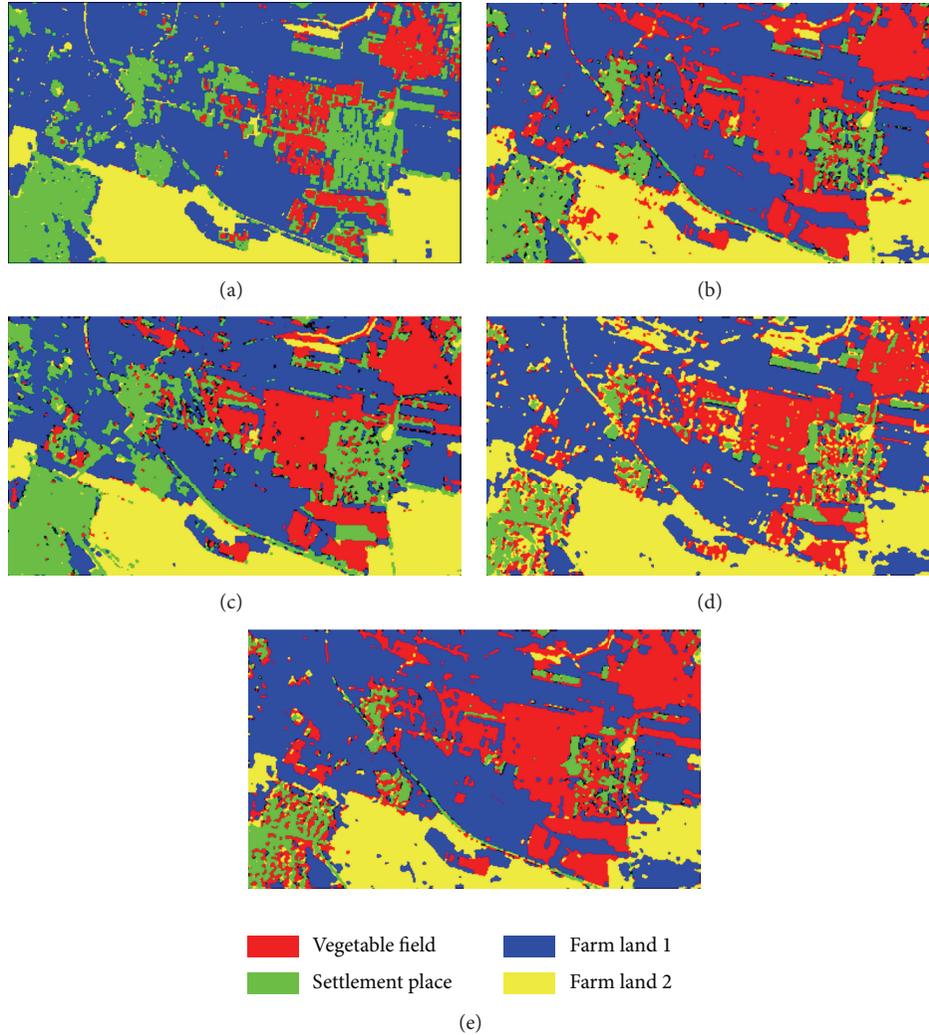


FIGURE 6: The classification results of base classifiers: (a) SVM; (b) MLC; (c) BPNN; (d) FCM; (e) MDC.

4. Experimental Results and Analysis

4.1. The Classification Results. We made classification of remote sensing images using SVM, MLC, BPNN, FCM, and MDC, respectively. The classification results of base classifiers were shown in Figure 6.

We used CMV, BA, and WA-AHP to construct MCS to classify remote sensing images. When using WA-AHP, it is required to obtain the priorities of base classifiers by their classification accuracy. The producer's accuracy of base classifiers was used to order the classifiers. Finally, the weight of each base classifier was obtained through AHP as shown in Table 1. The classification results of the three MCSs were shown in Figure 7.

4.2. Analysis of Classification Accuracy. We used ROC curve to evaluate the performance of all classifiers. Meanwhile, the confusion matrix is the base of calculating the accuracy evaluation indicators, so we constructed confusion matrix for

TABLE 1: Weight of each base classifier in WA-AHP.

Class	SVM	MLC	BPNN	FCM	MDC
Vegetable field	0.3651	0.3651	0.1622	0.0796	0.0279
Settlement place	0.0702	0.1325	0.5057	0.0337	0.258
Farm land 1	0.4174	0.1602	0.0615	0.0975	0.2634
Farm land 2	0.0733	0.2467	0.1589	0.3828	0.1384

each classifier including MCSs. More details are described as below.

4.2.1. ROC Curve. ROC curves for each class of different classifiers were shown in Figure 8, and Table 2 described the corresponding AUC. Although all the classifiers performed rather well, including base classifiers and MCSs, the performance of different classifiers varied for different land use types. To be specific, the best performances are SVM for vegetable field and farm land 1, BPNN for settlement place, and FCM for farm land 2, respectively. MCSs are superior to

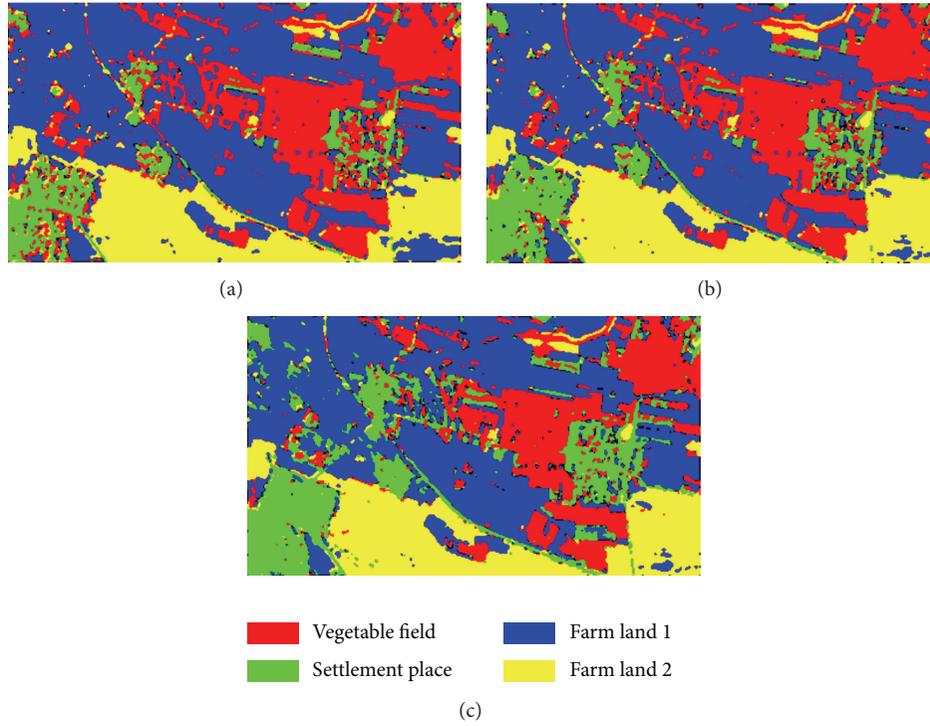


FIGURE 7: The classification results of MCSs: (a) CMV; (b) BA; (c) WA-AHP.

TABLE 2: AUC of each ROC curve.

Class	SVM	MLC	BPNN	FCM	MDC	CMV	BA	WA-AHP
Vegetable field	0.931	0.914	0.898	0.882	0.927	0.918	0.910	0.926
Settlement place	0.822	0.816	0.889	0.819	0.864	0.839	0.911	0.887
Farm land 1	0.949	0.902	0.869	0.943	0.911	0.920	0.923	0.955
Farm land 2	0.938	0.9327	0.925	0.950	0.925	0.939	0.936	0.956

TABLE 3: The comparison of overall accuracy and kappa coefficient.

	SVM	MLC	BPNN	FCM	MDC	BIC	CMV	BA	WA-AHP
Overall accuracy	0.873	0.862	0.832	0.784	0.807	SVM	0.895	0.859	0.942
Kappa coefficient	0.814	0.802	0.766	0.694	0.719	SVM	0.848	0.801	0.896

most base classifiers for different classes to a certain degree. We also found that AUC of WA-AHP is the largest for most classes except settlement place. As a result, the performance of WA-AHP rivals CMV and BA, proving its effectiveness in MCS.

4.2.2. Overall Accuracy and Kappa Coefficient. By the equations described in Section 3.3.2, we calculated the overall accuracy and kappa coefficient, as shown in Table 3, where BIC means the best individual classifier and SVM and MLC are two best base classifiers in this paper due to their better performance. MCS improved the classification accuracy significantly compared with base classifiers. The overall accuracy and kappa coefficient of WA-AHP were as good as those of CMV and are better than BA.

4.2.3. Producer's Accuracy and User's Accuracy. Producer's accuracy and user's accuracy were calculated to compare the classification accuracy of all the classifiers for each class, as shown in Tables 4 and 5. Table 4 shows that, in terms of producer's accuracy, the best individual classifiers (BIC) are SVM for vegetable field and farm land 1, BPNN for settlement place, and FCM for farm land 2, respectively. Table 5 shows that, in terms of user's accuracy, BIC are MDC for vegetable field, SVM for settlement place and farm land 2, and FCM for farm land 1, respectively. Compared with base classifiers, three MCSs improved the producer's accuracy for each land use type. We also found that the producer's accuracy of each class using WA-AHP rivals the other two MCSs. So we can draw a conclusion that WA-AHP is an effective method for MCS.

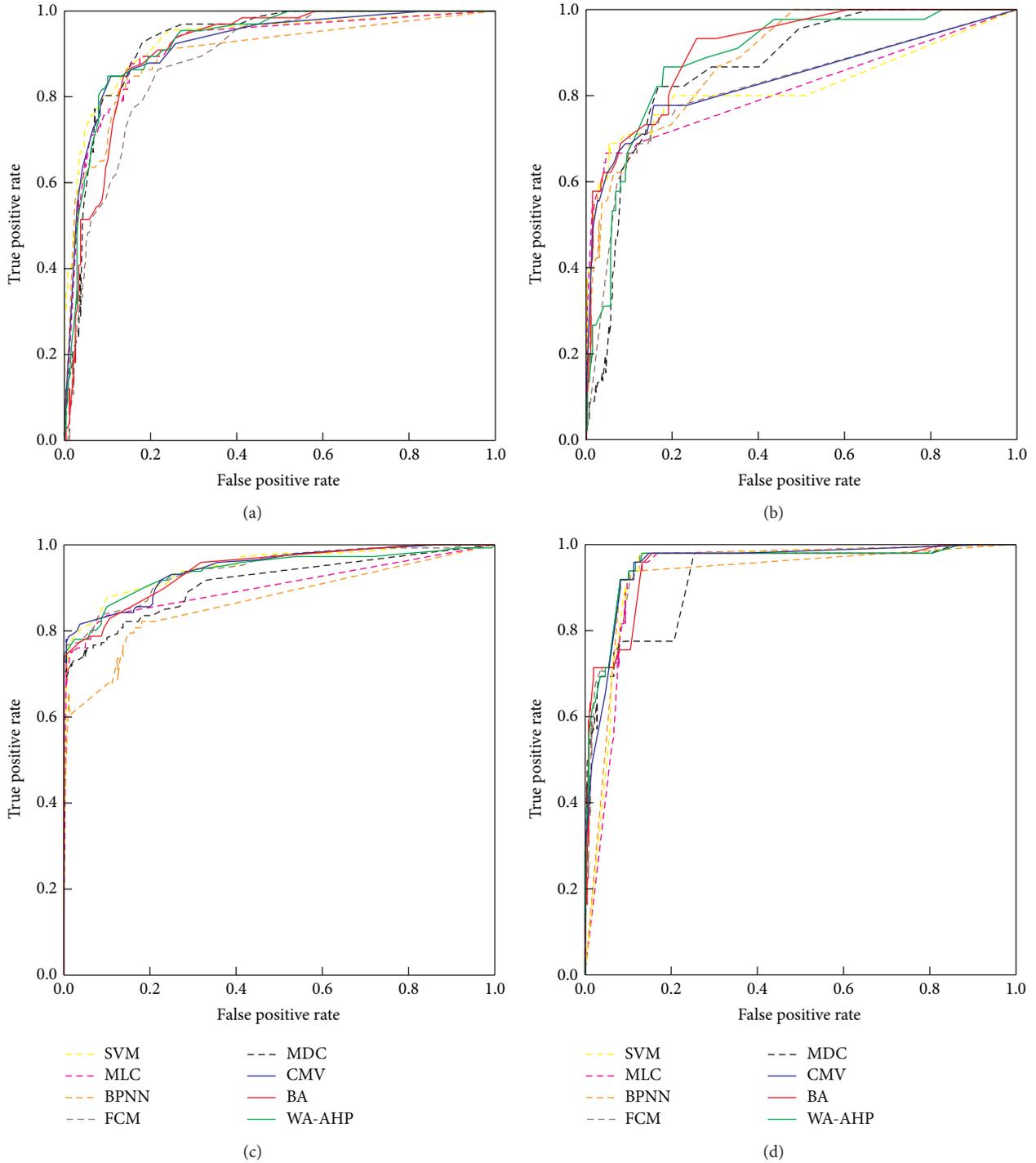


FIGURE 8: ROC curve for each class: (a) vegetable field; (b) settlement place; (c) farm land 1; (d) farm land 2.

TABLE 4: The comparison of producer's accuracy.

Class	SVM	MLC	BPNN	FCM	MDC	BIC	CMV	BA	WA-AHP
Vegetable field	0.939	0.894	0.833	0.727	0.500	SVM	0.939	0.864	0.955
Settlement place	0.689	0.778	0.933	0.533	0.889	BPNN	0.756	0.911	0.844
Farm land 1	0.890	0.842	0.767	0.815	0.884	SVM	0.884	0.808	0.904
Farm land 2	0.898	0.959	0.939	0.999	0.918	FCM	0.999	0.959	0.999

TABLE 5: The comparison of user's accuracy.

Class	SVM	MLC	BPNN	FCM	MDC	BIC	CMV	BA	WA-AHP
Vegetable field	0.705	0.720	0.833	0.696	0.917	MDC	0.747	0.814	0.840
Settlement place	0.939	0.897	0.618	0.800	0.563	SVM	0.944	0.672	0.950
Farm land 1	0.956	0.961	0.982	0.992	0.896	FCM	0.977	1.000	0.992
Farm land 2	0.898	0.825	0.793	0.563	0.818	SVM	0.891	0.825	0.845

5. Conclusion

We used multiple classifier systems in remote sensing image classification. We selected five classifiers to be base classifiers, including SVM, MLC, BPNN, FCM, and MDC. Three MCSs were constructed, namely, comparative major voting, Bayesian average, and weighted average based AHP proposed in this paper. We compared base classifiers and MCSs. Meanwhile, ROC curve is illustrated and some accuracy indicators were calculated to evaluate the classification results. From this paper, we can draw some conclusions as follows.

- (1) In terms of the overall classification accuracy, MCS is more accurate than base classifiers, such as MDC, MLC, BPNN, FCM, and SVM, because it integrates the advantage of each base classifier. The best base classifier is SVM in the selected five base classifiers. The overall accuracy and kappa coefficient of SVM are 0.873 and 0.814, respectively.
- (2) In terms of producer's accuracy of each type, the best individual classifiers are SVM for vegetable field and farm land 1, BPNN for settlement place, and FCM for farm land 2, respectively.
- (3) In terms of classification accuracy evaluation, the performance of WA-AHP proposed in this paper is superior to Bayesian average method and rivals comparative major voting method. That is because WA-AHP takes the accuracy difference of different base classifiers into consideration.
- (4) ROC curve used in the classification accuracy evaluation of remote sensing images is effective.

It is a challenge to improve classification accuracy of remote sensing images. And we introduced artificial intelligence technologies such as MCS, SVM, BPNN, MLC, FCM, and AHP to the research of remote sensing images classification, and the experimental results show that they improve the classification accuracy significantly. Our future research will focus on the application of more artificial intelligence technologies in processing of remote sensing images.

Conflict of Interests

The authors declare no conflict of interests.

Acknowledgments

The authors thank all reviewers for their critical review of the paper. This research was supported by the National 863 Program (2013AA12A302) and the National Natural Science Foundation of China (41301502).

References

- [1] U. Maulik and D. Chakraborty, "A robust multiple classifier system for pixel classification of remote sensing images," *Fundamenta Informaticae*, vol. 101, no. 4, pp. 287–304, 2010.
- [2] Y. S. Zhao, *The Principle and Method of Analysis of Remote Sensing Application*, Science Press, Beijing, China, 2003.
- [3] E. Merényi, W. H. Farrand, J. V. Taranik, and T. B. Minor, "Classification of hyperspectral imagery with neural networks: comparison to conventional tools," *EURASIP Journal on Advances in Signal Processing*, vol. 2014, no. 1, article 71, 2014.
- [4] B. Yang, Z. J. Liu, Y. Xing, and C. F. Luo, "Remote sensing image classification based on improved BP neural network," in *Proceedings of the International Symposium on Image and Data Fusion (ISIDF '11)*, pp. 1–4, IEEE, Yunnan, China, August 2011.
- [5] Y. Zhang, Z. Dong, L. Wu, and S. Wang, "A hybrid method for MRI brain image classification," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10049–10053, 2011.
- [6] G. M. Foody and A. Mathur, "A relative evaluation of multiclass image classification by support vector machines," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 6, pp. 1335–1343, 2004.
- [7] X. Ma, H. Shen, J. Yang, L. Zhang, and P. Li, "Polarimetric-spatial classification of SAR images based on the fusion of multiple classifiers," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 3, pp. 1064–1074, 2014.
- [8] P. Mitra, B. U. Shankar, and S. K. Pal, "Segmentation of multispectral remote sensing images using active support vector machines," *Pattern Recognition Letters*, vol. 25, no. 9, pp. 1067–1074, 2004.
- [9] J. Sun, J. Yang, C. Zhang, W. Yun, and J. Qu, "Automatic remotely sensed image classification in a grid environment based on the maximum likelihood method," *Mathematical and Computer Modelling*, vol. 58, no. 3–4, pp. 573–581, 2013.
- [10] J. C. Luo, Q. M. Wang, J. H. Ma, C. H. Zhou, and Y. Leung, "The EM-based maximum likelihood classifier for remotely sensed data," *Acta Geodaetica et Cartographica Sinica*, vol. 31, no. 3, pp. 234–239, 2002.
- [11] Y. C. Bo and J. F. Wang, "Combining multiple classifiers for the automatic classification of remotely sensed data," *Journal of Remote Sensing*, vol. 9, no. 5, pp. 555–563, 2005.
- [12] L. Xu, A. Krzyzak, and C. Y. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 3, pp. 418–435, 1992.
- [13] N. V. Chawla and K. W. Bowyer, "Designing multiple classifier systems for face recognition," in *Multiple Classifier Systems*, vol. 3541 of *Lecture Notes in Computer Science*, pp. 407–416, Springer, Berlin, Germany, 2005.
- [14] R. K. Nath, "Fingerprint recognition using multiple classifier system," *Fractals: Complex Geometry, Patterns, and Scaling in Nature and Society*, vol. 15, no. 3, pp. 273–278, 2007.

- [15] Y. Zhang, S. Wang, G. Ji, and Z. Dong, "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.
- [16] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: the fuzzy c-means clustering algorithm," *Computers and Geosciences*, vol. 10, no. 2-3, pp. 191-203, 1984.
- [17] B. Yang, C. X. Cao, R. Tian et al., "Recovery of evaluation the eco-environmental quality after the Wenchuan Earthquake," *Journal of Remote Sensing*, vol. 18, no. 4, pp. 946-957, 2014.
- [18] O. S. Vaidya and S. Kumar, "Analytic hierarchy process: an overview of applications," *European Journal of Operational Research*, vol. 169, no. 1, pp. 1-29, 2006.
- [19] X. Zhu, "Land cover classification using moderate resolution satellite imagery and random forests with post-hoc smoothing," *Journal of Spatial Science*, vol. 58, no. 2, pp. 323-337, 2013.
- [20] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861-874, 2006.

Research Article

Locating High-Impedance Fault Section in Electric Power Systems Using Wavelet Transform, k -Means, Genetic Algorithms, and Support Vector Machine

Ying-Yi Hong and Wei-Shun Huang

Department of Electrical Engineering, Chung Yuan Christian University, 200 Chung Pei Road, Chung Li 320, Taiwan

Correspondence should be addressed to Ying-Yi Hong; [yyh10632@yahoo.com.tw](mailto:yvh10632@yahoo.com.tw)

Received 22 July 2014; Accepted 14 November 2014

Academic Editor: Vishal Bhatnaga

Copyright © 2015 Y.-Y. Hong and W.-S. Huang. 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.

High-impedance faults (HIFs) caused by downed conductors in electric power systems are in general difficult to be detected using traditional protection relays due to small fault currents. The energized downed conductor can pose a safety risk to the public and cause a fire hazard. This paper presents a new method for locating the line (feeder) section of the HIF with the help of limited measurements in electric power systems. The discrete wavelet transform is used to extract the features of transients caused by HIFs. A modified k -means algorithm associated with genetic algorithms is then utilized to determine the placement of measurement facilities. The signal energies attained by wavelet coefficients serve as inputs to the support vector machine for locating the HIF line section. The simulation results obtained from an 18-busbar distribution system show the applicability of the proposed method.

1. Introduction

High-impedance faults (HIFs) in general occur in electric distribution systems. HIFs occur when a conductor contacts a tree with a high-impedance or when a broken conductor touches the ground. These faults may impose fire risks and cause electric shock that endangers lives of personnel. Therefore, HIF detection is essential to ensure safety. However, detection of HIFs using traditional protection devices (e.g., overcurrent and distance relay) is difficult because the resulting level of fault current is usually smaller than the nominal current.

Lien et al. proposed a method for detecting HIFs using three-phase energy variance for the second, fourth, and sixth harmonics of unbalanced current. Then counters are designed to detect HIF arcing through statistical confidence [1]. Emanuel et al. proposed that 120 Hz and 180 Hz components may be employed to detect HIFs. The field test was supported by a simple theoretical model and laboratory measurement [2]. Kim et al. used wavelet transform to extract HIF features for developing an HIF indicator [3]. Sedighi et al. presented a statistical pattern recognition, namely, principal

component analysis and Bayes classifier, for detecting HIF and discriminating it from other disturbances [4]. Lai et al. used the nearest neighbor rule approach to classify HIF and low-impedance fault (LIF) with the help of wavelet transform and voltage/current rms values [5]. Michalik et al. employed a phase displacement relation between wavelet coefficients of zero sequence voltages and currents to detect HIFs [6]. Sheng and Rovnyak used rms current, harmonic magnitudes, and phases in a decision tree for detecting HIFs [7].

On the other hand, the wavelet transform (WT) has been widely used for analyzing transient signals because of its varied window function for the time domain. The features of signals/functions can be easily extracted/decomposed via multiresolution analysis (MRA) [8]. There are many papers using discrete wavelet transform (DWT) to detect and classify PQ events [9–12]. Furthermore, artificial neural networks (ANNs) can be employed to map the input and output nonlinear relationship. The support vector machine (SVM), which is one of the ANNs, has recently been proposed for nonlinear regression and classification. Dash et al. used three SVMs for training to achieve fault classification, ground detection, and section identification, respectively, for the line

using thyristor-controlled compensated compensators [13]. Srinivasan et al. employed SVMs with linear and polynomial kernels developed for signature extraction and device identification [14]. Janik and Lobos used space phasor for feature extraction from three-phase signals to build distinguished patterns for SVM classifiers [15]. Other applications using SVM are, for example, load forecasting [16] and transient stability analysis [17].

In previous methods, “detection” means identification of an HIF in a feeder (or transmission line) [1–4] or in one of the multiple feeders (or transmission lines) [5–7] from the secondary side of a transformer at a substation. Locating a line (feeder) section, where an HIF occurs, has not been addressed in these papers. Moreover, different features, for example, even harmonics [1], low harmonics [2, 7], wavelet coefficients [3–5], voltage/current rms values [5], and phase displacement [6], were considered for detection. There was no salient result showing which features were better.

In this paper, locating the HIF line (feeder) section instead of detecting HIFs is addressed in a distribution system. Placement of multiple measurement facilities is determined first by a modified k -means algorithm associated with genetic algorithms. The discrete wavelet transform (DWT) is then used to extract features from these measurement locations for classification. Finally, the SVM is utilized to locate an exact HIF line section.

In Section 2, the problem description and assumptions are provided. The proposed method for locating the HIF section is given in Section 3. Simulation results obtained from an 18-busbar distribution system with HIFs are discussed in Section 4. Concluding remarks are given in Section 5.

2. Problem Description and Assumptions

Power engineers in general deal with the power event according to the following steps: (i) localization, (ii) classification, (iii) locating, and (iv) remedial action. These can be achieved with the help of the power supply monitoring system. When the monitored signals (voltage and current) are measured, the important features can be extracted using digital signal processing techniques. The monitoring system will assimilate the information including the features into useful knowledge/information through soft computing and machine learning for engineers to develop control strategy and to achieve decision-making.

In the last paragraph, “localization” means to identify the time for HIFs to occur. “Classification” indicates that HIFs should be discriminated from other disturbances, for example, load switching and low-impedance (short circuit) fault. “Locating” implies that an exact HIF line section should be identified. The second and third tasks will be emphasized in this paper. After locating the HIF, proper remedial actions will be activated by power engineers.

This paper deals with locating the HIF line section in a distribution system with multiple feeders using a power supply monitoring system including multiple measurement facilities at different lines. Locating a line (feeder) section,

where an HIF occurs, has not been addressed in the previous papers. There are several assumptions in this paper as follows.

- (i) The number of measurement facilities is given. In this paper, it is assumed that the supplier (utility) has a monitoring system including some measurement facilities that can be placed at different locations for recording.
- (ii) Locating single HIF is considered. The data-window size of the signal for processing in this paper is five cycles. Simultaneous HIFs at different lines hardly occur.
- (iii) Configuration of the studied distribution system is fixed. If the system topology is changed, the proposed neural network requires retraining. However, possible system configurations are generally known to engineers and the corresponding neural networks should be trained in advance.
- (iv) The HIF generally occurs at a single phase of a line section. The proposed method employed MATLAB/SIMULINK SimPowerSystems and all the three-phase transient voltages/currents at each busbar/line in the system are obtained.

3. The Proposed Method

The presented method includes three stages: (i) determining measurement sites, (ii) discriminating HIFs from other disturbances, and (iii) locating the HIF. The measurement sites are first determined by modified k -means algorithm associated with genetic algorithms. The proposed method then uses the wavelet coefficients of the currents (obtained by the measurements) as the features for classification of disturbances and the inputs of the SVM for locating the HIF.

3.1. Determination of Measurement Sites. In general, the number of power supply monitoring facilities is much smaller than the regular power, voltage, and current meters that are installed at all busbars and lines. Hence, a modified k -means algorithm is used to partition the system into C clusters (C is the number of power supply measurement facilities). C measurement facilities are placed at the lines near the pseudocenters of the C clusters. This subsection describes the modified k -means algorithm for partitioning the system for placement of the power supply measurement facilities. Moreover, the proposed modified k -means algorithm is enhanced from k -means [18, 19] and fuzzy- c -means (FCM) [20, 21] as follows.

Let $J(U, V)$ be an objective:

$$J(U, V) = \sum_{i=1}^N \sum_{c=1}^C (U_{ci}) \times \|X_i - V_c\|^2, \quad (1)$$

where C is the number of clusters; N represents the number of data (line section); V_c signifies the vector of the center in the c th clustering; X_i is the i th (known) data vector for clustering; U_{ci} denotes the characteristic value (0 or 1) as a weighting factor between V_c and X_i . If the minimum of $J(U, V)$ is

gained, the N sets of vectors are partitioned into C clusters and V_c is formulated by

$$V_c = \frac{\sum_{i=1}^N U_{ci} \times X_i}{\sum_{i=1}^N U_{ci}}, \quad 1 \leq c \leq C, \quad 1 \leq i \leq N. \quad (2)$$

Matrix of the characteristic values can be defined as follows:

$$U = [U_{ci}] \in R^{C \times N}. \quad (3)$$

For the i th column in the matrix U , the sum of all elements equals one and only one element in this column is unity. The traditional k -means algorithm did not consider (1), which is implemented in this proposed enhanced k -means algorithm.

The unknown variables in the problem of placement of measurement facilities are U_{ci} , $c = 1, \dots, C$, and $i = 1, \dots, N$. Traditional optimization methods involving the gradients of objective function cannot minimize (1) because of discontinuity of the objective function. The genetic algorithm was adopted to minimize (1) herein because the genetic algorithm can deal with binary variable U_{ci} efficiently [22]. The population size, crossover rate, and mutation rate in the genetic algorithm were assigned with 100, 0.9, and 0.01, respectively.

In this paper, X_i represents one of the current vectors (signal energies calculated by DWT) caused by an HIF at a line ℓ . The dimension (1×3660 herein) of X_i varies with the number of studied cases. Symbols C and N (3660 in this paper) are the numbers of measurement facilities and the scenarios with HIFs, respectively. Let L be the number of the line sections. Then $L X_i$'s need to be partitioned into C clusters. The vector V_c (1×3660) consisting of the virtual HIF currents serves as the center in the c th cluster. All vectors of the HIF currents X_i 's in the c th cluster geometrically center at V_c . Therefore, the criterion for placing measurement facilities in the electric distribution system is as follows: *place a measurement facility at line ℓ , at which the total Euclidean distance between X_i 's (HIFs occurring at line ℓ) and V_c is minimal, in the c th cluster.*

3.2. Discrete Wavelet Transform (DWT). Fourier transform (FT) is a suitable approach for studying problems with steady state responses. Short-time Fourier transform (STFT) divides the full-time interval into a number of small/equal-time intervals, which can be individually analyzed using FT. Although the result obtained from STFT contains time and frequency information, the equal-time intervals are fixed. Thus, STFT cannot be used to detect the transient signals. On the other hand, the discrete wavelet transform (DWT) has been widely used for analyzing the transient signals due to its varied scale and wavelet functions [23–25]. The features of signals can be easily extracted via the multiresolution analysis (MRA). DWT avoids the disadvantages of both FT and STFT.

A signal can be represented as a sum of wavelet functions $\varphi(t)$ and scale functions $\phi(t)$ with coefficients at different time shifts and scales (frequencies) using DWT. DWT can extract the features of transient signals by decomposing signal components overlapping in both time and frequency [8].

According to DWT, a time-varying function (signal) $f(t) \in L^2(R)$ can be expressed as follows:

$$\begin{aligned} f(t) &= \sum_k c_0(k) \phi(t-k) + \sum_k \sum_{j=1} d_j(k) 2^{-j/2} \varphi(2^{-j}t-k) \\ &= \sum_k c_{j_0}(k) 2^{-j_0/2} \phi(2^{-j_0}t-k) \\ &\quad + \sum_k \sum_{j=j_0} d_j(k) 2^{-j/2} \varphi(2^{-j}t-k), \end{aligned} \quad (4)$$

where c_0 and d_j represent the scaling (coarse) coefficient at scale 0 and wavelet (detailed) coefficient at scale j , respectively. The symbol k represents the translation coefficient. The scales $j = 1, 2, \dots$ denote the different (high to low) frequency bands. The variable j_0 is an integer. The translated and scaled (dilated) version of the wavelet, $\varphi(2^{-j}t-k)$, used in the multiresolution analysis (MRA), constructs a time-frequency picture of the signal.

There are some other wavelets in the wavelet theory [8]: Haar wavelets have compact support (a finite bounded set) but are discontinuous. Shannon wavelets are very smooth but are not compactly supported and they decay at infinity very slowly. Compared with these wavelets, Daubechies-4 belongs to a class of orthonormal basis-generating, continuous, and compactly supported wavelets. Daubechies-4 is adopted in this paper to extract the features of the line currents at scales 1, 2, and 3 with a sampling rate of 128 points/cycle.

3.3. Multiresolution Analysis (MRA). As shown in (4), $f(t)$ is constructed by $\phi(t)$ and decomposed by $\varphi(t)$ at different scales (resolution levels). $\varphi(t)$ generates the detailed version of $f(t)$ and $\phi(t)$ generates the coarse version of $f(t)$. It can be shown that [8]

$$c_{j+1}(k) = \sum_m h(m-2k) c_j(m), \quad (5)$$

$$d_{j+1}(k) = \sum_m h_1(m-2k) c_j(m), \quad (6)$$

where $h(m-2k)$ and $h_1(m-2k)$ are the low-pass and high-pass filters, respectively [8]. These two equations show that the scaling and wavelet coefficients at different scale levels can be obtained by convolving the expansion coefficients at scale j by the time-reversed recursion coefficients $h(\cdot)$ and $h_1(\cdot)$ and then downsampling or decimating to give the expansion coefficients at the next level of $j+1$. The term “downsampling” indicates that the number at lower scale j is double compared with that at higher scale $j+1$ due to the filters $h(m-2k)$ and $h_1(m-2k)$. This process is called the “analysis (decomposition)” from the fine scale to the coarse scale. The reverse process, called synthesis (construction), from the coarse scale to the fine scale, is omitted here. Figure 1 illustrates a three-scale MRA decomposition for a signal. The symbols h , h_1 , and “ $\downarrow 2$ ” denote the low-pass filter, high-pass filter, and “downsampling,” respectively.

The small scales represent high-frequency ranges. Only the wavelet coefficient (d_j) is regarded as a feature due to

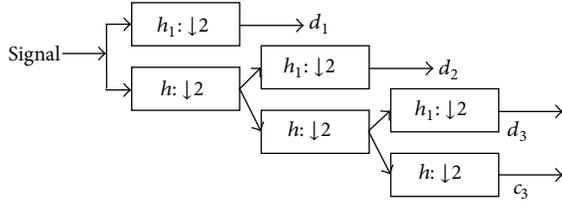


FIGURE 1: A three-scale MRA decomposition for a signal.

the high-frequency phenomena from HIFs. More specifically, if the sampling rate from the measurement facility is 128 points/cycle, then scales 1, 2, and 3 cover 3.84~1.92 kHz, 1.92~0.96 kHz, and 0.96~0.48 kHz, respectively. Lower harmonics were not considered for the SVM because they (with large values) do not provide significant discrimination among lines.

3.4. Parseval Theorem. When the MRA is applied to a transient signal, a large amount of wavelet coefficients will be attained. Although the wavelet coefficients are useful, it is difficult for SVM to train/validate that much information. More specifically, if sampling rate is 128 points/cycle and five cycles are utilized, the numbers of wavelet coefficients at scales 1, 2, and 3 are 320, 160, and 80 due to “downsampling,” respectively. Implementing $560 \times C$ input neurons in an SVM becomes impractical, where C is the number of measurement facilities defined in Section 3.1. A trade-off treatment using Parseval’s theorem is presented in this paper:

$$\int |f(t)|^2 dt = \sum_k |c_{j_0}(k)|^2 + \sum_k \sum_{j=1} |d_j(k)|^2. \quad (7)$$

In this paper, only $\sum_k \sum_{j=1} |d_j(k)|^2$ in (7) is calculated because the HIF belongs to transients. This term is called “current energy” or simply “energy” in this paper. Applications of $\sum_k \sum_{j=1} |d_j(k)|^2$ are as follows.

- (i) Determination of measurement facility placement: $\sum_k \sum_{j=1} |d_j(k)|^2$ is computed for each line section to be an element of X_i for a given scenario described in Section 3.1. The number of given scenarios is 3660 which will be discussed in Section 4.1.
- (ii) Feature extraction of transient signals: $\sum_k \sum_{j=1} |d_j(k)|^2$ is separated into the first to third scales ($d_1 \sim d_3$, $j = 1, 2$, and 3) for an HIF current at each line section. These features will serve as inputs for SVM.

3.5. Support Vector Machine (SVM). Traditional multilayer neural networks have some limitations: (i) many inputs due to need of diversity for inputs, (ii) requirement of crucial features for inputs, (iii) trial and error for number of neurons in the hidden layer, and (iv) multimodal with many local minimums. Avoiding the above demerits, SVM is a supervised artificial neural network designed for solving classification problems [26, 27]. In essence, SVM maximizes the margin

between the training data and the decision boundary, which can be formulated as a quadratic optimization problem. The subset of patterns that are closest to the decision boundary is called the support vector.

SVM maximizes the separating margin between two classes, given by a set of P data pairs (x_p, c_p) , where x_p and c_p denote the input vector and class, $p = 1, 2, \dots, P$, respectively. For linear separable training pairs of two classes, the separating hyperplane $h(x)$ is given by

$$h(x) = w^t x + b = 0, \quad (8)$$

where w and b are the vectors of weighting factors and biases, respectively. If a nonlinear hyperplane $\psi(\cdot)$ is considered, then

$$h(x) = w^t \psi(x) + b = 0. \quad (9)$$

The maximal separating margin can be attained by minimizing the following primal problem if two classes are not linearly distributed [28]:

$$\begin{aligned} \min \quad & \frac{1}{2} w^t w + K \sum_{p=1}^P \xi_p \\ \text{subject to} \quad & c_p (w^t \psi(x_p) + b) \geq 1 - \xi_p, \quad p = 1, 2, \dots, P, \end{aligned} \quad (10)$$

where ξ_p is the so-called fulfilling variable. The symbol K is a regularization parameter. In order to search a proper K , performance of the trained SVM needs assessment as follows. The training data are divided into two sets. One is used to train the SVM while the other, called the validation set, is used for evaluating the SVM. According to the performance on the validation set, a proper value of K can be attained.

Equations (10) and (11) can be transformed into the unconstrained Lagrangian:

$$\begin{aligned} L(w, b, \xi, \mu) = & \frac{1}{2} w^t w + K \sum_{p=1}^P \xi_p \\ & + \sum_{p=1}^P \mu_p [c_p (w^t \psi(x_p) + b) - 1 + \xi_p], \end{aligned} \quad (12)$$

where μ_p is the dual variable (Lagrange multiplier) for inequality constraint, (11). Obviously, the form of (11) is the same as the output of a neuron if $\varphi(\cdot)$ and ξ_p are considered the activating function of a neuron and nonnegative slack variable, respectively.

4. Simulation Results

4.1. Simulation Data. The applicability of the proposed methodology is verified by simulation results in this section. An 18-busbar radial system with 17 line sections illustrated in Figure 2 serves as a sample system in this paper. Its busbar and line data are provided in [29]. To train the SVM, the original load level was varied within $\pm 10\%$ (61 conditions) and the HIFs occur at different angles within $0^\circ \sim 359^\circ$ (4 conditions)

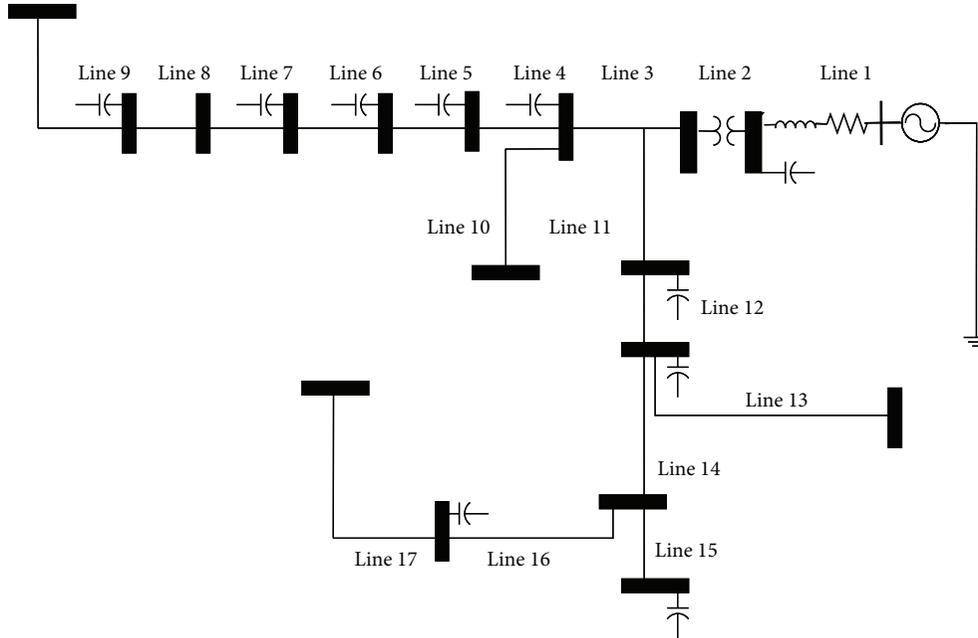


FIGURE 2: One-line diagram for studied distribution system.

and at 15 different line sections for obtaining a total of 3660 ($= 61 \times 4 \times 15$) data. 70%, 10%, and 20% of these 3660 data were used stochastically for training, validating, and testing the proposed SVM, respectively. The arc of HIF was modeled with two antiparallel DC sources and diodes which were connected to a random resistor [2]. The proposed methods were implemented by MATLAB 7.0 (SimPowerSystems) on a C2D (Core 2 Due) 2.13 GHz computer (RAM 3.5 G). The data-window size of the signal for processing in this paper is five cycles.

Because the power supply measurement facilities are more expensive than general meters, the number of measurement facilities is limited. Discussion of purchasing the measurement facilities and determination of a proper number for the measurement facilities are beyond the scope of this paper. Hence, 14, 11, 8, 4, and 2 measurement facilities (i.e., C) are assumed to be available in this paper. Table 1 illustrates the SVM information associated with measurements. Because HIF energies of the first to third scales ($d_1 \sim d_3$) were considered, the number of input neurons equals measurements (C) multiplied by 3. These are cases 1~5. Moreover, the current at the neural line of the main transformer is generally available and can be utilized. These are cases 6~10. Finally, four binary bits are sufficient for discriminating 15 line sections excluding primary sides (line 1) and the main transformer (denoted by line 2) in this system.

4.2. Feature Extraction by DWT. As described in Sections 3.3 and 3.4, the “energies” for HIF currents of the first to third scales at each line section are used as features for SVM inputs. Assume that an HIF occurs (90°) at line section 12. Figure 3 shows the energy distribution of the neighborhood of line section 12 (i.e., line sections 11 and 13). The energies for the

TABLE 1: SVM information associated with measurements.

	Facility number (C)	Input neurons	Output neurons
Case 1	2	2×3	4
Case 2	4	4×3	4
Case 3	8	8×3	4
Case 4	11	11×3	4
Case 5	14	14×3	4
Case 6	2 + 1	3×3	4
Case 7	4 + 1	5×3	4
Case 8	8 + 1	9×3	4
Case 9	11 + 1	12×3	4
Case 10	14 + 1	15×3	4

first to sixth scales at these three line sections are shown. It is apparent that the normal energy and HIF energy are almost the same for the current of the fourth (also for fifth and sixth) scale. Hence, current energies for the fourth to sixth scales cannot serve as features and only current energies for the first to third scales are considered further. Please note that the energies are normalized to be per unit and are in terms of \log_{10} because the energies of the fifth and sixth scales are much larger than those of other scales.

4.3. Scaled Energies at Line Sections. An HIF occurring at line section 12 is discussed in this section. Figure 4 illustrates the HIF currents in terms of the DWT-scaled energy distributions at each line section. The vertical axis denotes the energy magnitude while the horizontal axis means the $d_1 \sim d_3$ at each line section. Please note that the energies are normalized to be per unit.

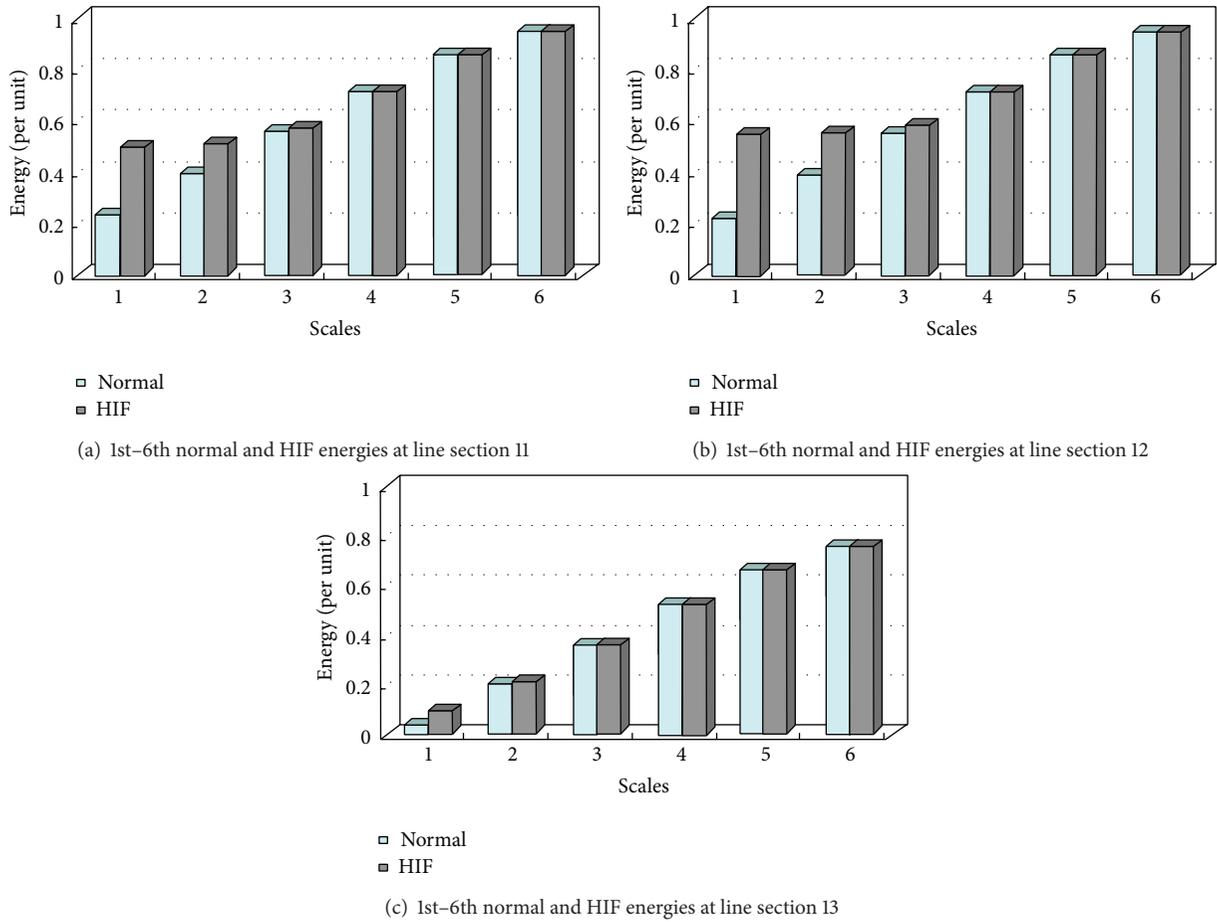


FIGURE 3: Energy distributions near faulted line.

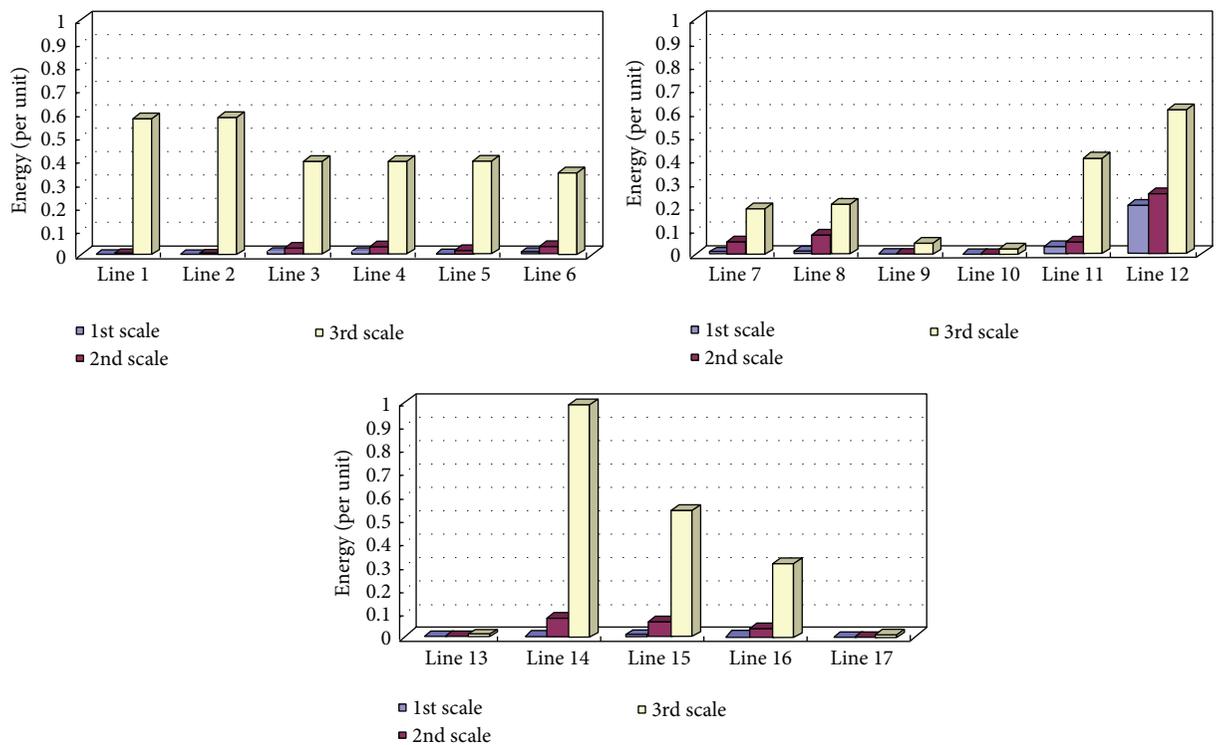


FIGURE 4: HIF energy distributions at all line sections.

TABLE 2: Different C s and corresponding clusters (k -means).

C	Line clusters
14	(1, 2), (3), (4), (5), (6), (7), (8), (9), (10), (11, 12), (13), (14), (15, 16), (17)
11	(1, 2), (3), (4, 5, 6), (7, 8), (9), (10), (11, 12), (13), (14), (15, 16), (17)
8	(1, 2), (3), (4, 5, 6), (7, 8), (9), (10), (11, 12), (13, 14, 15, 16, 17)
4	(1, 2), (3, 4, 5, 6, 7, 8, 9, 11, 12), (10), (13, 14, 15, 16, 17)
2	(1, 2), (3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17)

TABLE 3: Different C s and corresponding clusters (FCM).

C	Line clusters
14	(1), (2), (3), (4), (5), (6), (7), (8), (9, 10, 13, 17), (11), (12), (14), (15), (16)
11	(1, 2), (3), (4), (5), (6), (7, 8), (9, 10, 13, 17), (11), (12), (14), (15, 16)
8	(1, 2), (3), (4, 5, 6), (7, 8), (9, 10, 13, 17), (11), (12), (14, 15, 16)
4	(1, 2), (3, 4, 5, 6), (7, 8, 11, 12), (9, 10, 13, 14, 15, 16, 17)
2	(1, 2, 3, 4, 5, 6), (7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17)

When the busbar load varies and the HIF occurs at distinct angles, the above phenomena will be discriminated. Hence, the “energies” of the HIF currents of the first to third scales are important features for locating the HIF line section.

4.4. Measurement Facility Placement. Because 14, 11, 8, 4, and 2 power supply measurement facilities are assumed to be available in the test sample, C may be 14, 11, 8, 4, or 2 and $N = 3660$, respectively (C and N were defined in Section 3.1). For a given C , the same 3660 sets of data were employed to perform the HIF current energy clustering by the modified k -means algorithm. Table 2 illustrates the different C 's and corresponding clusters. Each cluster is quoted by parentheses. The line section with an italic font in Table 2 denotes the one installed with a measurement facility. Traditional FCM is also employed to study the line section clustering as shown in Table 3. As can be seen, some clusters obtained by the FCM are infeasible because lines in a cluster may not be adjacent to each other. For example, in the last row of Table 3, line sections 7, 8, and 9 in a cluster are not adjacent to line sections 10~17.

More specifically, the condition for $C = 2$ and $c = 2$ in the last row of Table 2 is described here: Table 4 illustrates the distances (norm) between X_i , $i = 3, 4, \dots, 17$, and its clustering center V_2 ($c = 2$) for $C = 2$. As can be seen, the distance between X_{12} and V_2 is the smallest. Therefore, the measurement facility is placed at line section 12.

In this paper, the dimension of X_i is 1×3660 where 3660 is the number of scenarios from the test system. Each element of X_i is the total energy at line section i for one of the 3660 cases. Moreover, there are 17 X_i 's (number of lines) for the 18-busbar system where 17 is the line number.

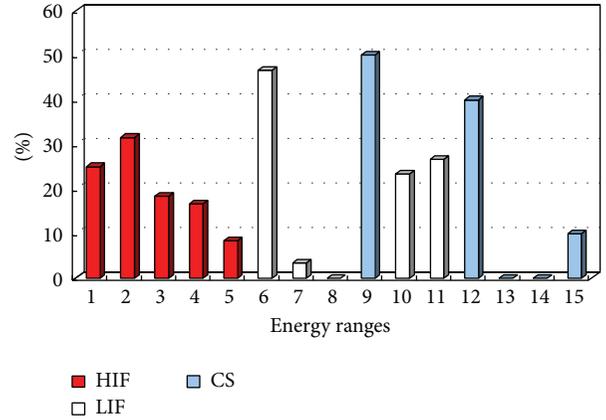


FIGURE 5: Total energy disturbances for 3 different disturbances.

4.5. Classification among HIF, Short Circuit, and Switching. There are 930 cases with short circuits (low-impedance faults, LIFs) and 620 cases with capacitor switching (CS) for further investigation of energy distributions. Figure 5 shows the total energy of the first to third scales for different HIFs, LIFs, and CS. The vertical axis denotes the percentage of occurrence for the three individual disturbances. The horizontal axis includes 15 energy ranges (\log_{10}). More specifically, ranges 1, 2, and 15 represent 3.48–3.56, 3.57–3.65, and 10.09–10.79 p.u., respectively. It can be found that HIFs include smaller total energies from range 1 to range 5. Hence, the total energy can be employed to discriminate HIFs from other disturbances, for example, LIF and CS.

4.6. Accuracy of Locating HIFs by SVM. As described in Section 4.1, 70%, 10%, and 20% of the 3660 data are used stochastically for training, validating, and testing, respectively. Table 5 illustrates the number of iterations, CPU time for training the SVM, and the accuracy rate for the 10 cases defined in Table 1. The following comments can be drawn from Table 5.

- (i) The numbers of iterations for all cases are almost the same for the SVM despite the different number of input neurons.
- (ii) The CPU time required varies with the number of input neurons.
- (iii) Accuracy rates are greater than 99% except for cases 1 and 6.
- (iv) Cases 1 and 6 with only two measurement facilities still gain accuracy rate of 97.4% and 98.36%, respectively. This result ensures the advantage of the proposed method using the SVM.
- (v) The neutral current at the substation improves accuracy rates only for cases 1 and 2 (corresponding to cases 6 and 7) with fewer measurements.

TABLE 4: Distances ($\times 10^5$) between X_i and its clustering center.

X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{16}	X_{17}
746.02	577.24	699.47	666.77	316.63	292.76	564.36	575.65	202.95	189.19	578.97	342.92	548.87	547.7	578.97

TABLE 5: Performance of SVM.

	Iterations number	CPU sec.	Accuracy rate (%)
Case 1	32	170	97.40
Case 2	29	163	99.18
Case 3	33	226	99.86
Case 4	30	234	99.72
Case 5	30	260	99.72
Case 6	32	171	98.36
Case 7	29	171	99.72
Case 8	33	234	99.86
Case 9	30	241	99.72
Case 10	31	279	99.72

5. Conclusions

This paper proposed a new method for locating the line section with an HIF using DWT, modified k -means, and SVM. Compared with the existing methods involving identification of an HIF in a feeder (or transmission line) or in one of the multiple feeders from the secondary side of a substation, the proposed approach is able to locate an HIF line section in a distribution system with multiple feeders using a power supply monitoring system including multiple power supply measurement facilities at different lines. Classification of disturbances and locating the HIF are addressed.

The features (current energies) at three distinct scales (frequency bands) were extracted by MRA in DWT. These features provide important information for the SVM to locate the line section with an HIF. Moreover, the energies of HIF are discriminated obviously from those of LIF and CS.

The simulation results obtained from an 18-busbar distribution system show that good accuracy can still be attained using only a few measurements (e.g., two in this paper) due to the SVM. Hence, SVM is very useful when only a few measurements are available.

Conflict of Interests

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

Acknowledgment

The authors gratefully acknowledge the financial support from the National Science Council, Taiwan, under Grant no. 96-2221-E-033-069-MY2.

References

- [1] K. Y. Lien, S. L. Chen, C. J. Liao, T. Y. Guo, T. M. Lin, and J. S. Shen, "Energy variance criterion and threshold tuning scheme for high impedance fault detection," *IEEE Transactions on Power Delivery*, vol. 14, no. 3, pp. 810–817, 1999.
- [2] A. E. Emanuel, D. Cyganski, J. A. Orr, S. Shiller, and E. M. Gulachenski, "High impedance fault arcing on sandy soil in 15kV distribution feeders: contributions to the evaluation of the low frequency spectrum," *IEEE Transactions on Power Delivery*, vol. 5, no. 2, pp. 676–686, 1990.
- [3] C. H. Kim, H. Kim, Y. H. Ko, S. H. Byun, R. K. Aggarwal, and A. T. Johns, "A novel fault-detection technique of high-impedance arcing faults in transmission lines using the wavelet transform," *IEEE Transactions on Power Delivery*, vol. 17, no. 4, pp. 921–929, 2002.
- [4] A.-R. Sedighi, M.-R. Haghifam, O. P. Malik, and M.-H. Ghassemian, "High impedance fault detection based on wavelet transform and statistical pattern recognition," *IEEE Transactions on Power Delivery*, vol. 20, no. 4, pp. 2414–2421, 2005.
- [5] T. M. Lai, L. A. Snider, E. Lo, and D. Sutanto, "High-impedance fault detection using discrete wavelet transform and frequency range and RMS conversion," *IEEE Transactions on Power Delivery*, vol. 20, no. 1, pp. 397–407, 2005.
- [6] M. Michalik, W. Rebizant, M. R. Lukowicz, S. J. Lee, and S. H. Kang, "High-impedance fault detection in distribution networks with use of wavelet-based algorithm," *IEEE Transactions on Power Delivery*, vol. 21, no. 4, pp. 1793–1802, 2006.
- [7] Y. Sheng and S. M. Rovnyak, "Decision tree-based methodology for high impedance fault detection," *IEEE Transactions on Power Delivery*, vol. 19, no. 2, pp. 533–536, 2004.
- [8] C. S. Burrus, R. A. Gopinath, and H. Guo, *Introduction to Wavelets and Wavelet Transforms*, Prentice Hall, Upper Saddle River, NJ, USA, 1998.
- [9] A. Elmitwally, S. Farghal, S. Abdelkader, and M. Elkateb, "Proposed wavelet-neurofuzzy combined system for power quality violations detection and diagnosis," *IEE Proceedings: Generation, Transmission and Distribution*, vol. 148, no. 1, pp. 15–20, 2001.
- [10] L. Angrisani, P. Daponte, M. D'Apuzzo, and A. Testa, "A measurement method based on the wavelet transform for power quality analysis," *IEEE Transactions on Power Delivery*, vol. 13, no. 4, pp. 990–998, 1998.
- [11] S. Santoso, E. J. Powers, W. M. Grady, and A. C. Parsons, "Power quality disturbance waveform recognition using wavelet-based neural classifier. Part 1. theoretical foundation," *IEEE Transactions on Power Delivery*, vol. 15, no. 1, pp. 222–228, 2000.
- [12] H. Mokhtari, M. Karimi-Ghartemani, and M. R. Iravani, "Experimental performance evaluation of a wavelet-based on-line voltage detection method for power quality applications," *IEEE Transactions on Power Delivery*, vol. 17, no. 1, pp. 161–172, 2002.
- [13] P. K. Dash, S. R. Samantaray, and G. Panda, "Fault classification and section identification of an advanced series-compensated transmission line using support vector machine," *IEEE Transactions on Power Delivery*, vol. 22, no. 1, pp. 67–73, 2007.
- [14] D. Srinivasan, W. S. Ng, and A. C. Liew, "Neural-network-based signature recognition for harmonic source identification," *IEEE Transactions on Power Delivery*, vol. 21, no. 1, pp. 398–405, 2006.

- [15] P. Janik and T. Lobos, "Automated classification of power-quality disturbances using SVM and RBF networks," *IEEE Transactions on Power Delivery*, vol. 21, no. 3, pp. 1663–1669, 2006.
- [16] S. Fan and L. Chen, "Short-term load forecasting based on an adaptive hybrid method," *IEEE Transactions on Power Systems*, vol. 21, no. 1, pp. 392–401, 2006.
- [17] L. S. Moulin, A. P. Alves Da Silva, M. A. El-Sharkawi, and R. J. Marks II, "Support vector machines for transient stability analysis of large-scale power systems," *IEEE Transactions on Power Systems*, vol. 19, no. 2, pp. 818–825, 2004.
- [18] N. K. Bose and P. Liang, *Neural Network Fundamentals with Graphs, Algorithms, and Applications*, McGraw-Hill, New York, NY, USA, 1996.
- [19] Y. K. Lam and P. W. M. Tsang, "eXploratory K-Means: a new simple and efficient algorithm for gene clustering," *Applied Soft Computing Journal*, vol. 12, no. 3, pp. 1149–1157, 2012.
- [20] T. Velmurugan, "Performance based analysis between k-Means and Fuzzy C-Means clustering algorithms for connection oriented telecommunication data," *Applied Soft Computing Journal*, vol. 19, pp. 134–146, 2014.
- [21] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: the fuzzy *c*-means clustering algorithm," *Computers & Geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984.
- [22] D. Datta, "Unit commitment problem with ramp rate constraint using a binary-real-coded genetic algorithm," *Applied Soft Computing Journal*, vol. 13, no. 9, pp. 3873–3883, 2013.
- [23] M. Jamil, A. Kalam, A. Q. Ansari, and M. Rizwan, "Generalized neural network and wavelet transform based approach for fault location estimation of a transmission line," *Applied Soft Computing*, vol. 19, pp. 322–332, 2014.
- [24] D. Bayram and S. Şeker, "Wavelet based Neuro-Detector for low frequencies of vibration signals in electric motors," *Applied Soft Computing Journal*, vol. 13, no. 5, pp. 2683–2691, 2013.
- [25] H. Eristi, "Fault diagnosis system for series compensated transmission line based on wavelet transform and adaptive neuro-fuzzy inference system," *Measurement*, vol. 46, no. 1, pp. 393–401, 2013.
- [26] S. Ekici, "Support Vector Machines for classification and locating faults on transmission lines," *Applied Soft Computing Journal*, vol. 12, no. 6, pp. 1650–1658, 2012.
- [27] M. Matsumoto and J. Hori, "Classification of silent speech using support vector machine and relevance vector machine," *Applied Soft Computing Journal*, vol. 20, pp. 95–102, 2014.
- [28] K. S. Chua, "Efficient computations for large least square support vector machine classifiers," *Pattern Recognition Letters*, vol. 24, no. 1–3, pp. 75–80, 2003.
- [29] W. M. Grady, M. J. Samotyj, and A. H. Noyola, "Minimizing network harmonic voltage distortion with an active power line conditioner," *IEEE Transactions on Power Delivery*, vol. 6, no. 4, pp. 1690–1697, 1991.

Research Article

Fault Diagnosis of Supervision and Homogenization Distance Based on Local Linear Embedding Algorithm

Guangbin Wang,¹ Jun Luo,¹ Yilin He,¹ and Qinyi Chen²

¹Hunan Provincial Key Laboratory of Health Maintenance for Mechanical Equipment, Hunan University of Science and Technology, Taoyuan Road No. 2, Xiangtan City, Hunan 411201, China

²ChuXiong Industrial School, Weichu Road No. 84, Chuxiong City, Yunnan 675099, China

Correspondence should be addressed to Guangbin Wang; jxxwgb@126.com

Received 6 August 2014; Revised 12 October 2014; Accepted 10 November 2014

Academic Editor: Saeed Balochian

Copyright © 2015 Guangbin 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.

In view of the problems of uneven distribution of reality fault samples and dimension reduction effect of locally linear embedding (LLE) algorithm which is easily affected by neighboring points, an improved local linear embedding algorithm of homogenization distance (HLLLE) is developed. The method makes the overall distribution of sample points tend to be homogenization and reduces the influence of neighboring points using homogenization distance instead of the traditional Euclidean distance. It is helpful to choose effective neighboring points to construct weight matrix for dimension reduction. Because the fault recognition performance improvement of HLLLE is limited and unstable, the paper further proposes a new local linear embedding algorithm of supervision and homogenization distance (SHLLE) by adding the supervised learning mechanism. On the basis of homogenization distance, supervised learning increases the category information of sample points so that the same category of sample points will be gathered and the heterogeneous category of sample points will be scattered. It effectively improves the performance of fault diagnosis and maintains stability at the same time. A comparison of the methods mentioned above was made by simulation experiment with rotor system fault diagnosis, and the results show that SHLLE algorithm has superior fault recognition performance.

1. Introduction

With the development of the modernization process, the structure of mechanical equipment has been more and more increasingly sophisticated while the degree of automation and function of realization grow increasingly stronger. In the fault diagnosis of rotating machinery, the more sophisticated the monitoring and control system, the more the numbers of sensors. What is more, the data of presenting state characteristics in real space is complex. These multiple variables make the data to present parameters of equipment running status become more complex which to describe the state of the data abstraction are the high-dimensional data. Faced with the characteristics of failure samples such as high-dimensional diversity, the traditional linear method has great limitations. However, the nonlinear manifold learning method was developed rapidly since it was first proposed in the journal of Science in 2000. From a huge amount of complicated and changeable high-dimensional observation

data, the methods make data analysis and state decision-making extend from the original Euclidean space to the manifold, which is able to identify the key information accurately and dig out signal essential characteristics and internal rules, which will be analyzed and judged for fault diagnosis. Classic manifold learning methods such as isometric feature mapping (ISOMAP) [1], local linear embedding (LLE) [2], local tangent space alignment (LTSA) [3, 4], and Laplacian eigenmap (LE) [5, 6] algorithm are mainly applied to the fields of data mining, image processing, pattern recognition, and information retrieval [7–9]. In recent years, the manifold learning methods have been applied to many kinds of rotating machinery fault diagnosis. Yang et al. [10] put forward the noise reduction algorithm in nonlinear time series based on reconstruction of phase space and main manifold identification and successfully extracted the impact characteristics of gearbox fault signal from the noise; Yang et al. [11] put forward a method of incremental local tangent space alignment.

In the different states of rolling bearing classification recognition, the algorithm of incremental learning for new samples has higher recognition rate; Liang et al. [12] proposed impact fault feature adaptive extraction algorithm based on manifold learning and extracted the optimal impact fault feature. For the extraction of weak feature of early fault, Li et al. [13] proposed the early fault diagnosis and feature extraction methods of the rolling bearing based on manifold learning, which improved the fault pattern classification performance. However, with the complex diversities and particularity of fault samples, the most of manifold learning methods in the practical application are optimization methods, and part of them are characterized by just the combination of two kinds of the methods and the stability of method which needs to be improved. Therefore, in this study with rotating machinery as the research object, an improved local linear embedding algorithm of homogenization distance (HLLLE) is proposed on the basis of the research of locally linear embedding (LLE) method to solve the problems that the real fault samples set are not distributed evenly and the phenomena that LLE is easily affected by the neighboring points. Moreover, in order to further improve the recognition performance rate and stability of the method, the supervised learning mechanism is added. And a new local linear embedding algorithm of supervision and homogenization distance (SHLLE) is proposed. Then the rotor system fault simulation experiment is used to analyze and validate the effectiveness of fault diagnosis.

2. Locally Linear Embedding Algorithm of Homogenization Distance

2.1. Locally Linear Embedding Algorithm. Euclidean distance is used in LLE algorithm for local linear fitting to show the overall topological structure, namely, to represent the global nonlinear structure characteristics according to the local linear fitting. LLE algorithm is a local weight matrix manifold learning algorithm with the assumption that observation data set is located in or approximately in the low-dimensional embedding manifold in the high-dimensional space. The basic idea of LLE is that the weight value of the data points in high-dimensional space can be best reconstructed and it can carry the local geometry information of manifold from high-dimensional space to low-dimensional space. LLE holds the opinion that constructed weight matrix can preserve the essence characteristics of the local neighborhood. That means the weight matrix can maintain the same geometric properties of local neighborhood of data set regardless of scaling or rotation.

The basic steps (described as Figure 1) of LLE algorithm are as follows:

Input. The original feature matrix $X_{D \times n}$ is composed of n numbers of D dimensional vector.

Output. The feature vector matrix $Y_{D' \times n}$ is composed of n numbers of D' dimensional vector ($D' \ll D$).

Step 1. Set the neighbor points (k) and (D) dimensional embedding values.

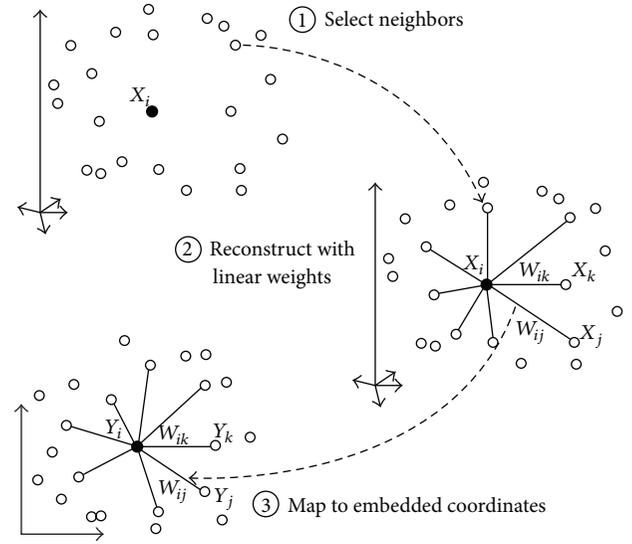


FIGURE 1: Steps of locally linear embedding algorithm.

Step 2. Calculate the Euclidean distance between any two points $(x_i, x_j$ among $i, j \in (1, 2, 3, \dots, n)$) in sample set $X = \{X_i, i = 1, 2, 3, \dots, n\} \subset R^m$, choosing the nearest neighbor corresponding points to constitute the neighbor area based on neighbor point.

Step 3. In the neighbor area, W_{ij} represents the weights between x_i and x_j , calculating the best reconstruction weights W_{ij} of X_i to make sure that reconstruction error ($W_{ij} = \sum_{i=1} |x_i - \sum_{j=1} w_{ij}x_j|^2$) is minimum and constraint conditions is $\sum_j W_{ij} = 1$.

Step 4. Calculate D' embedded dimensional vector $Y_{D' \times n}$ by the weight matrix W_{ij} based on the minimum reconstruction error $W_{ij} = \sum_{i=1} |x_i - \sum_{j=1} w_{ij}x_j|^2$.

In LLE algorithm, nonlinear data set is smartly divided into the data representation with local linear structure to reduce dimension effectively for nonlinear data. The main parameters are neighbor points and embedding dimension. The problem of solving the least squares is transformed into the eigenvalues in calculating process; thus it reduces the amount of calculation. In general, LLE has advantages of less undetermined parameters, overall analysis optimal solution, smaller computational complexity, and direct geometric meaning.

2.2. Locally Linear Embedding Algorithm of Homogenization Distance Algorithm. It is a certain difficulty when original LLE algorithm is used to deal with complicated data samples in the engineering practice. Because the LLE method assumes that the distribution of sample points on the manifold is continuous dense and uniform sampling, using Euclidean distance directly in high dimensional space to select local neighborhood may not truly reflect the intrinsic nature of the manifold structure. When manifold shows the curls or

cascade, namely, the short distance between the two manifold surfaces, refactoring process will cause points with shorter distance from different surface to the same local neighborhood space, thus causing the distortion of the manifold structure. In addition, the numbers of neighbor points (k) have obvious effects on dimension reduction, as the local neighborhood composed of (k) nearest neighbor points on the sparse area distribution of the sample points is obviously bigger than that on the dense area, while choosing neighbor points will have difference. Therefore, an improved local linear embedding algorithm of homogenization distance (HLL) is proposed. On the basis of the distance measurement between data in the Conformal-Isomap method and in accordance with the characteristics of practical fault samples, the following improving distance is used to replace the Euclidean distance in LLE algorithm:

$$D_{ij} = \frac{\|x_i - x_j\|}{\sqrt{M(i)M(j)}}. \quad (1)$$

Among them, $d_{ij} = \|x_i - x_j\|$ is the Euclidean distance between x_i and x_j , while $M(i)$ and $M(j)$, respectively, represent the average distance of x_i and x_j to the other points, namely,

$$M(i) = \frac{1}{N-1} \sum_{\substack{k=1 \\ k \neq i}}^N \|x_i - x_k\|, \quad (2)$$

$$M(j) = \frac{1}{N-1} \sum_{\substack{k=1 \\ k \neq j}}^N \|x_j - x_k\|.$$

The purpose of homogenization distance is getting the relatively narrow distance between sample points in the relatively sparse area and the relatively increased distance in the dense area through calculating improved distance between sample points, so that the overall distribution of sample points tends to be homogeneous, which is helpful to the classification of sample set; namely, the change of distance makes categories much more classifiable and reduces dimension reduction effects of the neighbor points. Figure 2 is the effect diagram of distance measurement change.

The molecules of original Euclidean distance are constant in the homogenization distance formula (1); when sample points are distributed in sparse area, the distance between the point and other points is longer, and the average distance with other points is also longer. Therefore, the original distance divided by the larger denominator to get new distance is reduced accordingly. On the contrary, when sample points are distributed in dense area, the distance between the point and other points is shorter, and the average distance with other points is also shorter, which makes the new distance be increased accordingly. As a result, homogenization distance makes the overall distribution of sample points tend to be homogeneous. Additionally, the distribution area between two kinds of sample points also changes accordingly. The new distance in sparse area becomes shorter, while sparse area becomes downsized and dense area becomes relatively

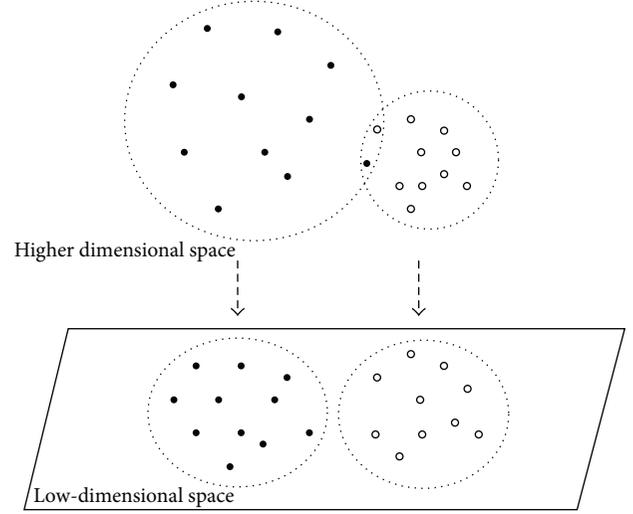


FIGURE 2: Effect diagram of distance measurement change.

upsized. Through distance formula, the difference of the new distance of sample points in the sparse and dense area can be released, namely, the ratio of the density.

(1) Sparse area:

$$A = \frac{d_{ij}}{D_{ij}} = \frac{\|x_i - x_j\|}{\|x_i - x_j\| / \sqrt{M(i)M(j)}} = \sqrt{M(i)M(j)}. \quad (3)$$

(2) Dense area:

$$B = \frac{d_{ij}}{D'_{ij}} = \frac{\|x_i - x_{j'}\|}{\|x_i - x_{j'}\| / \sqrt{M(i)M(j')}} = \sqrt{M(i)M(j')}. \quad (4)$$

Thereinto, x_j and $x_{j'}$ are neighbor point of x_i . If x_i is the same point, x_j and $x_{j'}$ are neighboring point in sparse area and dense regions, respectively, and the ratio of the density is as follows:

$$\frac{A}{B} = \frac{\sqrt{M(i)M(j)}}{\sqrt{M(i)M(j')}} = \frac{M(j)}{M(j')} > 1. \quad (5)$$

From formula (5), the change range ability of new distance in sparse area is bigger than that in dense area, so that sparse area can be separated from the dense one to a certain degree. In the subsequent algorithm process, the homogenization distance which chooses neighbor points according to k neighbor points can basically take the same kind of neighborhood. Thus, it can reduce the influence of the neighboring points and be more conducive to feature extraction. And gathering samples with similar features makes the samples classified effectively.

3. Locally Linear Embedding Algorithm of Supervision and Homogenization Distance

3.1. Supervised Locally Linear Embedding Algorithm. Locally linear embedding method is a higher efficient manifold method in nonlinear data dimension reduction method. But it is a nonlinear dimension reduction method without supervised learning in essence. The inadequate use of the samples category information results in a certain influence on the classification accuracy, which will not achieve an optimal effect if it is applied to areas such as classification. Therefore, de Ridder et al. [14] proposed SLLE algorithm [14]. Its main purpose is to increase the distance between classes and minimize the global reconstruction error of local data by reducing the distance within class. The key is to modify the calculation method of (k) neighbor points though adding a distance parameter to the different sample, thus increasing the category information of sample points. In SLLE algorithm, the following formula instead of Euclidean distance in LLE algorithm is used to build the neighborhood of sample points set for dimension reduction:

$$\Delta' = \Delta + \alpha \max(\Delta) \Lambda_{ij}. \quad (6)$$

From formula (6), Δ' is the distance that contains category information, Δ is the Euclidean distance without category information, and $\max(\Delta)$ is the maximum distance between classes. Λ_{ij} is the matrix whose value is 0 or 1. When two sample points belong to the same kind, the value is 0; otherwise the value is 1. α is a parameter to adjust the distance between point sets, $\alpha \in [0, 1]$, and α is an experienced parameter. When its value is 0, SLLE algorithm is equivalent to LLE algorithm.

3.2. Locally Linear Embedding Algorithm of Supervision and Homogenization Distance Algorithm. Locally linear embedding algorithm of homogenization distance by the change of distance makes categories much more classifiable. But the improvement of its classification performance is limited, and it is also a kind of unsupervised algorithm. In order to promote the fault recognition rate and stability of method, supervised learning mechanism is introduced to the HLLE, and then locally linear embedding algorithm of supervision and homogenization distance algorithm (SHLLE) is proposed. On the basis of homogenization distance, to increase the category information of sample points by supervised learning, so that the same fault category are gathered and heterogeneous fault are scattered. The main steps of method are as follows.

Step 1. Set neighbor points k , embedding dimension d , and parameter values $\alpha \in [0, 1]$.

Step 2. Calculate the homogenization distance. For a given data set $X = \{x_i, i = 1, 2, \dots, N\} \subset R^m$, calculate the Euclidean distance between sample points, and according to formula (1) receive the homogenization distance.

Step 3. Add sample information and select local neighbors. According to formula $D'_{ij} = D_{ij} + \alpha \max(D_{ij})\Lambda_{ij}$ to get D'_{ij} ,

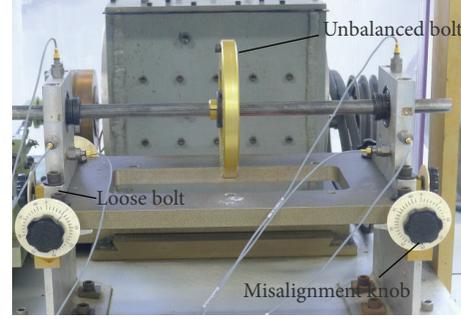


FIGURE 3: Rotor system fault simulation experiment.

look for the k ($k < N$) nearest neighbor points of each sample point in high-dimensional space.

Step 4. Calculate and reconstruct weight matrixes. Calculate locally optimal reconstruction weights of sample points and make the reconstruction error minimized. That is, acquire the optimal solution

$$\min \quad \varepsilon(W) = \sum_{i=1}^N \left\| x_i - \sum_{j=1}^N w_{ij} x_j \right\|_2^2 \quad (7)$$

$$\text{s.t.} \quad \sum_{j=1}^k w_{ij} = 1.$$

Among them, x_j ($j = 1, 2, \dots, k$) is neighbor point of x_i , and w_{ij} is the weights between x_i and x_j . When x_i and x_j are not neighbor, $w_{ij} = 0$.

Step 5. Calculate low-dimensional embedding matrix Y . From the above steps, weight matrix W is obtained which is used to get the optimal low-dimensional embedding matrix $Y = \{y_1, y_2, \dots, y_N\}^T$, by minimizing the reconstruction error and using function $\min \Phi(Y) = \sum_{i=1}^N \|y_i - \sum_{j=1}^N w_{ij} y_j\|_2^2$. Matrix Y needs to meet $\sum_{i=1}^N y_i = 0$, $(1/N) \sum_{i=1}^N y_i y_i^T = I$. Among them I is N dimensional unit matrix. The optimization problem is transformed into the following constrained optimization problem:

$$\begin{aligned} \min \quad \Phi(Y) &= \sum_{i=1}^N \|Y I_i - Y W_i\|^2 \\ &= \sum_{i=1}^N \|Y (I_i - W_i)\|^2 = \min \text{tr} Y M Y^T \end{aligned} \quad (8)$$

$$\text{s.t.} \quad Y Y^T = I.$$

Among them, $M = (I - W^T)(I - W)$ is solved to $M Y^T = \lambda Y^T$ by Lagrange multiplier method. Select the eigenvectors which are corresponded with d minimum nonzero eigenvalues of M that is the requested low-dimensional coordinate matrix Y . As usual, the minimum eigenvalue is almost zero. Taking

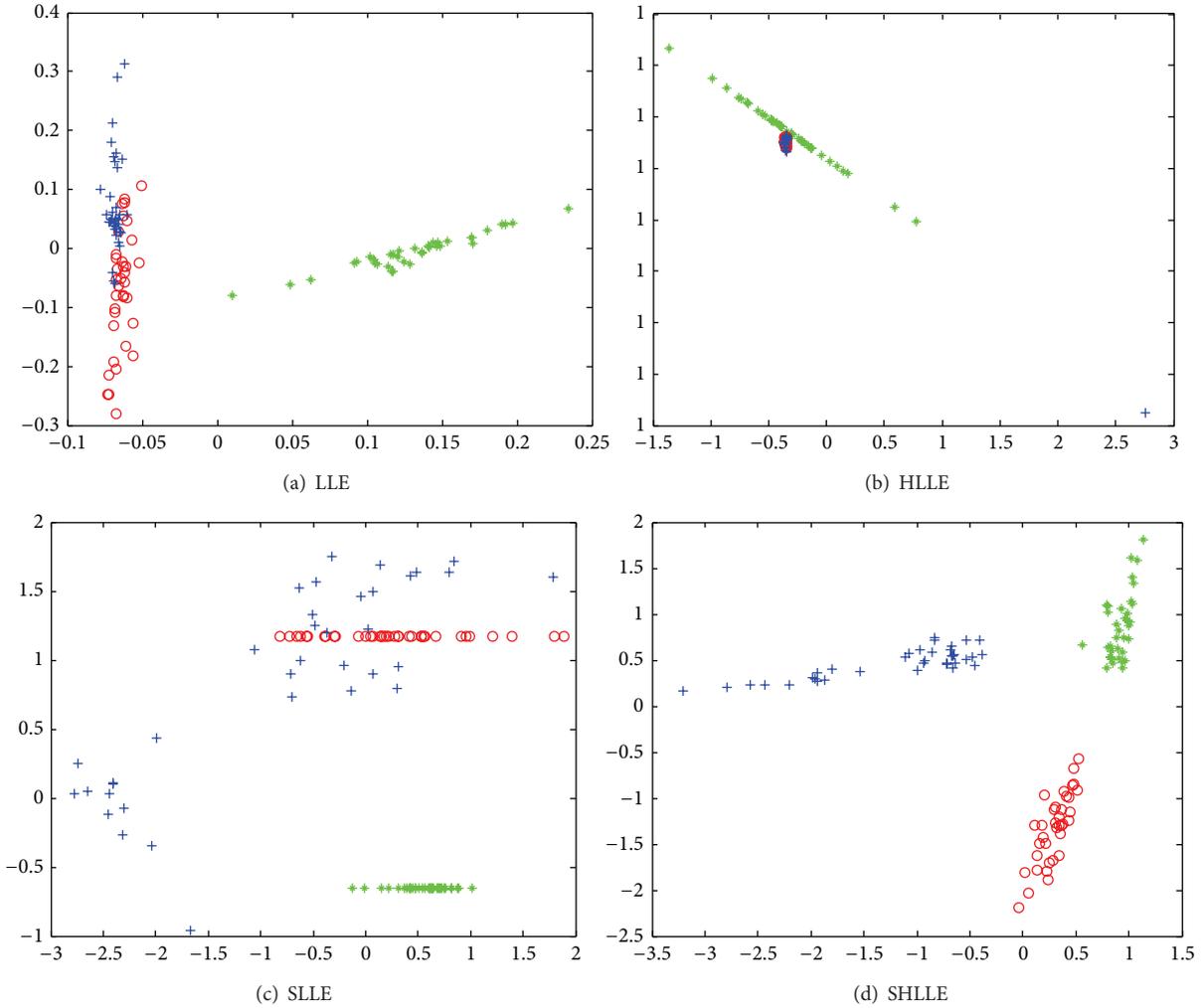


FIGURE 4: Fault recognition effect of data 1.

the eigenvectors (u_1, u_2, \dots, u_d) corresponded with the eigenvalues between $2 \sim (d + 1)$, so the optimal embedding results are obtained.

4. Simulation Experiment of Rotor System Fault

Comprehensive fault simulation test bed of Spectra Quest Company (USA) is used as experimental platform. Specifically, fault simulation experiment system is composed of Spectra Quest integrated fault simulation test bench and PULSE data acquisition system. As shown in Figure 3, the rotor disk contains two laps of a total of 36 threaded holes, in which the bolt can be randomly installed and can be used to simulate the rotor unbalance fault, and the quality of bolt is 5.596 g in the experiment. Adjusting two same knobs of the plinth to control both ends in the relative position of the axial and radial bearing can simulate rotor misalignment fault, 0.025 mm/scale. In the experiment, the left clockwise rotation for 5 scales is 0.125 mm, and the right end clockwise rotation for 10 scales is 0.25 mm. For the pedestal loose fault

of the experiment, it can be simulated to loosen the left end part of the bolt.

Through the simulation of the fault of the rotor system which is normal, unbalanced, misalignment, and loose in the pedestal at the running speed of 10 Hz, 20 Hz, and 30 Hz and using a total of six acceleration sensors at two bearing seats on three directions to carry out the vibration signal acquisition, a total of 144 groups of data signal were obtained.

5. Fault Diagnosis Based on SHLLE

Merging and reconstructing raw signal data, through analysis and comparison, a set of data 1 is shown as vector index of fault feature which could represent normal, misalignment, and unbalanced fault. Another group of data 2 is shown as vector index of fault feature which represent loose, misalignment, and unbalanced fault. Then two groups of the data space are filtered and extract 8 time domain parameters to form the original featured space which can be rolled into two $48 * 108$ data matrices.

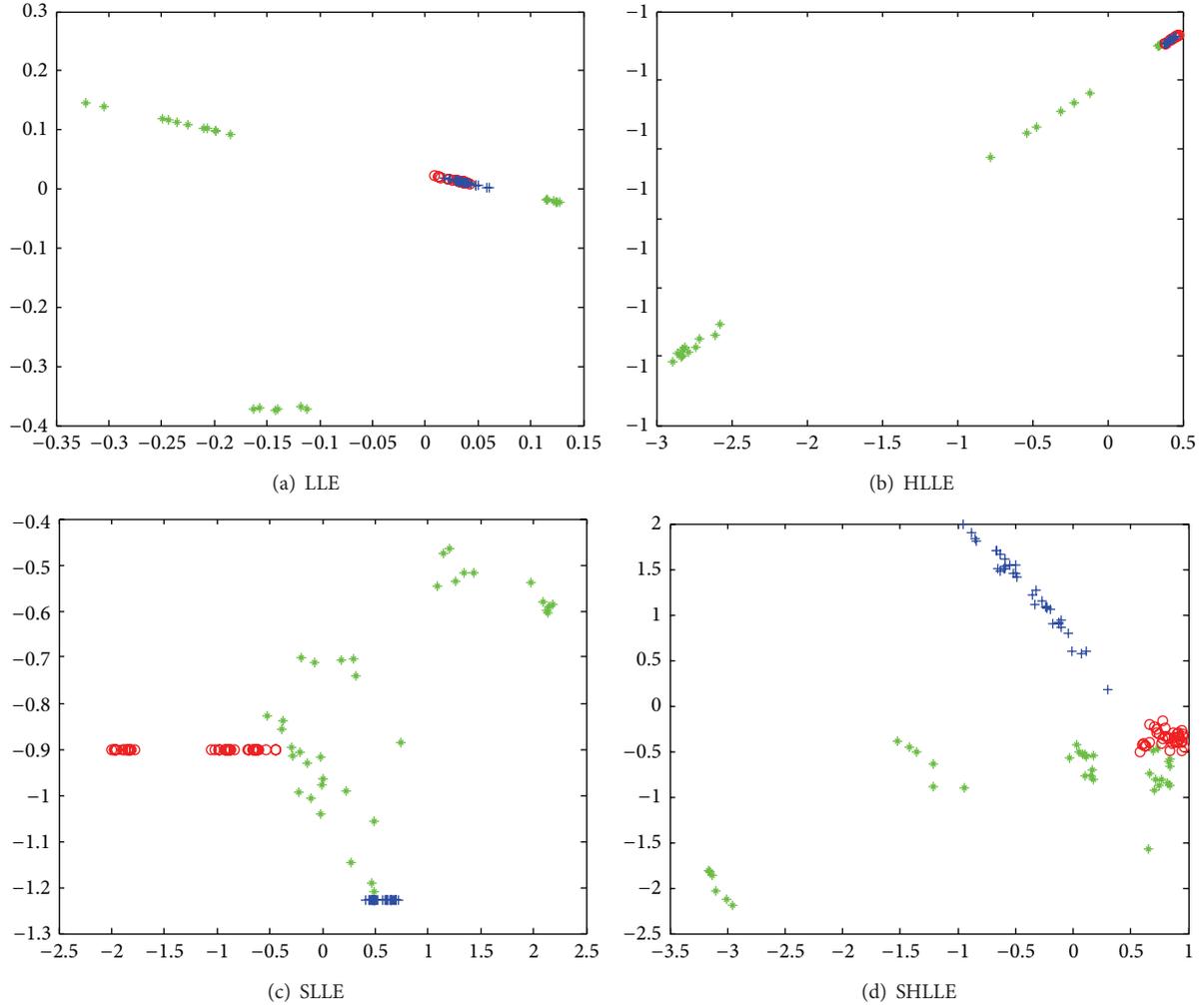


FIGURE 5: Fault recognition effect of data 2.

The analysis and comparison of classification and results of fault diagnosis between LLE, HLLE, SLE, and SHLLE are as follows.

5.1. Fault Identify and Comparison. As shown in Figures 4 and 5, (a), (b), (c), and (d) are two-dimensional classification map of LLE, HLLE, SLE, and SHLLE algorithm for original feature datasets, while horizontal and vertical axes represent the component 1 and component 2 of the main characteristics, respectively.

In Figure 4, in data 1, “*” indicates normal state, “O” indicates misalignment fault, and “+” indicates unbalanced fault. To know, (1) when using LLE and HLLE methods, normal state can be better separated, and overlapping phenomenon exists about unbalanced and misalignment faults. (2) When using SLE method and setting parameter $\alpha = 0.5$, the separated data set is gathered into columnar shape. Normal data set is identified, but misalignment and unbalanced fault are partially overlapped. (3) As for the SHLLE method, setting parameter $\alpha = 0.5$, normal, misalignment, and unbalanced

fault can be significantly classified and aggregated. It has better recognition effect compared with LLE and SLE.

In Figure 5, “*” represents loose fault; “O” indicates misalignment fault; “+” indicates unbalanced fault. To know, (1) when using LLE and HLLE methods, loose fault can be separated but not gathered, because of its impact that it is difficult to separate misalignment fault from unbalanced fault. (2) When using SLE method and setting parameter $\alpha = 0.5$, misalignment and unbalanced fault can be better separated and gathered without the effect of loose fault, but loose fault still cannot be gathered. (3) When using SHLLE method and also setting parameter $\alpha = 0.5$, normal, misalignment, and unbalanced fault can be effectively classified, and fault recognition effect is relatively better. In short, there is great superiority when using SHLLE to handle data sets of original features which is more effective on classification and identification of faults.

5.2. Comparison of Fault Recognition Rate. Two-dimensional map of data set according to various algorithms cannot fully reflect effect of fault recognition, because some methods also

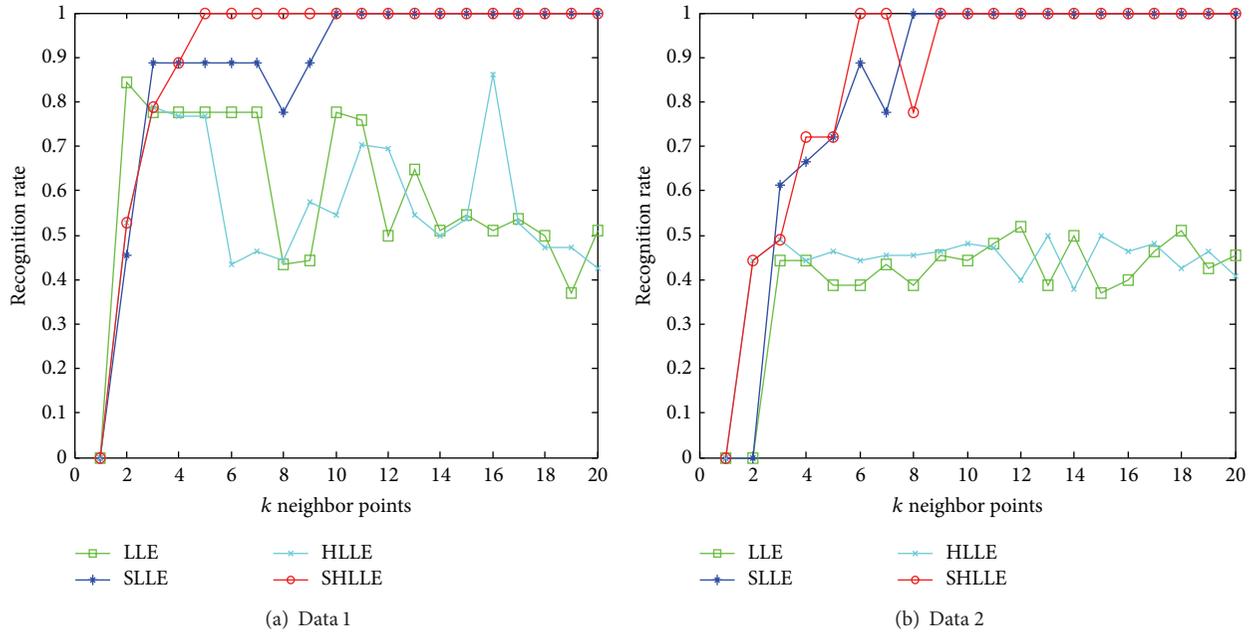


FIGURE 6: Recognition rate based on LLE, HLL, SLLE, and SHLLE.

can be used to classify and identify the fault when the number of neighborhood is small. Therefore, a comparative analysis of the recognition rate changing with the neighbor points was shown below.

Figure 6(a) is the changeable map for normal, misalignment, and unbalanced data sets (data 1) fault diagnosis accuracy rate with neighboring points of change in k based on LLE, HLL, SLLE, and SHLLE algorithms. Taking the embedding dimension $d = 2$, $\alpha = 0.5$. “Box” Line, “*” Line, “x” Line, and “O” Line represent a recognition rate curve of LLE, SLLE, HLL, and SHLLE, respectively. It can be obtained according to the diagram and the data: about 80% recognition rate of LLE and HLL is achieved with smaller neighbor points, which shows an unstable state of fluctuation obviously with the increase of k ; recognition rate of SLLE algorithm shows a higher trend which reaches 100% and remains stable when k values 10, which is suitable for the case where neighbor number is more than 10; SHLLE algorithm has great advantages of recognition rate of normal, misalignment, and unbalanced faults which reaches 100% and keeps stable when k values 5.

Figure 6(b) is the changeable map for loose fault, misalignment, and unbalanced data sets (data 2) fault diagnosis accuracy rate with neighboring points of change in k based on LLE, HLL, SLLE, and SHLLE algorithms. Take the embedding dimension $d = 2$, $\alpha = 0.5$. “Box” Line, “*” Line, “x” Line, and “O” Line represent a recognition rate curve of LLE, SLLE, HLL, and SHLLE, respectively. To know, loose fault still has some bad impacts on the effect of LLE and HLL algorithm, and even the purpose of fault diagnosis cannot be achieved; recognition rate of SLLE algorithm increases in general which shows unstable fluctuation when k is smaller. Recognition rate of SLLE algorithm reaches 100% and keeps

stable when k is equal to 8, which is still suitable for the case where the neighbor number is larger.

SHLLE algorithm still has a soft spot for recognition rate of loose, misalignment, and unbalanced faults which reaches 100% firstly and keeps stable after identifying one case of fluctuation when k value is equal to 8.

In conclusion, fault diagnosis based on SHLLE algorithm has superior performance compared to the other LLE algorithm. SLLE algorithm is suitable for fault identification of which the neighborhood is slightly larger, and it is relatively stable. However, SHLLE algorithm has optimal performance, of which fault identification is more stable than others.

6. Conclusion

The paper researches on rotating machinery. In order to get better recognition effect, LLE of homogenization distance (HLL) and LLE of supervision and homogenization distance (SHLLE) are proposed. Proving the validity of the fault diagnosis by simulating rotor system failure experiment, the following conclusions are reached.

- (1) In two-dimensional map of each algorithm for two types of data set, overlapping phenomenon exists between unbalanced fault and misalignment fault when using LLE, HLL, and SLLE methods. However, SHLLE has a strong advantage and is more effective in fault classification.
- (2) In the map with the changes of neighbor points of each algorithm for two types of data set, SLLE algorithm is suitable for the fault diagnosis when the number of neighbor points is slightly larger. However, the fault diagnosis of SHLLE algorithm has superior

performance compared to the other LLE algorithm, even fault identification is more stable.

Conflict of Interests

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

Acknowledgments

Financial support from National Natural Science Foundation of China (51175170), The Industrial Cultivation Program of Scientific and Technological Achievements in Higher Educational Institutions of Hunan Province (10CY008), Aid Program for Science and Technology Innovative Research Team in Higher Educational Institutions of Hunan province are gratefully acknowledged.

References

- [1] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [2] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
- [3] Z. Y. Zhang and H. Y. Zha, "Nonlinear dimension reduction via local tangent space alignment," in *Proceedings of the 4th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL '03)*, Hong Kong, March 2003.
- [4] Z. Y. Zhang and H. Y. Zha, "Principal manifolds and nonlinear dimension reduction via local tangent space alignment," *SIAM Journal on Scientific Computing*, vol. 26, no. 1, pp. 313–338, 2004.
- [5] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.
- [6] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Advances in Neural Information Processing Systems*, MIT Press, Whistler, Canada, 2002.
- [7] X. F. He, S. C. Yan, Y. X. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.
- [8] J. Nilsson, "Nonlinear dimensionality reduction of gene expression data," Printed in by KFS, Lund, Sweden, 2006.
- [9] H. Chang and D.-Y. Yeung, "Locally linear metric adaptation with application to semi-supervised clustering and image retrieval," *Pattern Recognition*, vol. 39, no. 7, pp. 1253–1264, 2006.
- [10] J. Yang, J. Xu, D. Yang, and M. Li, "Noise reduction method for nonlinear time series based on principal manifold learning and its application to fault diagnosis," *Chinese Journal of Mechanical Engineering*, vol. 42, no. 8, pp. 154–158, 2006.
- [11] Q. Yang, G. Chen, X. Tong, and Q. He, "Application of incremental local tangent space alignment algorithm to rolling bearings fault diagnosis," *Journal of Mechanical Engineering*, vol. 48, no. 5, pp. 81–86, 2012.
- [12] L. Liang, G. Xu, M. Li, Y. Zhang, and X. Liang, "Nonlinear manifold learning method of mechanical impact faults extraction," *Journal of Xi'an Jiaotong University*, vol. 43, no. 11, 2009.
- [13] M. Li, S. Wang, and L. Liang, "Feature extraction for incipient fault diagnosis of rolling bearings based on nonlinear manifold learning," *Journal of Xi'an Jiaotong University*, vol. 44, no. 5, pp. 45–49, 2010.
- [14] D. de Ridder, O. Kouropteva, O. Okun et al., "Supervised locally linear embedding," in *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP-2003*, vol. 2714 of *Lecture Notes in Computer Science*, pp. 333–341, Springer, Berlin, Germany, 2003.

Review Article

A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications

Yudong Zhang,¹ Shuihua Wang,^{1,2} and Genlin Ji¹

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*

Correspondence should be addressed to Yudong Zhang; zhangyudongnuaa@gmail.com

Received 15 December 2014; Revised 12 February 2015; Accepted 12 February 2015

Academic Editor: Shuming Wang

Copyright © 2015 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.

Particle swarm optimization (PSO) is a heuristic global optimization method, proposed originally by Kennedy and Eberhart in 1995. It is now one of the most commonly used optimization techniques. This survey presented a comprehensive investigation of PSO. On one hand, we provided advances with PSO, including its modifications (including quantum-behaved PSO, bare-bones PSO, chaotic PSO, and fuzzy PSO), population topology (as fully connected, von Neumann, ring, star, random, etc.), hybridization (with genetic algorithm, simulated annealing, Tabu search, artificial immune system, ant colony algorithm, artificial bee colony, differential evolution, harmonic search, and biogeography-based optimization), extensions (to multiobjective, constrained, discrete, and binary optimization), theoretical analysis (parameter selection and tuning, and convergence analysis), and parallel implementation (in multicore, multiprocessor, GPU, and cloud computing forms). On the other hand, we offered a survey on applications of PSO to the following eight fields: electrical and electronic engineering, automation control systems, communication theory, operations research, mechanical engineering, fuel and energy, medicine, chemistry, and biology. It is hoped that this survey would be beneficial for the researchers studying PSO algorithms.

1. Introduction

Artificial intelligence (AI) is the intelligence exhibited by machines. It is defined as “the study and design of intelligent agents” [1], where an intelligent agent represents a system that perceives its environment and takes action that maximizes its success chance. AI research is highly technical and specialized and is deeply divided into subfields that often fail to communicate with each other. Currently popular approaches of AI include traditional statistical methods [2], traditional symbolic AI, and computational intelligence (CI) [3]. CI is a fairly new research area. It is a set of nature-inspired computational methodologies and approaches to address complex real-world problems to which traditional approaches are ineffective or infeasible. CI includes artificial neural network (ANN), fuzzy logic, and evolutionary computation (EC).

Swarm intelligence (SI) is a part of EC. It researches the collective behavior of decentralized, self-organized systems, natural or artificial. Typical SI systems consist of a population of simple agents or boids interacting locally with one another

and with their environment. The inspiration often comes from nature, especially biological systems [4].

The agents in a SI system follow very simple rules. There is no centralized control structure dictating how individual agents should behave. The agents' real behaviors are local, and to a certain degree random; however, interactions between such agents lead to the emergence of “intelligent” global behavior, which is unknown to the individual agents. Well-known examples of SI include ant colonies, bird flocking, animal herding, bacterial growth, and fish schooling.

Dorigo [5] proposed an ant colony optimization (ACO) method based on ant colony. Kennedy and Eberhart [6] proposed a particle swarm optimization (PSO) method based on bird flocking. Those are two most famous SI-based optimization algorithms. In addition to them, scholars have shown great interest in proposing new intelligent approaches. Storn and Price [7] proposed a differential evolution (DE). Müller et al. [8] and Passino [9] proposed the bacterial foraging optimization (BFO), inspired by the group foraging behavior of bacteria such as *E. coli* and *M. xanthus*. Karaboga

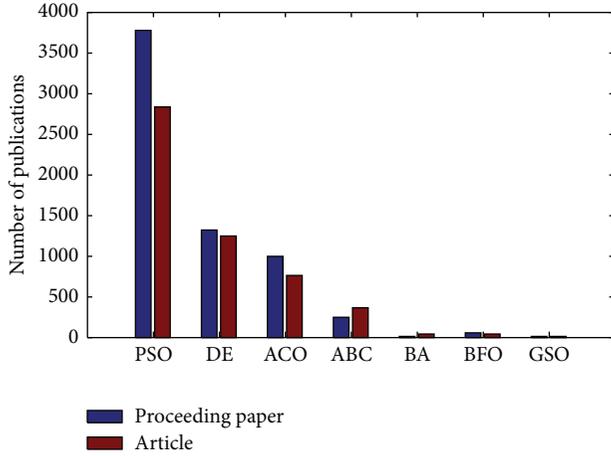


FIGURE 1: Number of all publications w.r.t. SI-based algorithms.

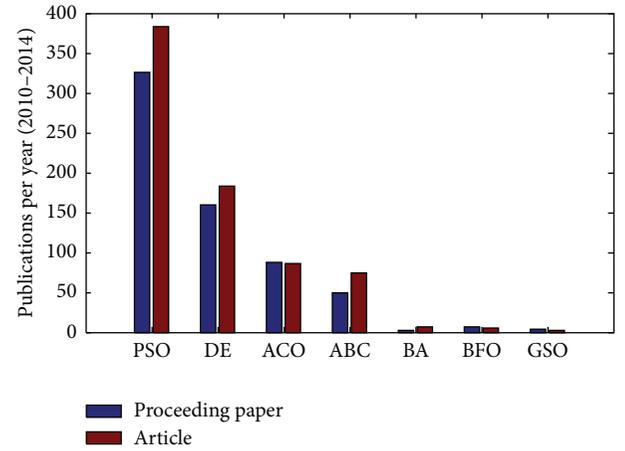


FIGURE 2: Publication per year (2010-2014) w.r.t. SI-based algorithms.

and Basturk [10] proposed artificial bee colony (ABC), which simulates the foraging behavior of honey bees. Krishnanand and Ghose [11] proposed glowworm swarm optimization (GSO) method, the agents in which are thought of as glowworms that carry a luminescence quantity called luciferin along with them. Yang [12] proposed a bat algorithm (BA), inspired by the echolocation behavior of microbats.

The distribution of all publications and publication per year w.r.t. SI-based algorithms is presented in Figures 1 and 2, respectively. As seen from the figures, the number of total publications related to PSO is even higher than the sum of six other algorithms, and the number of publication per year related to PSO is the highest among all seven SI-based algorithms. This suggests PSO is the most prevalent SI-based optimization algorithms. Therefore, we center this review on PSO.

Several public websites related to PSO (Table 1) were set up [13], dedicated to share the codes, ideas, and latest advances on PSO. There are several types of source codes, written in different programming languages, in those websites. In addition, many publications about PSO and its applications were presented.

This work first checked the coherency of PSO with principles required by SI. Second, we reviewed the studies on advances of PSO. Third, various applications of PSO is given. Finally, we conclude the paper by summarizing the improvements and analyzing potential research directions. This survey was carried out mainly by examining the “Web of Science Core Collection” database. In addition, IEEE Explorer and Google Scholar were also used.

2. Particle Swarm Optimization: PSO Approach

2.1. Features of Self-Organization. Self-organization is a key feature of SI system. It is a process where global order or coordination arises out of the local interactions between the components of an initially disordered system. This process is

spontaneous; that is, it is not controlled by any agent inside or outside of the system. Bonabeau et al. [14] interpreted the self-organization in swarms through three basic ingredients as follows.

- (1) Strong dynamical nonlinearity (often involving positive and negative feedback): positive feedback helps promote the creation of convenient structures, while negative feedback counterbalances positive feedback and helps to stabilize the collective pattern.
- (2) Balance of exploitation and exploration: SI identifies a suitable balance to provide a valuable mean artificial creativity approach.
- (3) Multiple interactions: agents in the swarm use information coming from neighbor agents so that the information spreads throughout the network.

2.2. Features of SI. In addition, Millonas [15] in Santa Fe Institute proposed five principles that SI must satisfy. They are proximity principle, quality principle, diverse response principle, stability principle, and adaptability principle. Their meanings are listed in Table 2.

2.3. Algorithmic Structure of Standard PSO. PSO performs searching via a swarm of particles that updates from iteration to iteration. To seek the optimal solution, each particle moves in the direction to its previously best (*pbest*) position and the global best (*gbest*) position in the swarm [16]. One has

$$\begin{aligned}
 pbest(i, t) &= \arg \min_{k=1, \dots, t} [f(P_i(k))], \quad i \in \{1, 2, \dots, N_p\}, \\
 gbest(t) &= \arg \min_{\substack{i=1, \dots, N_p \\ k=1, \dots, t}} [f(P_i(k))],
 \end{aligned} \tag{1}$$

where i denotes the particle index, N_p the total number of particles, t the current iteration number, f the fitness

TABLE 1: Public websites of PSO.

Website name	URL
Particle swarm optimization	http://www.swarmintelligence.org/
Particle swarm optimization	http://www.scholarpedia.org/article/Particle_swarm_optimization
Particle swarm central	http://www.particleswarm.info/
PSO toolbox	http://psotoolbox.sourceforge.net/
PSO visualization	http://www.projectcomputing.com/resources/psovis/
Particle swarm optimization toolbox	http://atoms.scilab.org/toolboxes/PSO

TABLE 2: Five basic principles of SI.

Principle	Definition
Proximity principle	The swarm should be able to do simple space and time computations
Quality principle	The swarm should be able to respond to quality factors in the environment
Diverse response principle	The swarm should not commit its activities along excessively narrow channels
Stability principle	The swarm should not change its mode of behavior every time the environment changes
Adaptability principle	The swarm should be able to change its behavior mode when it is worth the computational price

function, and P the position. The velocity V and position P of particles are updated by the following equations:

$$V_i(t+1) = \omega V_i(t) + c_1 r_1 (pbest(i, t) - P_i(t)) + c_2 r_2 (gbest(t) - P_i(t)), \quad (2)$$

$$P_i(t+1) = P_i(t) + V_i(t+1), \quad (3)$$

where V denotes the velocity, ω is the inertia weight used to balance the global exploration and local exploitation, r_1 and r_2 are uniformly distributed random variables within range $[0, 1]$, and c_1 and c_2 are positive constant parameters called “acceleration coefficients.”

It is common to set an upper bound for the velocity parameter. “Velocity clamping” [17] was used as a way to limit particles flying out of the search space. Another method is the “constriction coefficient” strategy, proposed by Clerc and Kennedy [18], as an outcome of a theoretical analysis of swarm dynamic, in which the velocities are constricted too.

The first part of formula (2), known as “inertia,” represents the previous velocity, which provides the necessary momentum for particles to roam across the search space. The second part, known as the “cognitive” component, represents the individual particle thinking of each particle. It encourages the particles to move toward their own best positions found so far. The third part, the “cooperation” component, represents the collaborative effect of the particles to find the global optimal solution [19].

2.4. Pseudocode of PSO. Let $f : \mathbb{R}^N \rightarrow \mathbb{R}$ be the cost function to be minimized. The function takes a candidate solution of a vector of N_p real numbers and produces a real number as output that indicates the cost function value. The gradient of f is either unknown or hard to calculate. The goal is to find the global minimal x^* (Pseudocode 1).

3. Studies on PSO

In this review, we center in reporting the advances on PSO in the form of formal publications. We divide advances into following six aspects:

- (i) modifications of PSO, including quantum-behaved PSO, bare-bones PSO, chaotic PSO, fuzzy PSO, PSOT-VAC, opposition-based PSO, topology, and other slight modifications,
- (ii) hybridization of PSO with other metaheuristic methods, including genetic algorithm (GA), artificial immune system (AIS), Tabu search (TS), ACO, simulated annealing (SA), ABC, DE, biogeography-based optimization (BBO), and harmonic search (HS),
- (iii) extensions of PSO to other optimization fields, including multiobjective, constrained, discrete, and binary optimization,
- (iv) theoretical analysis of PSO, parameter selection and convergence analysis.
- (v) parallel implementation of PSO, including multicore, GPU computing, and cloud computing.

3.1. Modifications

3.1.1. QPSO. Some researchers proposed quantum-behaved PSO (QPSO), which was motivated by concepts from quantum mechanics. For example, Jau et al. [20] proposed a modified QPSO, which used a high breakdown regression estimator and a least-trimmed-squares method to eliminate the influence caused by observations containing outliers. Besides, elitist crossover of GA and adaptive decay of SA are used for conquering premature and controlling search policy. Jamalipour et al. [21] presented QPSO with differential mutation operator (QPSO-DM) for optimizing WWER-1000 core fuel management. The results showed that QPSO-DM performs better than the others. Bagheri et al. [22] used QPSO to forecast financial time series, especially for

```

Step 1. Initialization
For each particle  $i = 1, \dots, N_p$ , do
  (a) Initialize the particle's position with a uniformly distribution as  $P_i(0) \sim U(\text{LB}, \text{UB})$ , where LB and UB represent the lower
  and upper bounds of the search space
  (b) Initialize  $pbest$  to its initial position:  $pbest(i, 0) = P_i(0)$ .
  (c) Initialize  $gbest$  to the minimal value of the swarm:  $gbest(0) = \text{argmin}_f[P_i(0)]$ .
  (d) Initialize velocity:  $V_i \sim U(-|\text{UB} - \text{LB}|, |\text{UB} - \text{LB}|)$ .
Step 2. Repeat until a termination criteria is met
For each particle  $i = 1, \dots, N_p$ , do
  (a) Pick random numbers:  $r_1, r_2 \sim U(0, 1)$ .
  (b) Update particle's velocity. See formula (2).
  (c) Update particle's position. See formula (3).
  (d) If  $f[P_i(t)] < f[pbest(i, t)]$ , do
    (i) Update the best known position of particle  $i$ :  $pbest(i, t) = P_i(t)$ .
    (ii) If  $f[P_i(t)] < f[gbest(t)]$ , update the swarm's best known position:  $gbest(t) = P_i(t)$ .
  (e)  $t \leftarrow (t + 1)$ ;
Step 3. Output  $gbest(t)$  that holds the best found solution.

```

PSEUDOCODE 1: A standard PSO.

the foreign exchange market. Tang et al. [23] proposed an improved QPSO algorithm for continuous nonlinear large-scale problems based on memetic algorithm and memory mechanism. The memetic algorithm was used to make all particles gain some experience through a local search before being involved in the evolutionary process, and the memory mechanism was used to introduce a “bird kingdom” with memory capacity, both of which can improve the global search ability of the algorithm. Davoodi et al. [24] proposed a new approach, based on a hybrid algorithm combining of improved QPSO and simplex algorithms. QPSO was the main optimizer of algorithm, which can give a good direction to the optimal global region. Nelder-Mead simplex method was used as a local search to fine-tune the obtained solution from QPSO. Li and Xiao [25] proposed an encoding approach based on qubits described on the Bloch sphere. Each particle contained three groups of Bloch coordinates of qubits, and all three groups of coordinates were regarded as approximate solutions describing the optimization result. Particles were updated using the rotation of qubits about an axis on the Bloch sphere. Yumin and Li [26] integrated artificial fish swarm to QPSO and used adaptive parameters to avoid premature. Jia et al. [27] proposed an enhanced QPSO based on GA to realize a synchronous optimization of sensor array and classifier. Gholizadeh and Moghadas [28] proposed an improved QPSO metaheuristic algorithm to implement performance-based optimum design process. Two numerical examples were presented to illustrate the efficiency of the presented method.

3.1.2. BBPSO. The bare-bones PSO (BBPSO) [29] is a version of the PSO algorithm in which the velocity and position update rules are substituted by a procedure that samples a parametric probability density function. Zhang et al. [30] used both mutation and crossover operators of DE algorithm to modify original BBPSO in order to update certain particles in the population. The performance of the resulting algorithm

was tested on 10 benchmark functions and applied to 16 vapor-liquid equilibrium problems. Zhang et al. [31] analyzed the sampling distribution in BBPSO, based on which they propose an adaptive version inspired by the cloud model, which adaptively produced a different standard deviation of the Gaussian sampling for each particle according to the evolutionary state in the swarm, which provided an adaptive balance between exploitation and exploration on different objective functions. Zhang et al. [32] proposed three global optimization algorithms for phase and chemical equilibrium calculations, which played a significant role in the simulation, design, and optimization of separation processes in chemical engineering. The proposed algorithms were unified BBPSO (UBBPSO), integrated DE (IDE), and IDE without Tabu list and radius (IDE_N). Zhang et al. [33] proposed a new bare-bones multiobjective PSO algorithm to solve the environmental/EDPs. The algorithm had three distinctive features: a particle updating strategy that did not require tuning up control parameters, a mutation operator with action range varying over time to expand the search capability, and an approach based on particle diversity to update the global particle leaders. Blackwell [34] formulated the dynamic update rule of PSO as a second-order stochastic difference equation. General relations were derived for search focus, search spread, and swarm stability at stagnation. The relations were applied to three particular PSO implementations: the standard PSO, a PSO with discrete recombination, and the BBPSO. Wang et al. [35] proposed a novel hybrid algorithm, called SM-MBBPSO, based on the Nelder-Mead simplex method (SM) and a modified BBPSO (MBBPSO). A new strategy based on K -means clustering was proposed to combine the powerful global search capability of MBBPSO and the high accurate local search capability of SM. This made the proposed algorithm achieve a nice balance between exploitation and exploration capability. Meanwhile, an adaptive reinitialization strategy on inactive particles was proposed to help the swarm get away from local optimal positions. Jiang and Wang [36] used

cooperative coevolution (CC) to improve the performance of PSO on clustering high-dimensional datasets. Based on CC framework, the original partitioned clustering problem was first decomposed to several subproblems, each of which was then evolved by an optimizer independently. BBPSO was employed as the optimizer to solve each subproblem cooperatively. In addition, a new centroid-based encoding schema was designed for each particle, and the Chernoff bounds were applied to decide a proper population size. Liu et al. [37] proposed a novel disruption strategy, originating from astrophysics, to shift the abilities between exploration and exploitation during the search process, with the aim of enhancing population diversity and speeding up convergence rate of BBPSO. They researched the distribution and diversity on the proposed disruption operator and illustrated the position relationship between the original and disrupted position. Campos et al. [38] proposed a variant of BBPSO with scale matrix adaptation (SMA), SMA-BBPSO for short reference, to address the drawback of premature convergence and improve the performance of the BBPSO. The position of a particle was selected from a multivariate t -distribution with a rule for adaptation of its scale matrix. The multivariate t -distribution was used in its hierarchical form, as a scale mixture of normal distributions. In addition, the approach included the normal distribution as a particular case. As a consequence, the t -distribution could be applied during the optimization process by maintaining the proper balance between exploration and exploitation. Zhang et al. [39] proposed a binary BBPSO to find optimal feature subset, which was a useful preprocessing technique for solving classification problems. In this algorithm, a reinforced memory strategy was designed to update the local leaders of particles for avoiding the degradation of outstanding genes in the particles, and a uniform combination was proposed to balance the local exploitation and the global exploration of algorithm.

3.1.3. CPSO. Concepts related to chaos theory have been integrated with PSO to improve its performance. This type of PSO variant is called chaotic PSO (CPSO). Chuang et al. [40] introduced chaotic maps into catfish particle swarm optimization. The proposed method increased the search capability via the chaos approach. Zhang and Wu [41] proposed adaptive CPSO (ACPSO) to train the weights/biases of two-hidden-layer forward neural network in order to develop a hybrid crop classifier for polarimetric synthetic aperture radar images. Dai et al. [42] proposed a novel adaptive chaotic embedded PSO (ACEPSO) and applied it in wavelet parameters estimation. ACEPSO embedded chaotic variables in standard PSO and adjusted parameters nonlinearly and adaptively. It also estimated particles whether being focusing or discrete by judging the population fitness variance of particle swarm and average distance amongst points; then chaotic researching was applied to escaping from premature convergence. Li et al. [43] proposed a novel chaotic particle swarm fuzzy clustering (CPSFC) algorithm based on a new CPSO and gradient method. The new CPSO algorithm is the combination of adaptive inertia weight factor (AIWF) and iterative chaotic map with infinite

collapses (ICMIC) based chaotic local search. The CPSFC algorithm utilized CPSO to search the fuzzy clustering model, exploiting the searching capability of fuzzy c -means (FCM) and avoiding its major limitation of getting stuck at locally optimal values. Meanwhile, gradient operator is adopted to accelerate convergence of the proposed algorithm. Wu et al. [44] proposed a novel support vector regression machine (SVRM) and then developed a CPSO to estimate its unknown parameters. The results of two experiments demonstrate the feasibility of this approach. Zhang et al. [45] proposed a fitness-scaling adaptive chaotic PSO (FAC-PSO) approach as a fast and robust approach for the task of path planning of unmanned combat aerial vehicle (UCAV). The FAC-PSO employed the fitness-scaling method, the adaptive parameter mechanism, and the chaotic theory. Experiments showed that the FAC-PSO was more robust and cost less time than elite GA with migration, SA, and chaotic ABC. Zhang et al. [46] combined CPSO with K2 algorithm and applied the method to Bayesian network structure learning. Yang et al. [47] applied PSO with double-bottom chaotic maps (DBM-PSO) in order to assist statistical methods in the analysis of associated variations to disease susceptibility. Analysis results supported that the proposed DBM-PSO could identify good models and provided higher chi-square values than conventional PSO. Son [48] used CPSO to optimize municipal solid waste collection in GIS based environments and took Danang city, Vietnam, as a case study. He et al. [49] proposed a novel hybrid model combining ANN and CPSO to improve forecast accuracy. The proposed model was found to perform better for fine particles than for coarse particles. Zeng and Sun [50] combined classical PSO with a chaotic mechanism, time-variant acceleration coefficients, and a self-adaptive mutation scheme to prevent premature convergence and improve solution quality. Multiple efficient constraint handling strategies were employed to deal with complex constraints. Pluhacek et al. [51] utilized different chaotic systems as pseudorandom number generators (PRNGs) for velocity calculation in the PSO algorithm. Two chaos-based PRNGs were used alternately within one run of the PSO algorithm and dynamically switched over when a certain criterion was met.

3.1.4. FPSO. In order to make PSO more powerful, it was combined with fuzzy sets theory. This type of PSO variant is called fuzzy PSO (FPSO). Juang et al. [52] proposed an adaptive FPSO (AFPSO) algorithm. The proposed AFPSO utilized fuzzy set theory to adjust PSO acceleration coefficients adaptively and was thereby able to improve the accuracy and efficiency of searches. Incorporating this algorithm with quadratic interpolation and crossover operator further enhanced the global searching capability to form a new variant called AFPSO-Q1. Alfi and Fateh [53] presented a novel improved FPSO (IFPSO) algorithm to the intelligent identification and control of a dynamic system. The proposed algorithm estimated optimally the parameters of system and controller by minimizing the mean of squared errors. The PSO was enhanced intelligently by using a fuzzy inertia weight to rationally balance the global and local exploitation

abilities. Every particle dynamically adjusted inertia weight according to particles best memories using a nonlinear fuzzy model. Yang et al. [54] proposed a novel FPSO algorithm based on fuzzy velocity updating strategy in order to optimize the machining parameters. The proposed FPSO algorithm achieved good results on few benchmark problems and obtained significant improvements on two illustrative case studies of multipass face milling. Norouzzadeh et al. [55] proposed a light adaptive PSO, which was a novel method that used a fuzzy control system to conduct the standard algorithm. The suggested method used two adjunct operators along with the fuzzy system in order to improve the base algorithm on global optimization problems. Robati et al. [56] studied an extension of PSO algorithm, the balanced fuzzy PSO algorithm, which was used for fundamental optimization problem entitled traveling salesman problem (TSP). Khan and Engelbrecht [57] presented a multiobjective PSO to efficiently solve the distributed local area networks (DLAN) topology design problem. Fuzzy logic was incorporated in the PSO algorithm to handle the multiobjective nature of the problem. Unified “And-Or” operator was used to aggregate the objectives. Results suggest that the fuzzy PSO is a suitable algorithm for solving the DLAN topology design problem. Galzina et al. [58] described the application of a hybrid of fuzzy logic and PSO in order to achieve suboptimal solutions for flow-shop scheduling problem. They named the proposed method adaptive fuzzy PSO. Nafar et al. [59] proposed a combination of FPSO and ACO method to estimate the parameters of Metal Oxide Surge Arrester (MOSA) models. The proposed method was named modified FPSO (MFPSO). The inertia weight was tuned by using fuzzy rules. Aminian and Teshnehlab [60] introduced a novel FPSO method in which the inertia weight as well as the cognitive and social coefficients was adjusted for each particle separately according to the information coming from a fuzzy logic controller. Chai et al. [61] used Hilbert-Huang transform for the features extractor and FPSO with cross mutated-based ANN (FPSOCM-ANN), for the classification of a three-class mental task-based brain-computer interface (BCI).

3.1.5. PSOTVAC. PSO with time-varying acceleration coefficients (TVAC) was proposed to further improve the performance of standard PSO. The new variant was termed PSOTVAC. Cai et al. [62] considered linear automation strategy may not work well in many cases. Therefore, a new variant, predicted modified PSO with time-varying accelerator coefficients, was proposed, in which the social and cognitive learning factors were adjusted according to a predefined predicted velocity index. The mechanism lied in that the large cognitive coefficient provided a large local search capability, whereas the small one employed a large global search capability. Chaturvedi et al. [63] employed PSOTVAC to solve the practical economic dispatch problem (EDP). TVAC here was to efficiently control the local and global search so that premature convergence was avoided and global solutions were achieved. Boonyaritdachochoai et al. [64] proposed an optimal congestion management approach in a deregulated electricity market using PSOTVAC. Initially,

the values of generator sensitivity were used to select redispatched generators. PSOTVAC was used to determine the minimum redispatch cost. Sun et al. [65] presented a comparative analysis of PSO, self-organizing hierarchical PSO (HPSO), and self-organizing hierarchical PSO with time-varying acceleration coefficients (HPSO-TVAC) for data clustering. They found that the HPSO and the HPSO-TVAC algorithms had better performance than the PSO algorithm in most cases, and all the clustering algorithms using PSO had good performance for large-scale data and high-dimensional data, over six well-known benchmarks. Abedinia et al. [66] presented an efficient approach for solving economic load dispatch (ELD) problems in different test power systems using PSOTVAC. Accordingly, for practical operation, many realistic constraints as ramp rate limits, generation limitation, prohibited operating zone, transmission loss, and nonlinear cost functions were considered. Mohammadi-Ivatloo et al. [67] presented a novel heuristic algorithm for solving EDP by employing iteration PSO with time-varying acceleration coefficients (IPSO-TVAC) method. EDP may even be more complicated if transmission losses were taken into account. Numerical results showed that the IPSO-TVAC method had a good convergence property. Mohammadi-Ivatloo et al. [68] implemented a novel time-varying acceleration coefficients PSO (TVAC-PSO) algorithm to solve combined heat and power ED (CHPED). The acceleration coefficients in PSO algorithm were varied adaptively during iterations to improve solution quality of original PSO and avoid premature convergence. Pookpant and Ongsakul [69] proposed a binary PSO (BPSO) with TVAC for solving optimal placement of wind turbines within a wind farm. The objective was to extract the maximum turbine power output in a minimum investment cost within a wind farm. The BPSO-TVAC algorithm was applied to 100-square-cell test site considering uniform wind and nonuniform wind speed with variable direction characteristics. Linear wake model was used to calculate downstream wind speed. Abedinia et al. [70] presented a hybrid PSO with time-varying acceleration coefficients (HPSOTVAC) and bacteria foraging algorithm (BFA) for solving a complex ELD problem. The effectiveness of the proposed HPSOTVAC/BFA was tested in 6-, 15-, and 40-unit generating systems. Abdullah et al. [71] proposed a modified PSO with TVAC (MPSO-TVAC) for solving ELD problem. To improve the solution quality and robustness of PSO algorithm, a new best neighbor particle called “*rbest*” was proposed. The *rbest* provided extra information for each particle that was randomly selected from other best particles in order to diversify the movement of particle and avoid premature convergence. Chih et al. [72] adapted standard PSO and proposed two novel PSO algorithms, namely, the binary PSO with TVAC (BPSOTVAC) and the chaotic binary PSO with TVAC (CBPSOTVAC), to solve the multidimensional knapsack problem (MKP). The results showed that the proposed algorithms were superior over the other methods according to success rate, mean absolute deviation, mean absolute percentage error, least error, and standard deviation.

3.1.6. OPSO. Opposition-based learning (OBL) theory was integrated with PSO, and the new variant was dubbed opposition-based PSO (OPSO). Dhahri and Alimi [73] proposed the OPSO using the concept of opposite number to create a new population during the learning process. They combined OPSO with BBFNN. The results showed that the OPSO-BBFNN produced a better generalization performance. Wang et al. [74] presented an enhanced PSO algorithm called GOPSO, which employed generalized OBL (GOBL) and Cauchy mutation. GOBL provided a faster convergence and the Cauchy mutation with a long tail helped trapped particles escape from local optima. Dong et al. [75] proposed an evolutionary circle detection method based on a novel chaotic hybrid algorithm (CHA). The method combined the strengths of PSO, GA, and chaotic dynamics and involved the standard velocity and position update rules of PSOs, with the ideas of selection, crossover, and mutation from GA. The OBL was employed in CHA for population initialization. In addition, the notion of species was introduced into the proposed CHA to enhance its performance in solving multimodal problems. Gao et al. [76] proposed a novel PSO called CSPSO to improve the performance of PSO on complex multimodal problems. Specifically, a stochastic search technique was used to execute the exploration in PSO. In addition, to enhance the global convergence, when producing the initial population, both opposition-based learning method and chaotic maps were employed. The numerical simulation and comparisons with some typical existing algorithms demonstrated the superiority of the proposed algorithm. Khan et al. [77] presented a new discrete PSO approach to induce rules from discrete data. The proposed algorithm, called Opposition-Based Natural Discrete PSO (ONDPSO), initialized its population by taking into account the discrete nature of the data. Particles were encoded using a Natural Encoding scheme. Each member of the population updated its position iteratively based on a newly designed position update rule. OBL was implemented in the optimization process. The encoding scheme and position update rule used by the algorithm allowed individual terms corresponding to different attributes within the rule's antecedent to be a disjunction of the values of those attributes. Kaucic [78] presented a multistart PSO algorithm for the global optimization of a function subject to bound constraints. The procedure consisted of three main steps. In the initialization phase, an OBL strategy was performed to improve the search efficiency. Then a variant of the adaptive velocity based on the differential operator enhanced the optimization ability of the particles. Finally, a reinitialization strategy based on two diversity measures for the swarm was acted in order to avoid premature convergence and stagnation. Dai et al. [79] established a mathematical model to study the motions of ships in order to control them effectively. They proposed an algorithm based on PSO and the OBL theory, known as the opposition-based particle swarm optimization (OPSO). Muñoz et al. [80] described how adequate hardware implementations of the PSO algorithm can be useful for real-time adaptation of mobile robot controllers. For achieving this goal, they proposed a new architecture, which was based on an FPGA implementation of the OBL approach applied

to the PSO and which explored the intrinsic parallelism of this algorithm in order to adjust the weights of a neural robot controller in real time according to desired behaviors.

3.1.7. SPSO. In contrast, some researchers objected to those researches that made PSO more and more complex, and they tended to simplify standard PSO without impairing its performance, with the aim of reducing computation time, improving convergence performance, or making it easier to implement. For example, Guochu [81] divided the swarm into three categories denoted as better particles, ordinary particles, and the worst particles, according to the fitness values. These three types of particles evolved dynamically according to three corresponding kinds of simplified algorithm models. The results showed that simplified PSO (SPSO) had better optimization performance than other improved PSOs. Pederesen and Chipperfield [82] simplified PSO method in order to increase its adaptability and used an overlaid metaoptimizer for efficiently tuning behavior parameters. The simplified version was dubbed Many Optimizing Liaisons (MOL). Experiments showed that MOL had comparable performance with PSO. Martins et al. [83] proposed a simplified PSO, which allowed saving some computational effort and obtained a considerable performance in the optimization of nonlinear functions. The method was tested by four nonlinear benchmark functions: Sphere, Schwefel, Schaffer, and Ackley. Panda et al. [84] presented the design and performance analysis of proportional-integral derivative (PID) controller for an automatic voltage regulator system using MOL. The superiority of MOL was shown by comparing the results with ABC, PSO, and DE. Vastrakar and Padhy [85] proposed a simplified PSO with proportional-integral proportional-derivative (PI-PD) controller for unstable processes. Complete search space was divided into small subsearch spaces. In each subspace, they calculated the global minima and local minima and took the minimum of all, that is, minimum of complete search space. Yeh [86] proposed a parameter-free simplified swarm optimization to adjust the weights in ANNs.

3.1.8. Topology. The premature of PSO can be avoided by not using the entire swarm's best known position *gbest* but just the best position of the local area around the particle that is moved. In such case, the PSO variant is said to be local best *lbest*. Further, suppose there is an information link between each particle and its neighbors, and the set of those links builds a graph, which is called the topology of PSO variant. Scholars have carried out numerous researches in this field. Wang and Watada [87] proposed a hybrid mutation-neighborhood-based PSO (MN-PSO) which comprised the approximation algorithm to search for the approximate optimal solution. Jiang et al. [88] proposed PSO with age-group topology (PSOAG), a novel age-based PSO. They presented a new concept of age to measure the search ability of each particle in local area. To keep population diversity during searching, particles were separated to different age-groups by their age. Particles in each age-group could only select the ones in younger groups or their own groups as their neighborhoods. To allow search escape from local optima,

the aging particles were regularly replaced by new and randomly generated ones. In addition, an age-group based parameter setting method was designed, where particles in different age-groups had different parameters to accelerate convergence. Marinakis and Marinaki [89] introduced a new algorithmic nature-inspired approach that used PSO with different neighborhood topologies, for successfully solving one of the most computationally complex problems, the permutation flow-shop scheduling problem (PFSP). The proposed algorithm combined a PSO algorithm, the variable neighborhood search strategy, and a path relinking strategy. Rada-Vilela et al. [90] presented a performance study on the effect of synchronicity in communications and neighborhood size in PSO on large-scale optimization problems. The algorithms under study were the Synchronous PSO (S-PSO), the Asynchronous PSO (A-PSO), and the recently proposed Random Asynchronous PSO (RA-PSO). Wang et al. [91] proposed a hybrid PSO algorithm called DNSPSO, which employed a diversity enhancing mechanism and neighborhood search strategies to achieve a trade-off between exploration and exploitation abilities. Comparison results showed that DNSPSO obtained a promising performance on the majority of the test problems. Fu et al. [92] proposed a new QPSO algorithm called NQPSO, in which one local and one global neighborhood search strategies were utilized to balance exploitation and exploration. Moreover, a concept of opposition-based learning was employed for population initialization. Computational results showed that the proposed approach outperformed some similar QPSO algorithms and five other state-of-the-art PSO variants. Ni and Deng [93] proposed to use random topology and analyzed its performance. The relationship between population topology and the performance of PSO was also explored from the perspective of graph theory characteristics in population topologies. Further, in logistic dynamic particle optimization, an extensive simulation study was presented to discuss the effectiveness of the random topology and the design strategies of population topology. Beheshti et al. [94] proposed an improved PSO scheme called fusion global-local-topology PSO (FGLT-PSO). The algorithm employed both global and local topologies in PSO to jump out of the local optima. The experimental results showed that the proposed method improved the performance of PSO algorithm in terms of solution accuracy and convergence speed. Lim and Isa [95] proposed PSO with increasing topology connectivity (PSO-ITC) to solve unconstrained single-objective optimization problems with continuous search space. An ITC module was developed to achieve better control of exploration/exploitation searches by linearly increasing the particle's topology connectivity with time as well as performing the shuffling mechanism. Furthermore, they introduced a new learning framework that consisted of a new velocity update mechanism and a new neighborhood search operator that aimed at enhancing the algorithm's searching performance. Kalayci and Gupta [96] proposed a new approach based on the PSO algorithm with a neighborhood-based mutation operator to solve the sequence-dependent disassembly line balancing problem.

3.1.9. Other Modifications. Some researchers make tentative research on improving the optimization performance of PSO by other efficient strategies. For example, Chuang et al. [97] proposed a novel catfish PSO, the mechanism of which is dependent on the incorporation of a catfish particle into the linearly decreasing weight particle swarm optimization. Unlike other ordinary particles, the catfish particles initialized a new search from the extreme points of the search space when the *gbest* fitness value had not been changed for a given time, which resulted in further opportunities to find better solutions for the swarm by guiding the whole swarm to promising new regions of the search space and accelerating convergence. Shi and Liu [98] proposed a hybrid improved PSO, in which chaos initialization was introduced to improve the population diversity, and adaptive parameters' control strategy was employed to make it independent from specific problem. Besides, novel acceptance policy based on Metropolis rule was taken to guarantee the convergence of the algorithm. Zhang et al. [99] proposed a new adaptive PSO (APSO) that could dynamically follow the frequently changing market demand and supply in each trading interval. A numerical example served to illustrate the essential features of the approach. Liu et al. [100] proposed an improved cooperative PSO to solve both the local extrema and the pseudominimum problem, with the aim of solving production scheduling more efficiently. The results showed that the convergent speed and solution quality of the improved cooperative PSO preceded the other two efficient algorithms. Shen et al. [101] presented a correlation PSO model in which a novel correlative strategy was used to process the personal experience and sharing experience. The relational expression between the correlation coefficient and population diversity was developed through theoretical analysis. They found that the processing strategy with positive linear correlation was helpful to maintain the population diversity. Lin et al. [102] introduced a jumping-out strategy named crown jewel defense (CJD). CJD was used to relocate the *gbest* position and reinitialized all particles' *pbest* position when the swarm was trapped in local optima. Taking the advantage of CJD strategy, the swarm could jump out of the local optimal region without being dragged back and the performance of PSO became more robust to the initialization. Experimental results on benchmark functions showed that the CJD-based PSOs were comparable to or better than the other representative state-of-the-art PSOs. Wang and Watada [103] studied a facility location model with fuzzy random parameters and its swarm intelligence approach. A Value-at-Risk (VaR) based fuzzy random facility location model (VaR-FRFLM) was built in which both the costs and demands were assumed to be fuzzy random variables, and the capacity of each facility was unfixed but a decision variable assuming continuous values. A hybrid modified PSO approach was proposed to solve the VaR-FRFLM. Li et al. [104] developed a knowledge-based heuristic PSO approach with the adjustment strategy (KHPSOA), inspired by the No Free Lunch Theorem to solve the weighted circle packing problem. The numerical experiments showed that KHPSOA was superior to the existing algorithms in the performances. Inspired by the ecological behavior, Lu et al. [105] developed an augmented PSO (AugPSO) algorithm using two new

strategies, boundary shifting and particle position resetting, with the aim of optimizing the design of truss structures. The boundary shifting approach forced particles to move to the boundary between feasible and infeasible regions in order to increase the convergence rate in searching. The particle position resetting approach was motivated by mutation scheme in GA to increase the diversity of particles and to prevent the solution of particles from falling into local minima. Mattos et al. [106] presented a constrained hybrid PSO algorithm in order to find feasible solutions to the resource allocation problem. Wu et al. [107] integrated some problem-oriented knowledge into the design of a certain PSO variant. They investigated the inner variable learning (IVL) strategy which could help the particle to inspect the relation among its inner variables, determine the exemplar variable for all other variables, and then make each variable learn from the exemplar variable in terms of their quantitative relations. Hence, a novel PSO algorithm with an IVL strategy was proposed and found particularly efficient for optimizing functions with symmetric variables. In addition, they designed a new trap detection and jumping-out strategy to help particles escape from local optima. Lim and Mat Isa [108] presented an adaptive two-layer PSO algorithm with elitist learning strategy (ATLPSO-ELS), which had better search capability than classical PSO. In ATLPSO-ELS, they performed evolution on both the current swarm and the memory swarm, motivated by the tendency of the latter swarm to distribute around the problem's optima. They proposed two adaptive division-of-labor modules to self-adaptively divide the swarms into exploration and exploitation sections. An elitist learning strategy module was introduced in the proposed algorithm to enhance the search efficiency of swarms and to mitigate premature convergence. Shimizu et al. [109] developed a novel algorithm of PSO associated with binary decision variables. It was quite effective for finding the optimum opening distribution centers in three-echelon logistic network by parallel computing. Eventually, they had implemented the procedure in the parallel algorithm deployed as a multi-population based approach using multithread programming technique. Fister Jr. et al. [110] presented a simple PSO, which allowed automatic creation of complex two-dimensional graphic characters. The method involved constructing the base characters, optimizing the modifications of the base characters with the PSO algorithm, and finally generating the graphic characters from the solution. They demonstrated the effectiveness of the approach with the creation of simple snowman.

3.2. Hybridization. PSO was combined with some traditional and evolutionary optimization algorithms in order to take the advantages of both methods and compensate the weaknesses of each other. This type of PSO is called hybridized PSO.

3.2.1. With GA. Kuo and Hong [111] presented a two-stage method of investment portfolio based on soft computing techniques. The first stage used data envelopment analysis to select most profitable funds, while hybrid of GA and PSO was proposed to conduct asset allocation in the second stage. Chen and Kurniawan [112] presented a two-stage

optimization system to find optimal process parameters of multiple quality characteristics in plastic injection molding. Taguchi method, BPNN, GA, and combination of PSO and GA (PSO-GA) were used in this study to find optimum parameter settings. Nazir et al. [113] extracted facial local features using local binary pattern (LBP) and then fused these features with clothing features, which enhanced the classification accuracy rate remarkably. In the following step, PSO and GA were combined to select the most important features' set that more clearly represented the gender and thus the data size dimension was reduced. Vidhya and Kumar [114] proposed a hybrid technique that included PSO and GA for channel estimation in MIMO-orthogonal frequency division multiplexing (MIMO-OFDM) systems. The result showed the performance of the proposed method was better than LS and MMSE methods in all the mutation and crossover values and in all the iterations computed. Xiao et al. [115] constructed three different types of neural network based models, that is, Elman network, generalized regression neural network (GRNN), and wavelet neural network (WNN) constructed by three nonoverlapping training sets. Their empirical results suggested the ensemble ANNs-PSO-GA approach significantly improved the prediction performance over other individual models and linear combination models. Ghamisi and Benediktsson [116] proposed a new feature selection approach based on the integration of a GA and PSO. The overall accuracy of a support vector machine (SVM) classifier on validation samples was used as a fitness value. The new approach was carried out on the well-known Indian Pines hyperspectral dataset. Results confirmed that the new approach was able to select automatically the most informative features in terms of classification accuracy within an acceptable processing time without requiring the number of desired features to be set a priori by users.

3.2.2. With AIS. Tang et al. [117] presented a novel dynamic PSO algorithm based on improved artificial immune network (IAINPSO). Based on the variance of the population's fitness, a kind of convergence factor was adopted in order to adjust the ability of search. The experimental results showed that not only did the new algorithm satisfy convergence precision, but also the number of iterations was much less than traditional scheme and had much faster convergent speed, with excellent performance in the search of optimal solution to multidimensional function. Zhang et al. [118] proposed a more pragmatic model for stochastic networks, which considered not only determinist variables but also the mean and variances of random variables. In order to accelerate the solution of the model, they integrated PSO with chaos operator and AIS. Ibrahim et al. [119] developed a power quality monitor positioning algorithm to find the optimal number and placement of PQMs in both transmission and distribution systems. First, the concept of topological monitor reach area was introduced. Then the binary PSO hybridized with AIS was used to solve multiobjective function in finding the optimal placement of PQMs. Kuo et al. [120] intended to propose a hybrid of AIS and PSO-based SVM (HIP-SVM) for optimizing SVM parameters and applied it to radio

frequency identification (RFID) based positioning system. The computational results showed that HIP-SVM had better performance than AIS-based SVM and PSO-based SVM. Liu et al. [121] proposed a coevolutionary PSO algorithm associating with the artificial immune principle. In the proposed algorithm, the whole population was divided into two kinds of subpopulations consisting of one elite subpopulation and several normal subpopulations. The best individual of each normal subpopulation will be memorized into the elite subpopulation during the evolution process. Darzi et al. [122] incorporated PSO, dynamic mutated AIS (DM-AIS), and gravitational search algorithm (GSA) into the existing LCMV technique in order to improve the weights of LCMV. The simulation result demonstrated that received signal to interference and noise ratio (SINR) of target user can be significantly improved by the integration of PSO, DM-AIS, and GSA in LCMV through the suppression of interference in undesired direction.

3.2.3. With TS. Li et al. [123] integrated nonlinear simplex method (NSM) into PSO in order to increase its convergence speed. They also integrated TS into PSO to assign Tabu attribute to local solution regions. The hybrid PSO algorithm was an organic composition of the PSO, NSM, and TS algorithms. Nakano et al. [124] presented a new form of PSO based on the concept of TS. The proposed Tabu list PSO (TL-PSO) was a method for combining the strong points of PSO and TS. This method stored the history of *pbest* in a Tabu list. When a particle had a reduced searching ability, it selected a *pbest* of the past from the historical values, which was used for the update. This made each particle active, and the searching ability of the swarm made progress. Zhang et al. [125] presented the production planning management architecture for iron-steel manufacturing factories based on make-to-order and make-to-stock management ideas. In order to solve this nonlinear integer program, the authors designed a hybrid PSO and TS algorithm, in which new heuristic rules were proposed to repair infeasible solutions. Ktari and Chabchoub [126] proposed a new heuristic approach such that various features inspired from the TS were incorporated in the Essential Particle Swarm Optimization queen (EPSOq) algorithm in order to obtain another improved discrete PSO version. Wang et al. [127] focused on long-term distribution system maintenance scheduling aided by available operation information. A combined algorithm that consisted of PSO and TS was designed and applied to the optimization problem. Numerical result verified that the proposed method could schedule long-term maintenance of distribution systems in smart grid economically and effectively.

3.2.4. With ACO. Chen and Chien [128] presented a new method, called the genetic simulated annealing ant colony system with particle swarm optimization techniques, for solving the TSP. The experimental results showed that both the average solution and the percentage deviation of the average solution to the best known solution of the proposed method were better than existing methods. Xiao et al. [129] considered the features of the MRCMPSP problem. They

employed ant colony's labor division to establish a task priority-scheduling model firstly. Then, they used improved PSO to find out the optimum scheduling scheme. The approach integrating the above two algorithms had abilities of both local search and global search. Kiran et al. [130] proposed a new hybrid method for estimating energy demand of Turkey using PSO and ACO. PSO was developed for solving continuous optimization problems; ACO was used for discrete optimizations. Hybrid method based PSO and ACO was developed to estimate energy demand using gross domestic product, population, import, and export. Huang et al. [131] incorporated ACOR with PSO to improve the search ability, investigating four types of hybridization as follows: (1) sequence approach, (2) parallel approach, (3) sequence approach with an enlarged pheromone-particle table, and (4) global best exchange. Among the four strategies of hybridization, the sequence approach with the enlarged pheromone table was superior to the other approaches because the enlarged pheromone table diversified the generation of new solutions of ACOR and PSO, which prevented traps into the local optimum. Rahmani et al. [132] used a new hybrid swarm technique (HAP) to forecast the energy output of a real wind farm located in Binaloud, Iran. The technique consisted of the hybridization of the ACO and PSO, which were two metaheuristic techniques under the category of swarm intelligence. The hybridization of the two algorithms to optimize the forecasting model led to a higher quality result with a faster convergence profile. Elloumi et al. [133] illustrated a novel optimization approach based on multiobjective PSO (MOPSO) and Fuzzy ACO (FACO). The basic idea was to combine these two techniques using the best particle of the Fuzzy Ant algorithm and integrate it as the best local PSO to formulate a new approach called hybrid MOPSO with FACO (H-MOPSO-FACO). This hybridization solved the multiobjective problem, which relied on both time performance criteria and the shortest path.

3.2.5. With SA. Sait et al. [134] proposed a hybrid of PSO and SA for solving the cell assignment in Cnnos\nano-wire\MOLEcular Hybrid (CMOL). Results showed that the proposed hybrid algorithm achieved better solution in terms of buffer count in reasonable time. Jiang and Zou [135] proposed an improved parameter optimization method based on traditional PSO algorithm by changing the fitness function in the traditional evolution process of SVMs. Then, this PSO method was combined with SA global searching algorithm to avoid local convergence that traditional PSO algorithms usually run into. This method achieved better results that reflected in the ROC curves in medical images classification and gained considerable identification accuracy in clinical disease detection. Niknam et al. [136] proposed a hybrid PSO and SA (PSO-SA) method to solve the dynamic optimal power flow (DOPF) problem while the prohibited zones, valve-point effects, and ramp rate constraints were taken into account. The hybrid PSO-SA algorithm could do an efficient search and explore solution space, while it profited from privileges of both PSO and SA algorithms. Khoshahval et al. [137] developed a new parallel optimization algorithm, P-PSOSA,

for performing the fuel management optimization; they defined two different fitness function considering the multiplication factor maximizing and power peaking factor minimizing objectives simultaneously. Numerical results of P-PSOSA confirmed that the proposed algorithm had a great strength to obtain a near global core pattern with respect to considered objective functions during suitable consuming runtime. Du et al. [138] presented a hybrid algorithm based on improved PSO and SA (IPSOSA) algorithm to solve the resource constrained multiproject scheduling problem. Aimed at overcoming the shortcomings of premature convergence of standard PSO, adaptive inertia weight with cyclical attenuation strategy and SA were employed in the hybrid algorithm. Zhang et al. [139] proposed an improved approach to decompose structuring elements of an arbitrary shape. They introduced in the restarted simulated annealing PSO method, which was the combination of restarted SA and PSO. Geng et al. [140] introduced robust v -support vector regression (RSVR) model to forecast port throughput. In order to search the more appropriate parameters combination for the RSVR model, they presented a chaotic simulated annealing PSO (CSAPSO) algorithm to determine the parameter combination.

3.2.6. With ABC. El-Abd [141] tested a hybrid PSO and ABC algorithm on the CEC13 testbed. The hybridization technique was a component-based one, where the PSO algorithm was augmented with an ABC component to improve the personal best of the particles. Sharma et al. [142] proposed a variant called Local Global variant ABC (LGABC) to balance the exploration and exploitation in ABC. The proposal harnessed the local and global variant of PSO into ABC. The proposed variant was investigated on a set of thirteen well-known constrained benchmarks problems and three chemical engineering problems, which showed that the variant can get high-quality solutions efficiently. Kiran and Gündüz [143] presented a hybridization of PSO and ABC approaches, based on recombination procedure. The global best solutions obtained by the PSO and ABC were used for recombination, and the solution obtained from this recombination was given to the populations of the PSO and ABC as the global best and neighbor food source for onlooker bees, respectively. Information flow, between particle swarm and bee colony, helped increase global and local search abilities of the hybrid approach. Vitorino et al. [144] put forward a mechanism based on the ABC to generate diversity when all particles of the PSO converged to a single point of the search space. Then, the swarm entities switched between two predefined behaviors by using fuzzy rules depending on the diversity of the whole swarm.

3.2.7. With DE. Maione and Punzi [145] proposed a two-step design approach. First, DE determined the fractional integral and derivative actions satisfying the required time-domain performance specifications. Second, PSO determined rational approximations of the irrational fractional operators as low-order, stable, minimum-phase transfer functions with poles interlacing zeros. Extensive time- and frequency-domain simulations validated the efficiency of the proposed

approach. Fu et al. [146] presented a hybrid DE with QPSO for the unmanned aerial vehicle (UAV) route planning on the sea. It combined DE algorithm with the QPSO algorithm in an attempt to enhance the performance of both algorithms. Experimental results demonstrated that the proposed method was capable of generating higher quality paths efficiently for UAV than any other tested optimization algorithms. Vasundhara et al. [147] presented an efficient way of designing linear-phase finite impulse response (FIR) low-pass and high-pass filters using a novel algorithm ADEPSO, which was hybrid of fitness based adaptive DE and PSO. ADEPSO overcame the above individual disadvantages faced by both algorithms and was used for the design of linear-phase low-pass and high-pass FIR filters. The simulation results showed that the ADEPSO outperformed PSO, ADE, and DE in combination with PSO not only in magnitude response but also in the convergence speed and thus proved itself to be a promising candidate for designing the FIR filters. Yu et al. [148] formulated a novel adaptive hybrid algorithm based on PSO and DE (HPSO-DE) by developing a balanced parameter between PSO and DE. Adaptive mutation was carried out on current population when the population clustered around local optima. The HPSO-DE enjoyed the advantages of PSO and DE and maintained diversity of the population. Compared with PSO, DE, and their variants, the performance of HPSO-DE was competitive. Wang et al. [149] proposed a robust hybrid metaheuristic optimization approach by adding DE mutation operator to the accelerated PSO (APSO) algorithm to solve numerical optimization problems. Yadav and Deep [150] proposed a new co-swarm PSO (CSHPSO) for constrained optimization problems, which was obtained by hybridizing shrinking hypersphere PSO (SHPSO) with the DE approach. The total swarm was subdivided into two subswarms in such a way that the first subswarms used SHPSO and second subswarms used DE. Experiments showed that CSHPSO was a promising new co-swarm PSO which could be used to solve any real constrained optimization problem.

3.2.8. With Other Approaches. Xu et al. [151] proposed a hybrid PSO integrated with trust region method. The simulation results on some multimodal global optimizations showed that the algorithm was far more effective than linear decreasing weight PSO (LDWPSO) and sequential quadratic programming (SQP) to search the global optimum. Mohanty et al. [152] presented a study on frequency regulation in isolated hybrid DG system. H-infinite loop shaping based on PSO as well as hybrid PSO and HS (PSOHS) controller was proposed to minimize the frequency deviation. Guo et al. [153] employed PSO and BBO to propose a hybrid algorithm termed biogeography-based PSO (BPSO), which could make a large number of elites effective in searching optima. The whole population was split into several subgroups; BBO was employed to search within each subgroup and PSO for the global search.

3.3. Extensions. The success of the PSO algorithm as a single-objective optimizer within continuous search space

has motivated researchers to extend its use to other areas, which consist of but not limited to multiobjective optimization, constrained optimization, binary optimization, discrete optimization, combinatorial optimization, and so forth.

3.3.1. Multiobjective Optimization. Multiple objective PSO (MOPSO) has been proposed to deal with multiobjective optimization problems, in which the objective function took Pareto dominance into account when moving the PSO particles and nondominated solutions were stored so as to approximate the Pareto front. Qiu et al. [154] proposed a MOPSO with a new *gbest* selection strategy. They used *K*-means algorithm and proportional distribution based approach to select *gbest* from the archive for each particle of the population. A symmetric mutation operator was incorporated to enhance the exploratory capabilities. The simulation results indicated that the proposed algorithm was highly competitive in terms of convergence and diversity in comparison with several state-of-the-art algorithms. Chen et al. [155] presented an ironless permanent magnet linear brushless motor with three objective functions: maximal thrust force, minimal temperature, and minimal volume. An elitist hybrid QPSO algorithm with mutation was used to solve this multiobjective optimization problem. Elitist mechanism with crowding distance sorting was used to improve the number and diversity of the solutions. Ghanei et al. [156] presented application of thermal-economic multiobjective optimization of shell and tube heat exchanger (STHE) using MOPSO to obtain the maximum effectiveness (heat recovery) and the minimum total cost as two objective functions. Duan et al. [157] developed a mathematical model based on thermodynamic analysis of Stirling engine considering regenerative losses and internal irreversibilities. Power output, thermal efficiency, and the cycle irreversibility parameter of Stirling engine were optimized simultaneously using MOPSO. Amiryousefi et al. [158] performed a multiobjective optimization for deep-fat frying of ostrich meat plates. MOPSO was used to obtain the best solutions. This problem had three objective functions that must be satisfied simultaneously. Results showed a Pareto where all the points on this Pareto were the best possible solutions. Ganguly [159] presented a PSO-based multiobjective planning algorithm for reactive power compensation of radial distribution networks with unified power quality conditioner (UPQC) allocation. The optimal location, the optimal reactive power compensation required at the location, and the optimal design parameters of UPQC were determined by minimizing three objective functions: the rating of UPQC, network power loss, and percentage of nodes with undervoltage problem. These objectives were simultaneously minimized to obtain a set of nondominated solutions using MOPSO. Zhang et al. [160] combined BBPSO and sensitivity-based clustering for solving multiobjective reliability redundancy allocation problems (RAPs). A two-stage process was performed to identify promising solutions. A new bare-bones MOPSO (BBMOPSO) was developed and applied in the first stage to identify a Pareto-optimal set. This algorithm mainly differed from other MOPSO algorithms in the parameter-free particle updating strategy,

which was especially suitable for handling the complexity and nonlinearity of RAPs. Perera et al. [161] applied MOPSO to identify intermediate debonding damage in the problem of fiber-reinforced-polymer-composites- (FRP-) plated RC beams. The use of permanently installed fiber Bragg grating sensors embedded into the FRP concrete interface or bonded onto the FRP strip together with the proposed methodology resulted in an automated method able to operate in an unsupervised mode. Cheng et al. [162] proposed an improved MOPSO with preference strategy (IMPESO-PS) and applied it to the optimal integration of distributed generation (DG) into the distribution system. Preference factors were introduced to quantify the degree of preference for certain attributes in the constraint-space. In addition to the application of a popular nondominated sorting technique for identifying Pareto solutions, the performance of IMPESO-PS was strengthened via the inclusion of a dynamic selection of the global best, a novel circular nondominated selection of particles, and a special mutation operation.

3.3.2. Constrained Optimization. Scholars have proposed several solutions to constrained optimization problem. For example, Daneshyari and Yen [163] proposed a cultural-based constrained PSO to incorporate the information of the objective function and constraint violation into four sections of the belief space, specifically normative knowledge, spatial knowledge, situational knowledge, and temporal knowledge. The archived information facilitated communication among swarms in the population space and assisted in selecting the leading particles in three different levels: personal, swarm, and global levels. Afshar [164] presented three constrained versions of PSO algorithm for the efficient optimal operation of multireservoir systems using storage/release volumes as decision variables of the problem. Proposed algorithms were based on identifying and excluding the infeasible region of the search space before and during the search. Koulinas et al. [165] proposed a PSO-based hyperheuristic algorithm for solving the resource constrained project scheduling problem. The hyperheuristic worked as an upper-level algorithm that controlled several low-level heuristics which operated to the solution space. The solution representation was based on random keys. Active schedules were constructed by the serial scheduling generation scheme using the priorities of the activities which were modified by the low-level heuristics of the algorithm. Shan and Ren [166] combined PSO with direct approach and applied the method to low-thrust trajectory optimization problems. A double-loop trajectory optimization algorithm was developed. The outer loop of this algorithm was a modified PSO optimizer, which dealt with constrained optimization problems and avoided premature convergence. The direct approach (fourth-order Runge-Kutta shooting/parallel shooting method) was adopted as the inner loop algorithm, whose main task was to correct the particles provided by the outer loop and ensure that all the constraints were satisfied. Yeh and Chien [167] proposed an efficient algorithm for solving a class of constrained minimal spanning tree (MST) problems. The proposed PSO-like strategy for solving constrained MST problems identified optimal MSTs under

degree and delay constraints. Singh et al. [168] presented a novel approach to detect a salient object which involved two phases. In the first phase, three features such as multi-scale contrast, center-surround histogram, and color spatial distribution were obtained. Constrained PSO was used in the second phase to determine an optimal weight vector to combine these features to obtain saliency map to distinguish a salient object from the image background. Paliwal et al. [169] presented a systematic approach for determination of optimal mix of resources. The considered constituent resources comprised of diesel, photovoltaic (PV), wind, and battery storage. A technosocioeconomic criterion was formulated in order to determine optimum combination of resources. PSO was used to determine optimal component sizing for each of the configuration. Cui et al. [170] developed a multitarget PSO (mPSO) to solve the parallel model of independent-component-analysis constrained by a 5-parameter Reference Curve. Shou et al. [171] proposed a hybrid PSO procedure to solve the preemptive resource constrained project scheduling problem in which a maximum of one interruption per activity was allowed. Four types of particle representations were designed and two schedule generation schemes were adopted to decode the particle representations. Particle-updating mechanisms based on the peak crossover operator were designed for all particle representations.

3.3.3. Discrete Optimization. Discrete PSO (DPSO) was proposed and harnessed to address discrete optimization/integer programming problems. Chen and Ludwig [172] devised a PSO-based discrete implementation with a local search strategy (DPSO-LS). The local search strategy helped to overcome local optima in order to improve the solution quality. The DPSO-LS used the Pittsburgh approach whereby a rule base was used to represent a particle. This rule base evolved over time as to finding the best possible classification model. Shen et al. [173] proposed a novel bivelocity DPSO (BVDPSO) approach and extended its application to the nondeterministic polynomial complete multicast routing problem (MRP). First, a novel bivelocity strategy was developed to represent the possibilities of each dimension being 1 and 0. Second, BVDPSO updated the velocity and position according to the learning mechanism of the original PSO in the continuous domain. Experiments showed BVDPSO outperformed not only several state-of-the-art and recent heuristic algorithms for the MRP problems, but also algorithms based on GA, ACO, and PSO. Chen et al. [174] proposed a revised DPSO (RDPSO) to solve the permutation flow-shop scheduling problem with the objective of minimizing makespan (PFSP-makespan). RDPSO proposed new particle swarm learning strategies to thoroughly study how to properly apply the *gbest* solution and the *pbest* solution to guide the search of RDPSO. A new filtered local search was developed to filter the solution regions that had been reviewed and guided the search to new solution regions in order to keep the search from premature convergence. Cai et al. [175] suggested using a novel DPSO for identifying community structures in signed networks. Particles' status had been redesigned in discrete form so as to make PSO proper for discrete

scenarios. Particles' updating rules had been reformulated by making use of the topology of the signed network. Kashan et al. [176] presented a novel DPSO, which used group-based operators, in place of arithmetic operators, in the body of the updating equations analogous to those of the classical PSO equations. All operators in the new algorithm worked with constructed cells (groups) rather than parts/machines (objects). Xu et al. [177] employed a DPSO approach to solve the requirements contradiction between high transparency in pass band and high reflectance in stop band. Garg and Singh [178] used ε -fuzzy dominance sort based DPSO (-FDPSO) approach to solve the workflow scheduling problem in the grid. The metric, fuzzy dominance which quantified the relative fitness of solutions in multiobjective domain was used to generate the Pareto optimal solutions. In addition, the scheme also incorporated a fuzzy-based mechanism to determine the best compromised solution. Zong et al. [179] proposed a DPSO with neighborhood learning factor algorithm to solve the temporal-spatial conflict and congestion for pedestrian-vehicle mixed evacuation. The proposed algorithm introduced a neighborhood learning factor to simulate the subgroup phenomenon among evacuees and to accelerate the evacuation process. Ezzeldin et al. [180] used integer DPSO as an optimization technique for the design of water distribution networks in order to minimize its total cost. Because the particle swarm was highly sensitive to its parameters and boundary conditions, the available restricted boundary conditions were applied. Also, a new boundary condition called the billiard boundary condition was introduced, which did not depend on the velocity clamping that mainly depended on human assumptions.

3.3.4. Binary Integer Programming. 0-1 integer programming or binary integer programming (BIP) is the special case of integer programming where variables are required to be 0 or 1. Binary PSO (BPSO) was used to solve this type of problems. Zhai and He [181] proposed a new immune BPSO (IBPSO) to solve the problem of instance selection for time series classification, whose objective was to find out the smallest instance combination with maximal classification accuracy. The proposed IBPSO was based on the BPSO algorithm. Its immune mechanism included vaccination and immune selection. Sarath and Ravi [182] developed a BPSO based association rule miner, which generated the association rules from the transactional database by formulating a combinatorial global optimization problem, without specifying the minimum support and minimum confidence unlike the a priori algorithm. The quality of the rule was measured by a fitness function defined as the product of support and confidence. Taha and Abu Al Nadi [183] presented a maximum likelihood estimate (MLE) to detect the number of vacant channels in the spectrum. The resulting MLE needed exhaustive search to be determined accurately. BPSO was proposed to solve the problem. Simulation results had shown significant improvement of the MLE-BPSO estimated over the conventional techniques by 3–5 dB. El-Maleh et al. [184] proposed an improved BPSO algorithm and demonstrated its effectiveness in solving the state assignment problem

in sequential circuit synthesis targeting area optimization. Experimental results demonstrated the effectiveness of the proposed BPSO algorithm in comparison to other BPSO variants and in comparison to GA, SimE, and deterministic algorithms like Jedi and Nova. Erturk et al. [185] proposed a novel spatial resolution enhancement method using fully constrained least squares (FCLS) spectral unmixing and spatial regularization based on modified BPSO to achieve spatial resolution enhancement in hyperspectral images, without using an additional image with higher spatial resolution. The proposed method had a highly parallel nature with respect to its counterparts in the literature and was fit to be adapted to field-programmable gate array architecture. Zhang et al. [186] proposed a novel spam detection method that focused on reducing the false positive error of mislabeling nonspam as spam. BPSO with mutation operator (MBPSO) was employed as the subset search strategy. The results showed that the MBPSO performed better than sequential forward selection (SFS) and sequential backward selection (SBS). Yin et al. [187] used a feature selection algorithm according to the separability criterion to preselect the discrete cosine transform (DCT) coefficients and then employed search algorithm based on BPSO and SVM to find an optimal combination of the DCT coefficients. Yang et al. [188] proposed a modified version of BPSO (MBPSO), which adopted a different transfer function and a new position updating procedure with mutation, for the task allocation problem to obtain the best solution. Each particle in MBPSO was encoded to represent a complete potential solution for task allocation. The task workload and connectivity were ensured by taking them as constraints for the problem. Multiple metrics, including task execution time, energy consumption, and network lifetime, were considered a whole by designing a hybrid fitness function to achieve the best overall performance. Ganesh et al. [189] proposed entropic BPSO (EBPSO), which generated an entropy map, the highest value of which was used to localize the ear in a face image. Experimental results showed the promising performance of EBPSO for ear detection on four benchmark face databases: CMU PIE, Pointing Head Pose, Color FERET, and UMIST.

3.4. Theoretical Analysis

3.4.1. Parameter Choice. The choice of parameters of PSO has a large impact on its optimization performance [190]. Therefore, how to select or tune the parameters yielding good results had been the hot topic. Kumar and Chaturvedi [191] developed a FPSO, in which inertia weight was adaptively adjusted using fuzzy logic controller (FLC) during the search process. The FLC presented had one input and one output into PSO. Zhang et al. [192] proposed a simple way to estimate the nonnegative real parameter tuple (ω , c_1 , and c_2) of standard PSO algorithm using control theory. The distribution of complex characteristic roots on the convergence region of particles was studied by means of linear discrete-time system analysis method. Yang [193] proposed an improved PSO variant, called PSO with modified velocity strategy (MVPSO), in which each particle was attracted

by the global best particle and a random particle chosen from a set of good particles. Simulation results showed that MVPSO obtained better performance than standard PSO and two other improved PSO variants. Sun et al. [194] presented a comprehensive analysis of the QPSO algorithm. They analyzed the behavior of a single particle in QPSO in terms of probability measure. Since the particle's behavior was influenced by the contraction-expansion (CE) coefficient, the goal of the theoretical analysis was to find out the upper bound of the CE coefficient, within which the value of the CE coefficient selected could guarantee the convergence or boundedness of the particle's position. Yassin et al. [195] found that BPSO algorithm was subject to several parameters: swarm size, maximum iterations, and initial positions. They investigated the effect of the swarm size parameter on the convergence of BPSO. The results over DC motor dataset indicated that the optimal swarm size for convergence was between 20 and 30 particles. Wang et al. [196] compared and analyzed the optimization performance of PSO under different parameters, in order to guarantee the convergence of PSO applied to the inverting of ellipsometry. The result showed that the range of inertia weight ω from 0.5 to 0.8, the sum of learning parameters c_1 and c_2 preferably no more than 3, and a smaller c_1 and a bigger c_2 ensured the better optimization performance of PSO. Hao et al. [197] proposed molecular force model based PSO (MFMP SO). Two parameters were introduced in the MFMP SO algorithm. The orthogonal test design method was applied to optimize the parameter combinations of three levels and four factors, which included $d(1)$ and $d(h)$, the population size, and the iteration number. Xu [198] proposed an adaptive parameter tuning of PSO based on velocity information (APSO-VI). This algorithm introduced the velocity information defined as the average absolute value of velocity of all the particles. A new strategy presented that the inertia weight was dynamically adjusted according to average absolute value of velocity, which followed a given nonlinear ideal velocity by feedback control. Under the guide of the nonlinear ideal velocity, APSO-VI maintained appropriate swarm diversity and alleviated the premature convergence validly. Chauhan et al. [199] proposed three new nonlinear strategies for selecting inertia weight which played a significant role in particle's foraging behavior. The PSO variants implying these strategies were named fine grained inertia weight PSO (FGIWPSO), double exponential self-adaptive IWPSO (DESIWPSO), and double exponential dynamic IWPSO (DEDIWPSO). In FGIWPSO, inertia weight was obtained adaptively, depending on particle's iteration wise performance, and decreased exponentially. DESIWPSO and DEDIWPSO employed Gompertz function, a double exponential function for selecting inertia weight. In DESIWPSO the particles' iteration wise performance was fed as input to the Gompertz function. On the other hand, DEDIWPSO evaluated the inertia weight for whole swarm iteratively using Gompertz function where relative iteration was fed as input. Zhang et al. [200] found that engineering experience could be used to determine the parameters of an optimization algorithm. They analyzed the dynamic characteristics of PSO through a large number of experiments and constructed a relationship between

the dynamic process of PSO and the transition process of a control system. A novel parameter strategy for PSO was proven using the overshoot and the peak time of a transition process. This strategy provided not only a series of flexible parameters for PSO but also a new way to analyze particle trajectories that incorporated engineering practices. The experimental results showed that the proposed strategy was effective and easy to implement. Kanemasa and Aiyoshi [201] combined a feedback element as an algorithm tuner with an original algorithm; the resulting algorithm was applied to the optimization problem in question. They used genetic programming (GP) to generate tuning rules to automatically tune the PSO algorithm, namely, augmented PSO by using GP as a meta-algorithm to solve the learning problem of automatically generating tuning rules for the parameters in the PSO algorithm.

3.4.2. Convergence Analysis. PSO may converge to global optimal or local optimal positions. The latter is unexpected for PSO-users. Attempts at mathematically analyzing PSO convergence exist in literatures. These analyses offer guides for selection parameters of PSO in another way that guarantees global convergence and avoid premature. Wang and Shen [202] gave the general mathematical description of PSO. Afterwards, they proved that the solution space of PSO in normed space and the iterative relations of PSO were contraction mapping. Using the theorem of Banach space and contraction mapping principle, they proved the existence and uniqueness of the convergence position. Sun et al. [203] presented a convergence analysis and performance evaluation of QPSO algorithm. They investigated in detail the convergence of the QPSO algorithm on a probabilistic metric space and proved that the QPSO algorithm was a form of contraction mapping and converged to the global optimum. It was the first time that the theory of probabilistic metric spaces had been employed to analyze a stochastic optimization algorithm. They provided a new definition for the convergence rate of a stochastic algorithm as well as definitions for three types of convergence according to the correlations between the convergence rate and the objective function values. Kurihara and Jin'no [204] analyzed the convergence property of the PSO and its application to the nonlinear blind source separation system. The interparticle communication of the PSO was realized by the past history of the neighbors and depended on the network structure of the swarm. Lin [205] considered the router node placement of wireless mesh networks in a dynamic network scenario. They first modelled a mathematical form for the concerned problem, then proposed a PSO approach, and, from a theoretical aspect, provided the convergence and stability analysis of the PSO with constriction coefficient, which was much simpler than the previous analysis. Zhang et al. [206] studied an improved BBPSO algorithm with adaptive disturbance (ABPSO). They used stochastic process theory to analyze the convergence of ABPSO, by regarding each particle's position as a stochastic vector. Lin et al. [207] thought the PSO had a tendency to get stuck in a near-optimal solution especially for middle and large size problems, and it was difficult to

improve solution accuracy by fine-tuning parameters. Hence, they proposed a local and global search combined PSO (LGSCPSOA), analyzed its convergence, and obtained its convergence qualification. Kim and Li [208] suggested the conventional derivative-based estimation approach was often terminated earlier without converging due to the singularity if a model was statistically nonidentifiable. To circumvent this difficulty, a derivative-free global optimization algorithm was developed by combining PSO with a derivative-free local optimization algorithm to improve the rate of convergence of PSO. They further checked the convergence of the proposed method.

3.5. Parallel Implementation. Parallel computing is a computational form, in which computations are carried out simultaneously. On one hand, computers using multicore, multiprocessor, and graphics processing unit (GPU) contain multiple processing elements within a single machine. On the other hand, clusters, grids, and clouds employed multiple computers to work on the same task.

3.5.1. Multicore. PSO can be implemented on multicore (multiprocessor) conditions. Waintraub et al. [209] used parallel computation to overcome the huge computational costs required by PSO. They investigated the master-slave approaches and developed several different PPSO algorithms exploring the advantages of enhanced neighborhood topologies implemented by communication strategies in multiprocessor architectures. Yu [210] proposed the incorporation of a local search heuristic into the basic PSO algorithm. The new, hybrid metaheuristic was called twin PSO (TPSO). The proposed metaheuristic scheme was applied to a flow shop with multiprocessors scheduling problem.

3.5.2. GPU Computing. GPU is a specialized electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display. Modern GPUs are very efficient at manipulating computation of PSO. Hung and Wang [211] focused on the acceleration of PSO for solving box-constrained, load-balanced optimization problems by parallelization on a GPU. They proposed a GPU-accelerated PSO (GPSO) algorithm by using a thread pool model and implement GPSO on a GPU. Numerical results showed that the GPU architecture fitted the PSO framework well by reducing computational timing, achieving high parallel efficiency, and finding better optimal solutions by using a large number of particles. Rymut et al. [212] discussed how to combine particle filter (PF) with PSO to achieve better object tracking. They also proposed a parallel resampling scheme for particle filtering running on GPU. They showed the efficiency of the parallel PF-PSO algorithm on 3D model based human motion tracking. Kumar et al. [213] thought the applications requiring massive computations got benefit from the GPU with compute unified device architecture (CUDA) by reducing the execution time. Hence, they presented a detailed study of parallel implementation of the cooperative PSO (CPSO). They also presented a comparative study on

CPSO implemented in C and C-CUDA. Awwad et al. [214] applied GPU computing solution based on CUDA to solve the topology control problem in hybrid radio frequency and free space optics wireless mesh networks by adapting and adjusting the transmission power and the beam-width of individual nodes according to QoS requirements. Their approach was based on a PSO and was implemented on the NVIDIA GeForce GTX 285 GPU. The implementation achieved a performance speedup factor of 392 over a CPU-only implementation. Chen et al. [215] proposed a PSO-based algorithm to efficiently find optimal uniform designs with respect to the central composite discrepancy (CCD) criterion. Parallel computation techniques based on state-of-the-art GPU were employed to accelerate the computations. They further demonstrated that the proposed algorithm could be extended to incorporate desirable space-filling properties, such as the noncollapsing property.

3.5.3. Cloud Computing. Cloud computing is a computing form in which large groups of remote servers are networked to allow centralized data storage and online access to computer services or resources. Scholars studied the performance of PSO implemented in cloud computing. Liu et al. [216] established a job scheduling model based on the PSO algorithm for cloud computing to reduce the energy consumption and improve the profit. Based on open source cloud computing simulation platform CloudSim, compared to GA and random scheduling algorithms, the results showed that the proposed algorithm obtained a better solution concerning the energy cost and profit. Xu and You [217] minimized the thermal residual stresses (TRS) of the unidirectional ceramic matrix composites (CMC) with multilayered interphases by controlling the interphases thicknesses. The MapReduce was extended to a new iterative MapReduce, which was combined with classical PSO algorithm to develop an iterative MapReduce guided PSO (IMPSO) algorithm. The IMPSO algorithm was interfaced with finite element code to find an optimal design for minimizing the TRS within CMCs. Ramezani et al. [218] developed a comprehensive multiobjective model for optimizing task scheduling to minimize task execution time, task transferring time, and task execution cost. They designed a multiobjective algorithm based on MOPSO method to provide an optimal solution for the proposed model. To implement and evaluate the proposed model, they extended Jswarm package to multiobjective Jswarm (MO-Jswarm) package. They also extended CloudSim toolkit applying MO-Jswarm as its task scheduling algorithm. Govindarajan et al. [219] captured the data from students and analyzed and clustered the data based on their individual performances in terms of accuracy, efficiency, and quality. The clustering process was carried out by employing PSO. The proposed PSO-based clustering was compared with existing K -means algorithm for analyzing the performance of intercluster and intracluster distances. Finally, the processed data was effectively stored in the Cloud resources using Hadoop distributed file system (HDFS). Ramezani et al. [220] proposed a task-based system load balancing method using PSO (TBSLB-PSO) that achieved system load balancing by only transferring extra tasks from an overloaded VM instead of migrating

the entire overloaded VM. They designed an optimization model to migrate these extra tasks to the new host VMs by applying PSO. To evaluate, they extended the cloud simulator (CloudSim) package and used PSO as its task scheduling model.

4. Applications of PSO

PSO has been used in many applications in various academic and industrial fields so far. Using the analytical tool provided by “Web of Science Core Collection,” the hottest application categories are “electrical and electronic engineering,” “automation control systems,” “communication theory,” “operations research,” “mechanical engineering,” “fuel and energy,” “medicine,” “chemistry,” “biology,” and so forth.

4.1. Electrical and Electronic Engineering. PSO was used by researchers to solve the optimization problems encountered in electrical and electronic engineering. Ganguly et al. [221] presented a multiobjective planning approach for electrical distribution systems under uncertainty in load demand incorporating DG. The optimization tool was a MOPSO variant that used heuristic selection and assignment of leaders or guides for efficient identification of nondominated solutions. Komsiyah [222] used Gaussian PSO and Lagrange multiplier to solve the EDP of electric power generation, scheduling the committed generating units outputs so as to meet the required load demand at minimum operating cost, while satisfying all units and system equality and inequality constraint. Feng et al. [223] employed orthogonal signal correction and PSO in order to detect wound infection by and improve the performance of electronic nose. Pekşen et al. [224] proposed a PSO method for estimating the model parameters of a layered anisotropic earth model such as horizontal and vertical resistivities and thickness. The result was promising and the proposed method could be used for evaluating one-dimensional direct current data in anisotropic media. Yang et al. [225] proposed a novel electric vehicle (EV) charging model. A PSO algorithm was proposed for the model optimization. Simulation results showed that the proposed strategy could reduce the operational cost of the power grid considerably, while meeting the EV owner’s driving requirement. de Mendonça [226] used PSO and heuristic information to find the minimum expansion cost of the electrical energy transmission system. The proposed methodology was applied to the Garver system and to two real equivalent systems for the south and southeast of Brazil. Liu et al. [227] employed DPSO to systematically investigate the structural stability and features of tetrahedral Pt-based bimetallic nanoparticles with high-index facets. Aich and Banerjee [228] applied SVM for developing the model of electrical discharge machining process, with the aim of predicting the output with reasonable accuracy. PSO was employed for the purpose of optimizing SVM parameter combinations. Chou et al. [229] surveyed defects of reservoir grounding system, considering the performance of lightning protection and improved design, based on soil drilling data and the PSO technique. They showed PSO was indeed

better than conventional planning and design method. Lee et al. [230] applied an electrical resistance tomography technique combining the PSO algorithm with the Gauss-Newton method, to the visualization of two-phase flows. Thakral and Bakhshi [231] employed PSO with negative factor counting technique and inverse iteration method, for designing novel binary and ternary copolymers based on thiophene, pyrrole, and furan skeletons. Fister et al. [232] tried to use the semantic tools such as resource definition framework (RDF) and RDF query language (SPARQL) for the optimization purpose. These tools were combined with PSO and the selection of the best solutions depends on its fitness. Aghaei et al. [233] presented a multiobjective optimization algorithm for the multistage distribution expansion planning (MDEP) in the presence of DGs using nonlinear formulations. The objective functions of the MDEP consisted of minimization of costs, energy-not-distributed, active power losses, and voltage stability index based on short circuit capacity. A modified PSO algorithm was developed and used for this multiobjective MDEP optimization. Selakov et al. [234] proposed a practical new hybrid model for short-term electrical load forecasting based on PSO and SVM. Proposed PSO-SVM model was targeted for forecast load during periods with significant temperature variations. The proposed model detected periods when temperature significantly changed based on weather (temperature) forecast and decided whether the model could be trained just on recent history. Shirvany et al. [235] used a modified PSO to solve the electroencephalography (EEG) source localization problem. The method was applied to non-invasive EEG recording of somatosensory evoked potentials for a healthy subject. Tungadio et al. [236] discussed the application of PSO to solve the state estimation problem, that is, to estimate the state variables of the power system by minimizing all measurement errors available at the control center. PSO was shown to be more effective than GA.

4.2. Automatic Control System. PSO has found several applications in automatic control systems. Cai and Yang [237] proposed an improved PSO-based approach for a team of mobile robots to cooperatively search for targets in complex unknown environments. The improved cooperation rules for a multirobot system were applied in the potential field function, which acted as the fitness function of the PSO. Kolomvatsos and Hadjieftymiades [238] held a negotiation between intelligence agents, which undertook the responsibility of representing buyers and sellers and negotiated over the conclusion of purchases. They studied concurrent negotiations between a buyer and a set of sellers. The PSO algorithm was adopted by each thread, which followed a specific strategy and adopted swarm intelligence techniques for achieving the optimal agreement. Pandey et al. [239] presented the load frequency control (LFC) problem using different optimization algorithms for two types of power system configurations. The control scheme proposed was based on linear matrix inequalities with its parameters tuned by PSO. Štimac et al. [240] applied PSO to tune PID controller, using active magnetic bearing that suspended the rotating shaft and maintained it in levitated position by

applying controlled electromagnetic forces on the rotor in radial and axial directions. Nedic et al. [241] proposed a new cascade load force control design for a parallel robot platform. A parameter search based on PSO was suggested to effectively search the parameters of the cascade controller. Simulation results showed the advantages of the proposed optimal tuned cascade controller to solve the formulated tracking problem in relation to the classical proportional-integral controller. Chang and Chen [242] aimed at the PID control system design for multivariable input and multivariable output (MIMO) processes. An improved version of PSO algorithm was utilized to design PID control gains in MIMO control systems. The velocity updating formula of the developed algorithm included a new factor, that is, the best particle of each subpopulation, to enhance the search capacity. Xiang et al. [243] proposed a new sensorless control scheme without the use of a position feedback sensor. With the help of the system identification technique and PSO, the control scheme showed a satisfactory command tracking performance for the conducting polymer actuator's step and dynamic displacement responses, especially under a disturbance, without needing a physical feedback loop, but using a simulated feedback loop. Danapalasingam [244] presented an autonomous helicopter stabilization in the presence of a wind disturbance. A controller was designed based on the nonlinear adaptive output regulation and robust stabilization of a chain of integrators by a saturated feedback to stabilize the position and velocity of a helicopter. PSO was applied to automate the tuning process. Mahmoodabadi et al. [245] presented a novel MOPSO called Ingenious-MOPSO and compared its capability with three well-known multiobjective optimization algorithms. They applied the proposed method to an intellectual challenge in robotics, that is, a biped robot walking in the lateral plane on slope. Zhong et al. [246] presented an improved PID intelligent control algorithm, which was applied to the electric gas pressure regulator. The algorithm used the improved radial basis function (RBF) neural network based on PSO algorithm to make online adjustment on PID parameters. Both theoretical analysis and simulation result showed that the algorithm shortened the step response time and improved tracking performance. Perng et al. [247] proposed a combination of PSO and RBFNN to determine the optimal operating point for the PID controller of a wind turbine and to identify the stability regions in the parameter space. Huang and Li [248] utilized an improved PSO with bounded constraints technique on velocity and positioning for adjusting the gains of a PID and iterative learning control (ILC) controllers. Simulations and experiment results showed that the proposed controller could reduce the error significantly after two learning iterations. The developed method using bounded constraints technique provided valuable programming tools to practicing engineers. Nisha and Pillai [249] demonstrated control accuracy and computational efficiency of nonlinear model predictive control (NMPC) strategy, which utilized a probabilistic sparse kernel learning technique called relevance vector regression (RVR) and PSO with controllable random exploration velocity (PSO-CREV).

4.3. Communication Theory. An interesting application area of PSO is communication theory. Yousefi et al. [250] presented the design procedure and implementation of a dual band planar quadrature hybrid with enhanced bandwidth. Small dual frequency transformers in two sections were used to modify conventional broad band hybrid coupler for having arbitrary dual band operation frequencies. By merging compilers and full-wave simulators, PSO was used to achieve proper values to have desired goals. Sun et al. [251] applied PSO algorithm to design ultrawideband (UWB) pulse waveform, which was converted into a constraint problem with multiband chirp signals. Yongqiang et al. [252] investigated the receive antenna selection problem to maximize capacity in wireless MIMO communication system, which could be formulated as an integer programming optimization problem and could not be directly solved because of its nonconvex characteristics caused by the discrete binary antenna selection factor. Hence, they introduced PSO, in which the particle was defined as the discrete binary antenna selection factor and the objective function was associated with the capacity corresponding to the specified antenna subsection represented by the particle. Chiu et al. [253] presented a method for determining the required number and locations of transmitting antennas (TXs) to optimize wireless propagation coverage in an indoor UWB communication system. They used the 3D ray-tracing technique associated with a PSO and Asynchronous PSO (APSO) for optimizing the TXs location in an indoor environment. Minasian and Bird [254] applied PSO as a design tool for a parasitically coupled microstrip antenna array. The antenna was characterized by a unique nonintuitive design which resulted from an application of PSO with no constraints implemented on the shape of the array during optimization apart from the maximum dimensions. Zubair and Moinuddin [255] used PSO for joint optimization of three different shape-printed monopole antennas, namely, printed square monopole antenna, printed circular monopole antenna, and printed hexagonal monopole antenna, for UWB applications. Kim and Lee [256] applied metaheuristic to solve the optimum scheduling of multiple channels and time slots in multihop networks. SA and PSO were adopted to schedule the resources. The simulation results demonstrated that PSO-based scheduling outperformed SA-based scheduling in terms of end-to-end delay. Yazgan and Hakki Cavdar [257] comparatively studied PSO and least mean square (LMS) algorithms to estimate the optical communication channel parameters for radio over fiber systems. PSO reached better mean square error values, when selecting high noisy fiber optical channels or free space optical channels. Rabady and Ababneh [258] employed PSO to reach a global optimal design for optical interference filters, which were widely used in modern optical communication systems, biomedical applications, astronomy, and many others. Das et al. [259] applied ANN training with PSO for the problem of channel equalization. Existing applications of PSO to ANN training had only been used to find optimal weights of the network. Novelty in this paper was that it also took care of appropriate network topology and transfer functions of the neuron. The PSO algorithm optimized all the variables, including

network weights and network parameters. Scott-Hayward and Garcia-Palacios [260] introduced a resource allocation solution capable of handling mixed media applications within the constraints of a 60 GHz wireless network. A new channel time allocation PSO (CTA-PSO) was proposed to solve the network utility maximization resource allocation problem. CTA-PSO optimized the time allocated to each device in the network in order to maximize the quality of service experienced by each user. Omidvar and Mohammadi [261] used PSO in intelligent choosing of number of message copies. Regarding message delivery ratio and network overhead, PSO greatly helped in finding the suitable number of copies. Kuila and Jana [262] presented linear/nonlinear programming formulations of energy efficient clustering and routing problems for wireless sensor networks. They proposed two PSO-based algorithms to solve the optimization problem.

4.4. Operations Research. PSOs have been employed to solve problems in operations research. Liu and Wang [263] proposed a new task scheduling model. In the model, the task execution time was optimized in view of both the task running time and the system resource utilization. They improved the standard PSO and introduced a simple mutation mechanism and a self-adapting inertia weight method by classifying the fitness values. Che et al. [264] proposed a novel method based on the analytic network process and turbo PSO to evaluate partners and to determine an optimal supply chain network pattern and production-distribution mode. Hajipour and Pasandideh [265] focused on determination of the number of required facilities along with the relevant allocation process. They first proposed a novel biobjective facility location problem within batch arrival queuing framework under capacity, budget, and nearest-facility constraint. Then, PSO algorithm with considering a specific representation process was proposed to solve the biobjective problem. Cabrerizo et al. [266] used PSO to solve the granulation of the linguistic terms problem. A performance index was maximized by a suitable mapping of the linguistic terms on information granules. Prescilla and Selvakumar [267] applied a modified BPSO algorithm and a novel BPSO algorithm to solve the real-time task assignment in heterogeneous multiprocessor. The problem consisted of a set of independent periodic tasks, which had to be assigned to a heterogeneous multiprocessor without exceeding the utilization bound. The objective was to schedule maximum number of tasks with minimum energy consumption. Duan et al. [268] developed a multiple UCAVs cooperative air combat simulation platform, which was based on PSO, ACO, and game theory. The Matlab program was used as the developing tool. The practitioners could investigate the inherent mechanism by applying game theory to solve the mission decision-making problem of multiple UCAVs in attacking multiple objects. Belmecheri et al. [269] proposed a PSO with a local search to solve the vehicle routing problem (VRP) with heterogeneous fleet, mixed backhauls, and time windows. Hu et al. [270] proposed a hybrid chaos PSO (HPSO) to solve VRP with time window. The chaos algorithm was employed to reinitialize the particle swarm.

An efficient insertion heuristic algorithm was also proposed to build the valid vehicle route in the particle decoding process. A premature convergence judgment mechanism was formulated and combined with the chaos algorithm and Gaussian mutation into HPSO when the particle swarm fell into the local convergence. Al Badawi and Shatnawi [271] developed an efficient method based on PSO to solve the multiprocessor task scheduling problem, which determined the assignment of tasks to the processors in a multiprocessor environment. Goksal et al. [272] presented a heuristic solution approach based on PSO, in which a local search was performed by variable neighborhood descent algorithm to solve the VRP with simultaneous pickup and delivery. Zhang et al. [273] first formulated and described the energy-aware real-time task scheduling problem in heterogeneous multiprocessors. Then they proposed a PSO-based algorithm, which could successfully reduce the energy cost and the time for searching feasible solutions. Kechagiopoulos and Beligiannis [274] designed a PSO-based algorithm, aiming at the efficient solution of urban transit routing problem, which comprised an NP-hard problem that dealt with the construction of route networks for public transit networks.

4.5. Mechanical Engineering. A few applications of PSO are related to mechanical engineering. Ming et al. [275] investigated the implicit relationship between the compositions and mechanical properties of as-cast Mg-Li-Al alloys. A momentum back-propagation (BP) neural network with a single hidden layer was established. PSO was applied to optimize the BP model. Chen et al. [276] aimed to develop a method for reliability-based optimum design of composite structures. A procedure combining PSO and finite element analysis (FEA) was proposed. Examples showed that the proposed method had good stability and was efficient in dealing with the probabilistic optimal design of composite structures. Mohan et al. [277] evaluated the use of frequency response function (FRF) with the help of PSO technique, for structural damage detection and quantification. It was observed that the use of FRF as response of damaged structure had led to better accuracy, since it contained data related to mode shape in addition to natural frequencies. Chen et al. [278] applied a surrogate based PSO algorithm, which combined the surrogate modeling technique and PSO, to the reliability-based robust design (RBRD) of composite pressure vessels. An optimization problem for maximizing the performance factor was formulated by choosing the winding orientation of the helical plies in the cylindrical portion, the thickness of metal liner, and the drop-off region size as the design variables. Strength constraints for composite layers and the metal liner were constructed by using Tsai-Wu failure criterion and Mises failure criterion, respectively. Zhang et al. [279] presented a methodology for the identification of parameter values in the Barcelona basic model (BBM) by inverse analysis of the experimental cavity pressure-cavity strain curve from pressure-meter tests in unsaturated soils. A novel parallel PSO algorithm was utilized to minimize the difference between measured and computed values on the cavity pressure-cavity strain curve. Wang et al. [280]

proposed a decomposition-based multiobjective differential evolution PSO (DMDEPSO) algorithm for the design of a tubular permanent magnet linear synchronous motor (TPMLSM), which took into account multiple conflicting objectives. DMDEPSO hybridized DE and PSO together, first decomposed the multiobjective optimization problem into a number of single-objective optimization subproblems, each of which was associated with a Pareto optimal solution, and then optimized these subproblems simultaneously. Lazrag et al. [281] aimed to identify all the hydraulic parameters of sand by using a unique column test, which was a gravity drainage test where only the flow-rate temporal evolution was measured. They used several tensiometers installed in different positions along the column to measure the pressure head inside the soil sample. Inverse analysis was performed thanks to the PSO algorithm and the finite element modeling of the column test. Lee et al. [282] first developed 3D numerical models of a fractured femur with the locking compression plates (LCP). Then, the best screw position and number of LCPs were determined by using a simulation-based PSO algorithm, in order to discover the best number and positions of LCP screws to achieve acceptable fixation stability. Lake et al. [283] used PSO to optimize the geometry of a slotted MEMS resonator, in order to reduce energy dissipation from thermoelastic dissipation (TED). The optimization technique combined fundamental physics with PSO to navigate the complicated design space that arises from multiphysical problems. Vosoughi and Gerist [284] proposed a hybrid optimization technique that combines FEA, continuous GA, and PSO, for damage detection of laminated composite beams. Ribeiro et al. [285] investigated the geometric effects (thickness, width, and internal cell angle) of auxetic structures made of recycled rubber composites based on experimental and numerical data. The response surface models integrated with the PSO and FEA were proposed in order to obtain a range of solutions that provided useful information to the user during the selection of geometric parameters for reentrant cells. Kitak et al. [286] presented a method for determination of heat transfer coefficients for finite-element method housing model of medium-voltage switchgear cell. Suggested method was based on the PSO algorithm. A real test model of partition wall had been created as well as an equivalent numerical finite-element model. Kalatehjari et al. [287] applied PSO in 3D slope stability problem to determine the critical slip surface of soil slopes. In the contrast, past publications were limited to 2D slope stability analysis.

4.6. Civil Engineering. PSO was widely used to solve the optimization problems existing in civil engineering. Ashuri and Tavakolan [288] presented a hybrid GA-PSO approach to solve complex time-cost-resource optimization problem in construction project planning. The proposed method also used the fuzzy set theory to characterize uncertainty about the input data (i.e., time, cost, and resources required to perform an activity). Experiments showed that the proposed fuzzy enabled hybrid GA-PSO approach was superior to existing optimization algorithms at finding better project

schedule solutions with less total project cost, less total project duration, and less total variation in resource allocation. Bozorgi-Amiri et al. [289] investigated a relief chain design problem where not only demands but also supplies and the cost of procurement and transportation were considered as the uncertain parameters. An efficient solution approach based on PSO was developed in order to solve the proposed mathematical model. Sadoghi Yazdi et al. [290] used a neurofuzzy model in conjunction with PSO for calibration of soil parameters used within a linear elastic-hardening plastic constitutive model with the Drucker-Prager yield criterion. The neurofuzzy system was used to provide a nonlinear regression between the deviatoric stress and axial strain obtained from a consolidated drained triaxial test on samples of poorly graded sand. The soil model parameters were determined in an iterative optimization loop with PSO and an adaptive network based on a fuzzy inference system such that the equations of the linear elastic model and (where appropriate) the hardening Drucker-Prager yielded criterion are simultaneously satisfied. Bolat et al. [291] developed a particle swarm optimization (PSO) algorithm to deal with car-call allocation problem. In vertical transportation, when a passenger makes a hall call by pressing a landing call button installed at the floor and located near the cars of the elevator group, the complex-elevator-group-control system must allocate one of the cars of the group to the hall call. Babu and Vijayalakshmi [292] presented a hybrid model PSO-GA, aimed at effectively utilizing local and global search capabilities of PSO and GA, respectively, to reduce the computational burden. Their analyses on different water distribution networks uncovered that the proposed hybrid model was capable of exploring the optimal combination of pipe diameters with minimal computational effort. Mohan et al. [293] showcased the efficacy of PSO and GA in damage assessment of structures, for early detection of cracks, corrosion, and structural failure in aging structures. The results showed the effectiveness of PSO in crack identification and the possibility of implementing it in a real-time structural health monitoring system for aircraft and civil structures. Asadnia et al. [294] presented the application of an improved PSO technique for training an ANN to predict water levels for the Heshui Watershed, China. The results indicated that the LM-NN model performed poorly in predicting the low and peak water levels in comparison to the PSO-based ANNs. Moreover, the IPSONN model was superior to CPSONN in predicting extreme water levels. Lastly, IPSONN had a quicker convergence rate compared to CPSONN. Yan et al. [295] presented a new imaging-based intelligent method for quantitatively rating the corrosion states of weathering steel bridges. To enhance the performance of a SVM in corrosion state classification, a PSO was developed to obtain the optimal parameters of the SVM. Their comparative study indicated that PSO-SVM could achieve better classification accuracy rates than ANN did. Wang and Li [296] advanced PSO and applied it to find effectively near-optimal solutions to the redundancy allocation problem of multistate systems with bridge topology, which was a commonly used structure for load balancing and control in applications such as electric

power generation and transmission, transportation and computer networks, and electronic circuits. Sadeghi et al. [297] used an improved PSO to optimize a hybrid vendor-managed inventory and transportation problem with fuzzy demand. The parameters of both algorithms were calibrated using the Taguchi method to have better quality solutions. Kanović et al. [298] presented the comparison of some well-known global optimization techniques in optimization of an expert system controlling a ship locking process. The purpose was to find the best algorithm for optimization of membership function parameters of fuzzy expert system for the ship-lock control. The results confirmed that all these procedures showed similar results and provided overall improvement of ship-lock operation performance.

4.7. Fuel and Energy. PSO had been introduced for the fuel and energy area. Mandal et al. [299] described the problem of short-term wind power production forecasting based on meteorological information. Aggregated wind power forecasts were produced for multiple wind farms using a hybrid intelligent algorithm that used a data filtering technique based on wavelet transform (WT) and a soft computing model based on neural network, which was optimized by using PSO algorithm. Chao [300] presented an adaptive maximum power point (MPP) tracking approach for PV power generation system. An optimization approach was proposed on the basis of a PSO algorithm for the complexity reduction in the determination of weighting values. Chen et al. [301] proposed chaotic improved PSO-based multiobjective optimization (MOCIPSO) and improved PSO-based multiobjective optimization (MOIPSO) approaches, for solving complex multiobjective, mixed integer nonlinear problems such as minimization of power losses and L index in power systems simultaneously. Hu et al. [302] presented a data-driven method for estimating the capacity of Li-ion battery based on the charge voltage and current curves. An adaptation of PSO was proposed to find the optimal combination of feature weights for creating a k NN regression model that minimizes the cross validation (CV) error in the capacity estimation. Tabet et al. [303] used a mathematical model to determine the solar radiation incident on an inclined surface and the optimum slope angles for each month, season, and year are calculated for solar hybrid collectors. PSO method was applied to obtain the tilt angle setting of the tilt angle of photovoltaic thermal (PVT) hybrid collector. The objective was to improve the efficiency of PVT collector. Bahrami et al. [304] thought STLF (short-term electric load forecasting) played an important role in the operation of power systems. A new model based on combination of the WT and grey model (GM) was presented for STLF and was improved by PSO algorithm. Askarzadeh [305] studied the performance of different PSO variants for estimating Iran's electricity demand. Seven PSO variants, namely, original PSO, PSO-w (PSO with weighting factor), PSO-cf (PSO with constriction factor), PSO-rf (PSO with repulsion factor), PSO-vc (PSO with velocity control), CLPSO (comprehensive learning PSO), and a MPSO (modified PSO), were used to find the unknown weighting factors based on the data

from 1982 to 2003. Sharafi and ElMekkawy [306] proposed a novel approach for optimal design of hybrid renewable energy systems (HRES) including various generators and storage devices. The epsilon-constraint method was applied to minimize simultaneously the total cost of the system, unmet load, and fuel emission. A PSO-simulation based approach was used to tackle the multiobjective optimization problem. García-Triviño et al. [307] presented and evaluated three energy management systems (EMSs) based on PSO for long-term operation optimization of a grid-connected hybrid system. It was composed of wind turbine (WT) and PV panels as primary energy sources and hydrogen system (fuel cell, electrolyzer and hydrogen storage tank) and battery as energy storage system (ESS). Biao et al. [308] proposed and established an electricity market based joint optimization scheduling model of thermal power plants, from the perspective of economics and environmental science. Mutation PSO (MPSO) was proposed to solve the model. Using an authentication instance, they compared and analyzed the performances of the MPSO method and the mixed integer programming method in solving the model. Chen et al. [309] proposed a sliding mode extremum seeking control (SMESC) of chaos embedded PSO (CEPSO) algorithm, applied to the design of MPP tracking in wind power systems. Its features were that the control parameters in SMESC were optimized by CEPSO, making it unnecessary to change the output power of different wind turbines, the designed in-repetition rate was reduced, and the system control efficiency was increased. Seyedmahmoudian et al. [310] aimed to employ PSO in MPP detection, while most conventional MPP tracking methods developed errors under certain circumstances and reduced the efficiency of PV systems even further. Aman et al. [311] presented a new approach for optimum simultaneous multi-DG placement and sizing based on maximization of system loadability without violating the system constraints. DG penetration level, line limit, and voltage magnitudes were considered as system constraints. Hybrid PSO algorithm was also proposed to find the optimum solution considering maximization of system loadability and the corresponding minimum power losses. Xiao and Huang [312] proposed an optimization method combining PSO with FEA to design the heating system for rapid thermal cycling molding mold, aiming at high heating efficiency and uniform cavity surface temperature distribution. Lian et al. [313] proposed a MPP tracking method based on PSO in order to track the global maximum point. The proposed method was better than conventional MPP tracking methods such as perturb-and-observe method that could only track the first local maximum point and stop progressing to the next maximum point.

4.8. Medical Engineering. In medicine, Qasem and Shamsuddin [314] introduced a time variant MOPSO (TVMOPSO) of RBF network for diagnosing the medical diseases. The approach was tested on three standard datasets from UCI machine learning repository. Zhang et al. [315] developed an adaptive CPSO to train the parameters of feed-forward neural network, with the aim of accurate classification of magnetic resonance (MR) brain images. Their results on 160 images

showed the classification accuracy of the proposed method was 98.75%. Guo et al. [316] proposed an adaptive PSO with neighborhood search to obtain the position and orientation of the medical microdevices. Experimental results showed that the tracking method was valid and the modified algorithm succeeded in dealing with the nonlinear system of equations in localization. Chang et al. [317] applied a PSO model to construct a decision-making system for diseases identification. The average forecasting accuracy for breast cancer was 97.4% and for liver disorders was 76.8%. Chen et al. [318] proposed an analytical approach by integrating PSO and the 1-NN method. The proposed approach was applied to an actual case on the diagnosis of obstructive sleep apnea (OSA). Sung and Chiang [319] examined wireless sensor network with real-time remote identification using the Android study of things (HCIOT) platform in community healthcare. An improved PSO method was proposed to efficiently enhance physiological multisensors data fusion measurement precision in the Internet of Things (IOT) system. Cruz-Aceves et al. [320] presented a novel image segmentation method based on multiple active contours driven by PSO (MACPSO), which was used to segment the human heart and the human left ventricle from datasets of sequential computed tomography and MR images, respectively. Subasi [321] proposed a novel PSO-SVM model that hybridized the PSO and SVM in order to improve the electromyography (EMG) signal classification accuracy. Zhang et al. [322] proposed a novel hybrid system to classify a given MR brain image as either normal or abnormal. They used PSO to optimize the parameters C and σ , which were the key parameters of kernel SVM (KSVM). The results showed that the PSO was more effective to build optimal KSVM than random selection method. Mangat and Vig [323] discussed a rule mining classifier called DA-AC (dynamic adaptive-associative classifier), which was based on a dynamic particle swarm optimizer. It could avoid premature convergence and provided a better value in every dimension. The method was then applied to predict life expectancy of postoperative thoracic surgery patients. Mandal et al. [324] formulated the fitting energy minimization problem to be solved using PSO technique. The algorithm was developed for two-phase level set implementation of the Chan and Vese model and it was successfully utilized for both scalar-valued and vector-valued medical images. Hsieh et al. [325] proposed a class of hyperrectangular composite neural networks (HRCNNs) of which synaptic weights could be interpreted as a set of crisp If-Then rules and proposed a PSO-based fuzzy hyperrectangular composite neural network (PFHRCNN) which applied PSO to trim the rules generated by a trained HRCNN while the recognition performance would not be degraded or even be improved. Ganapathy et al. [326] proposed a new pattern classification system by combining temporal features with fuzzy min-max (TFMM) neural network based classifier for effective decision support in medical diagnosis. A PSO algorithm based rule extractor was also proposed for improving the detection accuracy.

4.9. Chemical Engineering. In chemistry, Thakral et al. [327] adopted PSO along with negative factor counting technique

and inverse iteration method (IIM) for investigating the electronic properties of a model binary copolymer, of Type I class of quasi one-dimensional polymeric superlattices. Parastar et al. [328] proposed a multivariate curve resolution-PSO (MCR-PSO) algorithm to exploit pure chromatographic and spectroscopic information from multicomponent hyphenated chromatographic signals. Khajeh et al. [329] developed a simple and fast method for preconcentration and determination of trace amount of methylene blue from water samples by silver nanoparticles based solid-phase extraction method and UV-Vis spectrophotometry. Response surface methodology and hybrid of ANN-PSO were used to develop predictive models for simulation and optimization of solid-phase extraction method. Wu et al. [330] predicted CO₂ solubility in polymers by RBF-ANN, based on a chaotic self-adaptive PSO (CSPSO) and fuzzy clustering method. The proposed method was also used to investigate solubility of CO₂ in polystyrene, polypropylene, poly(butylene succinate), and poly(butylene succinate-co-adipate), respectively. Khajeh and Dastafkan [331] proposed a hybrid of ANN and PSO to develop predictive models for simulation and optimization of solid-phase extraction method, with the aim of developing a simple and fast method for preconcentration and determination of trace amount of molybdenum from water samples by silver nanoparticles based solid-phase extraction method and UV-Vis spectrophotometry. Thakral et al. [332] investigated the electronic properties of the novel binary and ternary copolymers using PSO in combination with simple negative factor counting and inverse iteration method, using the ab initio Hartree Fock crystal orbital band structure results of the homopolymers, namely, polypyrrolo[3, 4-C]pyrrole, polythieno[3, 4-C]thiophene, and polyfurano[3, 4-C]furan. Sun and Li [333] provided two improved mathematical expressions of attenuation function to quantify the effect of water in the process of methanol transformed to olefins on SAPO-34. Double PSO was employed to minimize the error objective function and the calculated values agree well with the experimental data. Skvortsov [334] researched on the rotation ambiguity (RA) in multivariate curve resolution (MCR), which was an undesirable case when the physico-chemical constraints were not sufficiently strong to provide a unique resolution of the data matrix of the mixtures into spectra and concentration profiles of individual chemical components. They proposed a method to estimate RA with charged PSO (cPSO). cPSO-MCR was shown to be capable of estimating the strength of the constraints and of revealing RA in noisy data. Khansary and Sani [335] aimed for the prediction of liquid-liquid equilibria (LLE), Wilson, universal quasi chemical (UNIQUAC) and nonrandom two liquid (NRTL) models were used. Evolutionary algorithms such as GA and PSO were used for estimation of binary interaction parameters of these models. The reliability of GA and PSO in LLE applications was successfully approved. Nasimi and Irani [336] combined PSO with a BP algorithm to form a new learning algorithm for training artificial neural networks. This strategy was applied to model a highly nonlinear system of yeast fermentation bioreactor. Based on the results comparison, the PSO-BP model was found to be superior to the BP-ANN model in identification of nonlinear systems.

4.10. Biological Engineering. Saraswathi et al. [337] used a combination of integer-coded GA and PSO, coupled with the neural-network-based extreme learning machine (ELM), for gene selection and cancer classification. The proposed algorithm could handle sparse data and sample imbalance. Mansour et al. [338] presented a PSO-based algorithm for predicting protein structures in the 3D hydrophobic polar model. Starting from a small set of candidate solutions, their algorithm efficiently explored the search space and returns 3D protein structures with minimal energy. Karabulut and Ibriki [339] proposed a PSO-based motif-finding method that utilized a proven Bayesian scoring scheme as the fitness function to identify transcription factor binding sites, which was a vital task in contemporary biology, since it helped researchers to comprehend the regulatory mechanism of gene expression. Liu et al. [340] established a classifier based on the two-layer PSO (TLPSO) algorithm and uncertain training sample sets. Samples of diffuse large B cell lymphoma (DLBCL) gene expression data were used for training and validating. The classification stability and accuracy by the proposed TLPSO algorithm increased significantly compared with the results obtained by using algorithms known as PSO and *K*-means. Salahi et al. [341] applied a variant of PSO with linear decreasing inertia weight to solve the problem of finding the global minimum of multifunnel-shaped functions with many local minima, which was a well-known and interesting problem in computational biology. Du et al. [342] presented construction of gene network using combined QPSO and K2 algorithm. Results showed QPSO-K2 algorithm performed better than stand-alone K2 and QPSO for all datasets. Chuang et al. [343] compared patients with high and low facial emotion perception (FEP) performances ($n = 89$ and 93 , resp.). A PSO algorithm was used to identify the best single-nucleotide polymorphisms (SNP) barcodes, that is, the SNP combinations and genotypes that revealed the largest differences between the high and low FEP groups. Mandal and Mukhopadhyay [344] organized the feature selection problem as a graph-theoretic problem where a feature-dissimilarity graph was shaped from the data matrix. They proposed a multiobjective BPSO that optimized average node-weight and average edge-weight of the candidate subgraph simultaneously. The proposed algorithm was applied for identifying relevant and nonredundant disease-related genes from microarray gene expression data. Chen et al. [345] developed a novel method utilizing PSO combined with a decision tree as the classifier to achieve efficient gene selection from thousands of candidate genes that could contribute in identifying cancers. Based on statistical analysis, the proposed method outperformed other popular classifiers for all test datasets and was compatible to SVM for certain specific datasets. García Nieto et al. [346] focused on turbidity parameter and how it was influenced by other water quality parameters in order to simplify water quality controls since they were expensive and time consuming. Turbidity values were predicted here by using the hybrid PSO-SVM-based model from the remaining measured water quality parameters (input variables) in the Nalon River basin (Northern Spain) with success.

TABLE 3: Adherence to SI principles.

Principle	Adherence of PSO
Proximity principle	Particle swarm carried out N -dimensional space calculation over a series of time steps.
Quality principle	Particle swarm responds to quality factors $pbest$ & $gbest$.
Diverse response principle	Particle swarm responses allocated between $pbest$ and $gbest$.
Stability principle	Particle swarm changes state only when $gbest$ changes.
Adaptability principle	Particle swarm changes state every time $gbest$ changes.

5. Discussion and Conclusion

5.1. Adherence to SI Principles. Revisiting Section 2, we can find close adherence of PSO to the five SI principles by Millonas [15]. They can be expressed in Table 3. When the behavior of particle swarm explained below is reexamined, it is clear that the four principles of self-organization defined by Bonabeau et al. [14] are also fully satisfied.

5.2. Drawback, Improvement, and Innovation. Like all other SI-based optimization approaches, PSO has some drawbacks like premature, high computational complexity, slow convergence, sensitivity to parameters, and so forth. The reasons behind the problems are complicated. One potential reason is that PSO does not utilize the crossover operator as employed in GA or DE; hence the distribution of good information between candidates is not at a required level. Another reason may fall within the fact that PSO does not appropriately handle the relationship between exploitation (local search) and exploration (global search), so it usually converges to a local minimum quickly.

To address abovementioned problems, scholars have proposed many solutions listed in Section 3 (summarized in Table 4), which can be divided into following three types.

- (i) Major modifications, including quantum-behaved PSO, bare-bones PSO, chaotic PSO, fuzzy PSO, PSOT-VAC, OPSO, SPSO, and topology.
- (ii) Minor modifications, including constriction coefficient, velocity clamping, trap detection, adaptive parameter, fitness-scaling, surrogate modeling, cooperative mechanism, boundary shifting, position resetting, entropy map, ecological behavior, jumping-out strategy, preference strategy, neighborhood learning, and local search.
- (iii) Hybridization, PSO being hybridized with GA, SA, TS, AIS, ACO, HS, ABC, DE, and so forth.

Now there are over hundreds of both various PSO variants and test functions at present. It is impossible for each newly proposed PSO variant to compare with all other variants and to go through all test functions. Therefore, it is difficult to proclaim which type of modification is better or promising. In our opinion, it is important and essential to create a platform, to which the authors who proposed PSO variants can submit their programs. After the comprehensive and fair comparison, we then can conclude which PSO variant is the winner. Particularly, the design of PSO without parameters is worth studying.

Success of PSO as a single-objective optimizer for processing continuous optimization problem has motivated researchers to extend the use of PSO to other optimization fields, like multiobjective optimization, constrained optimization, and discrete optimization (binary optimization and combinatorial optimization).

- (i) For multiobjective optimization, the optimizer needs to consider Pareto dominance every time it updates particles and store nondominated solutions to approximate the Pareto front.
- (ii) For constrained optimization, both infeasible and feasible solutions should be generated at the search stage, and constraints are dealt with when evaluating solutions using penalty factor
- (iii) For discrete optimization, a commonly used method is to map the discrete space to a continuous domain and apply standard PSO, followed by demapping the results. However, that method is not efficient and costs high computational resources. Particular PSO-based methods have been developed as discrete PSO, binary PSO, and their variants.

Study of PSO in those fields had impressive achievements; however, lack of in-depth research on theoretical aspects impaired its application potentials. We believe it should be interesting to perform a more comprehensive theoretical study of both the run-time and convergence properties of PSO and its variants. Other aspects such as fitness landscapes and dynamics of PSO are also very attractive theoretical research directions.

For the programmers, writing parallel programs are more difficult than to write sequential programs, since they need to take into consideration the communication and synchronization between different subtasks. Therefore, there are merely a few publications related to PSO implemented in parallel. Some new computational forms are worth investigating within the next years, such as performing PSO in computer clusters, in grid computing, and in MPP (massively parallel processor). In addition to them, different computational levels of parallel computing are also interesting: bit-level, instruction-level, data-level, and task-level.

5.3. Trends of Applications. The first practical application of PSO was in the field of ANN training and was reported together with the algorithm itself [6]. Many areas of application were explored ever since. In addition to what we discussed in Section 4 (see Table 5), there are thousands of publications reporting application of PSO to other fields. Due to the page limit, we cannot list all of them.

TABLE 4: Different modifications of PSO algorithm.

Contributions	Studies
QPSO	Jau et al. [20], Jamalipour et al. [21], Bagheri et al. [22], Tang et al. [23], Davoodi et al. [24], Li and Xiao [25], Yumin and Li [26], Jia et al. [27], and Gholizadeh and Moghadas [28]
BBPSO	Zhang et al. [30], Zhang et al. [31], Zhang et al. [32], Zhang et al. [33], Blackwell [34], Wang et al. [35], Jiang and Wang [36], Liu et al. [37], Campos et al. [38], and Zhang et al. [39]
CPSO	Chuang et al. [40], Zhang and Wu [41], Dai et al. [42], Li et al. [43], Wu et al. [44], Zhang et al. [45], Zhang et al. [46], Yang et al. [47], Son [48], He et al. [49], Zeng and Sun [50], and Pluhacek et al. [51]
FPSO	Juang et al. [52], Alfi and Fateh [53], Yang et al. [54], Norouzzadeh et al. [55], Robati et al. [56], Khan and Engelbrecht [57], Galzina et al. [58], Nafar et al. [59], Aminian and Teshnehlab [60], and Chai et al. [61]
PSOTVAC	Cai et al. [62], Chaturvedi et al. [63], Boonyaritdachochoai et al. [64], Sun et al. [65], Abedinia et al. [66], Mohammadi-Ivatloo et al. [67], Mohammadi-Ivatloo et al. [68], Pookpant and Ongsakul [69], Abedinia et al. [70], Abdullah et al. [71], and Chih et al. [72]
OPSO	Dhahri and Alimi [73], Wang et al. [74], Dong et al. [75], Gao et al. [76], Khan et al. [77], Kaucic [78], Dai et al. [79], and Muñoz et al. [80]
SPSO	Guochu [81], Pedersen and Chipperfield [82], Martins et al. [83], Panda et al. [84], Vastrakar and Padhy [85], and Yeh [86]
Population topology	Wang and Watada [87], Jiang et al. [88], Marinakis and Marinaki [89], Rada-Vilela et al. [90], Wang et al. [91], Fu et al. [92], Ni and Deng [93], Beheshti et al. [94], Lim and Isa [95], and Kalayci and Gupta [96]
Other modifications	Chuang et al. [97], Shi and Liu [98], Zhang et al. [99], Liu et al. [100], Shen et al. [101], Lin et al. [102], Wang and Watada [103], Li et al. [104], Lu et al. [105], Mattos et al. [106], Wu et al. [107], Lim and Mat Isa [108], Shimizu et al. [109], and Fister Jr. et al. [110]
PSO-GA	Kuo and Hong [111], Chen and Kurniawan [112], Nazir et al. [113], Vidhya and Kumar [114], Xiao et al. [115], and Ghamisi and Benediktsson [116]
PSO-AIS	Tang et al. [117], Zhang et al. [118], Ibrahim et al. [119], Kuo et al. [120], Liu et al. [121], and Darzi et al. [122]
PSO-TS	Li et al. [123], Nakano et al. [124], Zhang et al. [125], Ktari and Chabchoub [126], and Wang et al. [127]
PSO-ACO	Chen and Chien [128], Xiao et al. [129], Kiran et al. [130], Huang et al. [131], Rahmani et al. [132], and Elloumi [133]
PSO-SA	Sait et al. [134], Jiang and Zou [135], Niknam et al. [136], Khoshahval et al. [137], Du et al. [138], Zhang et al. [139], and Geng et al. [140]
PSO-ABC	El-Abd [141], Sharma et al. [142], Kiran and Gündüz [143], and Vitorino et al. [144]
PSO-DE	Maione and Punzi [145], Fu et al. [146], Vasundhara et al. [147], Yu et al. [148], Wang et al. [149], and Yadav and Deep [150]
Other hybridization	Xu et al. [151], Mohanty et al. [152], and Guo et al. [153]
Multiple objective	Qiu et al. [154], Chen et al. [155], Ghanei et al. [156], Duan et al. [157], Amiryousefi et al. [158], Ganguly [159], Zhang et al. [160], Perera et al. [161], and Cheng et al. [162]
Constrained optimization	Daneshyari and Yen [163], Afshar [164], Koulinas et al. [165], Shan and Ren [166], Yeh and Chien [167], Singh et al. [168], Paliwal et al. [169], Cui et al. [170], and Shou et al. [171]
Discrete optimization	Chen and Ludwig [172], Shen et al. [173], Chen et al. [174], Cai et al. [175], Kashan et al. [176], Xu et al. [177], Garg and Singh [178], Zong et al. [179], and Ezzeldin et al. [180]
Binary optimization	Zhai and He [181], Sarath and Ravi [182], Taha and Abu Al Nadi [183], El-Maleh et al. [184], Erturk et al. [185], Zhang et al. [186], Yin et al. [187], Yang et al. [188], and Ganesh et al. [189]
Parameter choice	Kumar and Chaturvedi [191], Zhang et al. [192], Yang [193], Sun et al. [194], Yassin et al. [195], Wang et al. [196], Hao et al. [197], Xu [198], Chauhan et al. [199], Zhang et al. [200], and Kanemasa and Aiyoshi [201]
Convergence analysis	Wang and Shen [202], Sun et al. [203], Kurihara and Jin'no [204], Lin [205], Zhang et al. [206], Lin et al. [207], and Kim and Li [208]
Multicore	Waintraub et al. [209] and Yu [210]
GPU	Hung and Wang [211], Rymut et al. [212], Kumar et al. [213], Awwad et al. [214], and Chen et al. [215]
Cloud	Liu et al. [216], Xu and You [217], Ramezani et al. [218], Govindarajan et al. [219], and Ramezani et al. [220]

TABLE 5: Application areas of PSO algorithm.

Area	Publication
Electrical and electronic engineering	Ganguly et al. [221], Komsiyah [222], Feng et al. [223], Pekşen et al. [224], Yang et al. [225], de Mendonça et al. [226], Liu et al. [227], Aich and Banerjee [228], Chou et al. [229], Lee et al. [230], Thakral and Bakhshi [231], Fister et al. [232], Aghaei et al. [233], Selakov et al. [234], Shirvany et al. [235], and Tungadio et al. [236]
Automatic control	Cai and Yang [237], Kolomvatsos and Hadjieftymiades [238], Pandey et al. [239], Štimac et al. [240], Nedic et al. [241], Chang and Chen [242], Xiang et al. [243], Danapalasingam [244], Mahmoodabadi et al. [245], Zhong et al. [246], Perng et al. [247], Huang and Li [248], and Nisha and Pillai [249]
Communication	Yousefi et al. [250], Sun et al. [251], Yongqiang et al. [252], Chiu et al. [253], Zubair and Moinuddin [255], Kim and Lee [256], Yazgan and Hakki Cavdar [257], Rabadly and Ababneh [258], Das et al. [259], Scott-Hayward and Garcia-Palacios [260], Omidvar and Mohammadi [261], and Kuila and Jana [262]
Operations	Liu and Wang [263], Che et al. [264], Hajipour and Pasandideh [265], Cabrerizo et al. [266], Prescilla and Selvakumar [267], Duan et al. [268], Belmecheri et al. [269], Hu et al. [270], Al Badawia and Shatnawi [271], Goksal et al. [272], Zhang et al. [273], and Kechagiopoulos and Beligiannis [274]
Mechanical engineering	Ming et al. [275], Chen et al. [276], Mohan et al. [277], Chen et al. [278], Zhang et al. [279], Wang et al. [280], Lazrag et al. [281], Lee et al. [282], Lake et al. [283], Vosoughi and Gerist [284], Ribeiro et al. [285], Kitak et al. [286], and Kalatehjari et al. [287]
Civil engineering	Ashuri and Tavakolan [288], Bozorgi-Amiri et al. [289], Sadoghi Yazdi et al. [290], Bolat et al. [291], Babu and Vijayalakshmi [292], Mohan et al. [293], Asadnia et al. [294], Yan et al. [295], Wang and Li [296], Sadeghi et al. [297], and Kanović et al. [298]
Fuel and energy	Mandal et al. [299], Chao [300], Chen et al. [301], Hu et al. [302], Tabet et al. [303], Bahrami et al. [304], Askarzadeh [305], Sharafi and Elmekawy [306], García-Triviño et al. [307], Biao et al. [308], Chen et al. [309], Seyedmahmoudian et al. [310], Aman et al. [311], Xiao and Huang [312], and Lian et al. [313]
Medicine engineering	Qasem and Shamsuddin [314], Zhang et al. [315], Guo et al. [316], Chang et al. [317], Chen et al. [318], Sung and Chiang [319], Cruz-Aceves et al. [320], Subasi [321], Zhang et al. [322], Mangat and Vig [323], Mandal et al. [324], Hsieh et al. [325], and Ganapathy et al. [326]
Chemical engineering	Thakral et al. [327], Parastar et al. [328], Khajeh et al. [329], Wu et al. [330], Khajeh and Dastafkan [331], Thakral et al. [332], Sun and Li [333], Skvortsova [334], Khansary and Sani [335], and Nasimi and Irani [336]
Biological engineering	Saraswathi et al. [337], Mansour et al. [338], Karabulut and Ibriki [339], Liu et al. [340], Salahi et al. [341], Du et al. [342], Chuang et al. [343], Mandal and Mukhopadhyay [344], Chen et al. [345], and García Nieto et al. [346]

From the table, it is clear that the distribution of PSO exceeded our expectations. It is now applied to nearly every discipline that comes across optimization problems. Generally, PSO can be applied for optimization in dynamic and uncertain environments. The engineers need to first transform the problem to an optimization problem and then apply PSO to solve it.

Several research topics need to be fully investigated in future, since there are so far either no or merely few publications applying PSO to the optimization problem in those fields including what follows.

- (i) Symbolic regression, which is a type of regression analysis that searches the space of mathematical expressions to find the model that best fits a given dataset, both in terms of accuracy and simplicity, is a crucially important theoretical and practical problem.
- (ii) Floorplanning is to design the layout of equipment in a factory or components on a computer chip to reduce manufacturing time.
- (iii) Weapon target assignment problem is to find an optimal assignment of a set of weapons of various types to a set of targets in order to maximize the expected damage done to the opponent.

(iv) Supply chain management is the systemic, strategic coordination of the traditional business functions and the tactics across these business functions within a particular company and across businesses within the supply chain, for the purposes of improving the long-term performance of the individual companies and the supply chain as a whole [347].

(v) Nurse scheduling problem [348] is to find an optimal way to assign nurses to shifts, typically with a set of hard constraints in which all valid solutions must follow and a set of soft constraints which define the relative quality of valid solutions.

(vi) Queuing theory is the mathematical study of waiting lines. In queueing theory a model is constructed so that queue lengths and waiting times can be predicted to make business decisions about the resources needed [349].

5.4. Publication Analysis. Figure 3 demonstrated the number of publications related to PSO against year. It is clearly observed that the number of publications increases exponentially from 2000 to 2006, and then it fluctuates constantly and continually around about 1,000 publications per year from 2007 to 2013. We can conclude that the research in PSO has

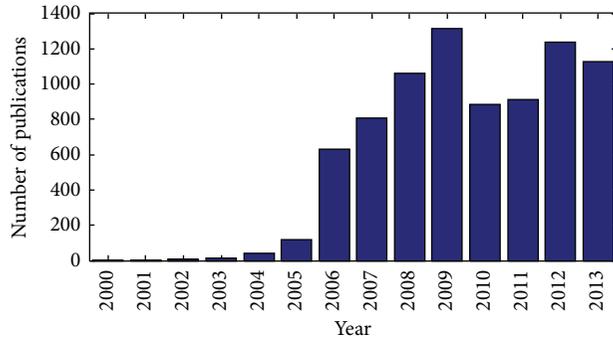


FIGURE 3: Publication number against year.

arrived at mature stage, and we expect a lot of more results within the coming few years.

Acronyms

ABC:	Artificial bee colony
ACO:	Ant colony optimization
AI:	Artificial intelligence
AIS:	Artificial immune system
ANN:	Artificial neural network
BA:	Bat algorithm
BBO:	Biogeography-based optimization
BBPSO:	Bare-bones PSO
BCI:	Brain-computer interface
BFO:	Bacterial foraging optimization
BP:	Back-propagation
BPSO:	Binary PSO
CI:	Computational intelligence
CPSO:	Chaotic PSO
CUDA:	Compute unified device architecture
DCT:	Discrete cosine transform
DE:	Differential evolution
DG:	Distributed generation
DPSO:	Discrete PSO
EC:	Evolutionary computation
EDP:	Economic dispatch problem
EEG:	Electroencephalography
ELD:	Economic load dispatch
EMG:	Electromyography
FCM:	Fuzzy c-means
FEA:	Finite element analysis
FPSO:	Fuzzy PSO
GA:	Genetic algorithm
GP:	Genetic programming
GPU:	Graphics processing unit
GSA:	Gravitational search algorithm
GSO:	Glowworm swarm optimization
HS:	Harmonic search
HDFS:	Hadoop distributed file system
KSVM:	Kernel SVM
MIMO:	Multivariable input and multivariable output
MLE:	Maximum likelihood estimate
MOL:	Many Optimizing Liaisons

MOPSO:	Multiple objective PSO
MPP:	Maximum power point
MR:	Magnetic resonance
MRP:	Multicast routing problem
MST:	Minimal spanning tree
OBL:	Opposition-based learning
OPSO:	Opposition-based PSO
PID:	Proportional-integral derivate
PSO:	Particle swarm optimization
PSOTVAC:	PSO with TVAC
PV:	Photovoltaic
QPSO:	Quantum-behaved PSO
RBF:	Radial basis function
SA:	Simulated annealing
SBS:	Sequential backward selection
SFS:	Sequential forward selection
SI:	Swarm intelligence
SPSO:	Simplified PSO
SQP:	Sequential quadratic programming
SVM:	Support vector machine
SVRM:	Support vector regression machine
TS:	Tabu search
TSP:	Traveling salesman problem
TVAC:	Time-varying acceleration coefficients
UAV:	Unmanned aerial vehicle
UCAV:	Unmanned combat aerial vehicle
UWB:	Ultrawideband
VRP:	Vehicle routing problem
WT:	Wavelet transform.

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 NSFC (no. 610011024), Program of Natural Science Research of Jiangsu Higher Education Institutions of China (no. 14KJB520021), Jiangsu Key Laboratory of 3D Printing Equipment and Manufacturing (BM2013006), and Nanjing Normal University Research Foundation for Talented Scholars (no. 2013119XGQ0061).

References

- [1] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice Hall, New York, NY, USA, 2010.
- [2] B. L. Agarwal, *Basic Statistics*, New Age International, 2006.
- [3] K. E. Voges and N. Pope, *Business Applications and Computational Intelligence*, Idea Group Publishing, 2006.
- [4] V. Kothari, J. Anuradha, S. Shah, and P. Mittal, "A survey on particle swarm optimization in feature selection," in *Global Trends in Information Systems and Software Applications: Proceedings of the 4th International Conference, ObCom 2011, Vellore, TN, India, December 9–11, 2011, Part 2*, P. V. Krishna, M. R. Babu, and E. Ariwa, Eds., vol. 270, pp. 192–201, Springer, Berlin, Germany, 2012.

- [5] M. Dorigo, *Optimization, learning and natural algorithms [Ph.D. thesis]*, Politecnico di Milano, Milano, Italy, 1992.
- [6] 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.
- [7] 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.
- [8] S. Müller, S. Airaghi, J. Marchetto, and P. Koumoutsakos, "Optimization algorithms based on a model of bacterial chemotaxis," in *Proceedings of the 6th International Conference on Simulation of Adaptive Behavior: From Animals to Animats*, pp. 375–384, 2000.
- [9] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 52–67, 2002.
- [10] D. Karaboga and B. Basturk, "Artificial Bee Colony (ABC) optimization algorithm for solving constrained optimization problems," in *Proceedings of the Foundations of Fuzzy Logic and Soft Computing*, P. Melin, O. Castillo, L. T. Aguilar, and W. Pedrycz, Eds., pp. 789–798, Springer, New York, NY, USA, 2007.
- [11] K. N. Krishnanand and D. Ghose, "Detection of multiple source locations using a glowworm metaphor with applications to collective robotics," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '05)*, pp. 84–94, June 2005.
- [12] X. S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nicso 2010: Nature Inspired Cooperative Strategies*, J. R. Gonzalez, D. A. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor, Eds., pp. 65–74, Springer, Berlin, Germany, 2010.
- [13] Y. Zhang, P. Agarwal, V. Bhatnagar, S. Balochian, and X. Zhang, "Swarm intelligence and its applications 2014," *The Scientific World Journal*, vol. 2014, Article ID 204294, 4 pages, 2014.
- [14] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, New York, NY, USA, 1999.
- [15] M. Millonas, "Swarms, phase transitions and collective intelligence," in *Artificial life III*, C. Langton, Ed., pp. 417–445, Addison-Wesley, Reading, Mass, USA, 1994.
- [16] Y. Zhang, S. Balochian, P. Agarwal, V. Bhatnagar, and O. J. Housheya, "Artificial intelligence and its applications," *Mathematical Problems in Engineering*, vol. 2014, Article ID 840491, 10 pages, 2014.
- [17] F. Shahzad, S. Masood, and N. K. Khan, "Probabilistic opposition-based particle swarm optimization with velocity clamping," *Knowledge and Information Systems*, vol. 39, no. 3, pp. 703–737, 2014.
- [18] 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.
- [19] Y. Zhang, S. Wang, and Z. Dong, "Classification of alzheimer disease based on structural magnetic resonance imaging by kernel support vector machine decision tree," *Progress in Electromagnetics Research*, vol. 144, pp. 171–184, 2014.
- [20] Y.-M. Jau, K.-L. Su, C.-J. Wu, and J.-T. Jeng, "Modified quantum-behaved particle swarm optimization for parameters estimation of generalized nonlinear multi-regressions model based on Choquet integral with outliers," *Applied Mathematics and Computation*, vol. 221, pp. 282–295, 2013.
- [21] M. Jamalipour, R. Sayareh, M. Gharib, F. Khoshahval, and M. R. Karimi, "Quantum behaved particle swarm optimization with differential mutation operator applied to WWER-1000 in-core fuel management optimization," *Annals of Nuclear Energy*, vol. 54, pp. 134–140, 2013.
- [22] A. Bagheri, H. Mohammadi Peyhani, and M. Akbari, "Financial forecasting using ANFIS networks with Quantum-behaved Particle Swarm Optimization," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6235–6250, 2014.
- [23] D. Y. Tang, Y. Cai, J. Zhao, and Y. Xue, "A quantum-behaved particle swarm optimization with memetic algorithm and memory for continuous non-linear large scale problems," *Information Sciences*, vol. 289, pp. 162–189, 2014.
- [24] E. Davoodi, M. T. Hagh, and S. G. Zadeh, "A hybrid improved quantum-behaved particle swarm optimization-simplex method (IQPSOS) to solve power system load flow problems," *Applied Soft Computing Journal*, vol. 21, pp. 171–179, 2014.
- [25] P. Li and H. Xiao, "An improved quantum-behaved particle swarm optimization algorithm," *Applied Intelligence*, vol. 40, no. 3, pp. 479–496, 2014.
- [26] D. Yumin and Z. Li, "Quantum behaved particle swarm optimization algorithm based on artificial fish swarm," *Mathematical Problems in Engineering*, vol. 2014, Article ID 592682, 10 pages, 2014.
- [27] P. F. Jia, F. C. Tian, S. Fan, Q. H. He, J. W. Feng, and S. X. Yang, "A novel sensor array and classifier optimization method of electronic nose based on enhanced quantum-behaved particle swarm optimization," *Sensor Review*, vol. 34, no. 3, pp. 304–311, 2014.
- [28] S. Gholizadeh and R. K. Moghadas, "Performance-based optimum design of steel frames by an improved quantum particle swarm optimization," *Advances in Structural Engineering*, vol. 17, no. 2, pp. 143–156, 2014.
- [29] J. Kennedy, "Bare bones particle swarms," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '03)*, pp. 80–87, Indianapolis, Ind, USA, 2003.
- [30] H. Zhang, D. D. Kennedy, G. P. Rangaiah, and A. Bonilla-Petriciolet, "Novel bare-bones particle swarm optimization and its performance for modeling vapor-liquid equilibrium data," *Fluid Phase Equilibria*, vol. 301, no. 1, pp. 33–45, 2011.
- [31] J. Q. Zhang, L. Ni, J. Yao, W. Wang, and Z. Tang, "Adaptive bare bones particle swarm inspired by cloud model," *IEICE Transactions on Information and Systems*, vol. 94, no. 8, pp. 1527–1538, 2011.
- [32] H. Zhang, J. A. Fernández-Vargas, G. P. Rangaiah, A. Bonilla-Petriciolet, and J. G. Segovia-Hernández, "Evaluation of integrated differential evolution and unified bare-bones particle swarm optimization for phase equilibrium and stability problems," *Fluid Phase Equilibria*, vol. 310, no. 1-2, pp. 129–141, 2011.
- [33] Y. Zhang, D.-W. Gong, and Z. Ding, "A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch," *Information Sciences*, vol. 192, pp. 213–227, 2012.
- [34] T. Blackwell, "A study of collapse in bare bones particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 3, pp. 354–372, 2012.
- [35] P. P. Wang, L. Shi, Y. Zhang, and L. Han, "A hybrid simplex search and modified bare-bones particle swarm optimization," *Chinese Journal of Electronics*, vol. 22, no. 1, pp. 104–108, 2013.

- [36] B. Jiang and N. Wang, "Cooperative bare-bone particle swarm optimization for data clustering," *Soft Computing*, vol. 18, no. 6, pp. 1079–1091, 2014.
- [37] H. Liu, G. Ding, and B. Wang, "Bare-bones particle swarm optimization with disruption operator," *Applied Mathematics and Computation*, vol. 238, pp. 106–122, 2014.
- [38] M. Campos, R. A. Krohling, and I. Enriquez, "Bare bones particle swarm optimization with scale matrix adaptation," *IEEE Transactions on Cybernetics*, vol. 44, no. 9, pp. 1567–1578, 2014.
- [39] Y. Zhang, D. Gong, Y. Hu, and W. Zhang, "Feature selection algorithm based on bare bones particle swarm optimization," *Neurocomputing*, vol. 148, pp. 150–157, 2015.
- [40] L.-Y. Chuang, S.-W. Tsai, and C.-H. Yang, "Chaotic catfish particle swarm optimization for solving global numerical optimization problems," *Applied Mathematics and Computation*, vol. 217, no. 16, pp. 6900–6916, 2011.
- [41] Y. Zhang and L. Wu, "Crop classification by forward neural network with adaptive chaotic particle swarm optimization," *Sensors*, vol. 11, no. 5, pp. 4721–4743, 2011.
- [42] Y. S. Dai, Y. Wei, J. Chen, Y. Zhang, and J. Ding, "Seismic wavelet estimation based on adaptive chaotic embedded particle swarm optimization algorithm," in *Proceedings of the 5th International Symposium on Computational Intelligence and Design (ISCID '12)*, vol. 2, pp. 57–60, October 2012.
- [43] C. S. Li, J. Zhou, P. Kou, and J. Xiao, "A novel chaotic particle swarm optimization based fuzzy clustering algorithm," *Neurocomputing*, vol. 83, pp. 98–109, 2012.
- [44] Q. Wu, R. Law, E. Wu, and J. Lin, "A hybrid-forecasting model reducing Gaussian noise based on the Gaussian support vector regression machine and chaotic particle swarm optimization," *Information Sciences*, vol. 238, pp. 96–110, 2013.
- [45] 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.
- [46] Q. Zhang, Z. Li, C. J. Zhou, and X. P. Wei, "Bayesian network structure learning based on the chaotic particle swarm optimization algorithm," *Genetics and Molecular Research*, vol. 12, no. 4, pp. 4468–4479, 2013.
- [47] C.-H. Yang, Y.-D. Lin, L.-Y. Chuang, and H.-W. Chang, "Double-bottom chaotic map particle swarm optimization based on chi-square test to determine gene-gene interactions," *BioMed Research International*, vol. 2014, Article ID 172049, 10 pages, 2014.
- [48] L. H. Son, "Optimizing municipal solid waste collection using chaotic particle swarm optimization in GIS based environments: a case study at Danang city, Vietnam," *Expert Systems with Applications*, vol. 41, no. 18, pp. 8062–8074, 2014.
- [49] H.-D. He, W.-Z. Lu, and Y. Xue, "Prediction of particulate matter at street level using artificial neural networks coupling with chaotic particle swarm optimization algorithm," *Building and Environment*, vol. 78, pp. 111–117, 2014.
- [50] Y. J. Zeng and Y. G. Sun, "An improved particle swarm optimization for the combined heat and power dynamic economic dispatch problem," *Electric Power Components and Systems*, vol. 42, no. 15, pp. 1700–1716, 2014.
- [51] M. Pluhacek, R. Senkerik, and I. Zelinka, "Particle swarm optimization algorithm driven by multichaotic number generator," *Soft Computing*, vol. 18, no. 4, pp. 631–639, 2014.
- [52] Y.-T. Juang, S.-L. Tung, and H.-C. Chiu, "Adaptive fuzzy particle swarm optimization for global optimization of multimodal functions," *Information Sciences*, vol. 181, no. 20, pp. 4539–4549, 2011.
- [53] A. Alfi and M.-M. Fateh, "Intelligent identification and control using improved fuzzy particle swarm optimization," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12312–12317, 2011.
- [54] W.-A. Yang, Y. Guo, and W.-H. Liao, "Optimization of multi-pass face milling using a fuzzy particle swarm optimization algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 54, no. 1–4, pp. 45–57, 2011.
- [55] M. S. Norouzzadeh, M. R. Ahmadzadeh, and M. Palhang, "LADPSO: using fuzzy logic to conduct PSO algorithm," *Applied Intelligence*, vol. 37, no. 2, pp. 290–304, 2012.
- [56] A. Robati, G. A. Barani, H. Nezam Abadi Pour, M. J. Fadaee, and J. R. Anaraki, "Balanced fuzzy particle swarm optimization," *Applied Mathematical Modelling*, vol. 36, no. 5, pp. 2169–2177, 2012.
- [57] S. A. Khan and A. P. Engelbrecht, "A fuzzy particle swarm optimization algorithm for computer communication network topology design," *Applied Intelligence*, vol. 36, no. 1, pp. 161–177, 2012.
- [58] V. Galzina, R. Lujčić, and T. Šarić, "Adaptive fuzzy particle swarm optimization for flow-shop scheduling problem," *Tehnicki Vjesnik—Technical Gazette*, vol. 19, no. 1, pp. 151–157, 2012.
- [59] M. Nafar, G. B. Gharehpetian, and T. Niknam, "Using modified fuzzy particle swarm optimization algorithm for parameter estimation of surge arresters models," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 1, pp. 567–581, 2012.
- [60] E. Aminian and M. Teshnehlab, "A novel fuzzy particle swarm optimization," in *Proceedings of the 13th Iranian Conference on Fuzzy Systems*, pp. 1–6, IEEE, Qazvin, Iran, 2013.
- [61] R. Chai, S. H. Ling, G. P. Hunter, Y. Tran, and H. T. Nguyen, "Brain—computer interface classifier for wheelchair commands using neural network with fuzzy particle swarm optimization," *IEEE Journal of Biomedical and Health Informatics*, vol. 18, no. 5, pp. 1614–1624, 2014.
- [62] X. J. Cai, Y. Cui, and Y. Tan, "Predicted modified PSO with time-varying accelerator coefficients," *International Journal of Bio-Inspired Computation*, vol. 1, no. 1-2, pp. 50–60, 2009.
- [63] K. T. Chaturvedi, M. Pandit, and L. Srivastava, "Particle swarm optimization with time varying acceleration coefficients for non-convex economic power dispatch," *International Journal of Electrical Power & Energy Systems*, vol. 31, no. 6, pp. 249–257, 2009.
- [64] P. Boonyaritdachochai, C. Boonchuay, and W. Ongsakul, "Optimal congestion management in an electricity market using particle swarm optimization with time-varying acceleration coefficients," *Computers & Mathematics with Applications*, vol. 60, no. 4, pp. 1068–1077, 2010.
- [65] C. Y. Sun, H. N. Zhao, and Y. F. Wang, "A comparative analysis of PSO, HPSO, and HPSO-TVAC for data clustering," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 23, no. 1, pp. 51–62, 2011.
- [66] O. Abedinia, N. Amjady, and K. Kiani, "Optimal complex economic load dispatch solution using particle swarm optimization with time varying acceleration coefficient," *International Review of Electrical Engineering*, vol. 7, no. 2, pp. 4249–4256, 2012.
- [67] B. Mohammadi-Ivatloo, A. Rabiee, A. Soroudi, and M. Ehsan, "Iteration PSO with time varying acceleration coefficients for

- solving non-convex economic dispatch problems,” *International Journal of Electrical Power & Energy Systems*, vol. 42, no. 1, pp. 508–516, 2012.
- [68] B. Mohammadi-Ivatloo, M. Moradi-Dalvand, and A. Rabiee, “Combined heat and power economic dispatch problem solution using particle swarm optimization with time varying acceleration coefficients,” *Electric Power Systems Research*, vol. 95, pp. 9–18, 2013.
- [69] S. Pookpant and W. Ongsakul, “Optimal placement of wind turbines within wind farm using binary particle swarm optimization with time-varying acceleration coefficients,” *Renewable Energy*, vol. 55, pp. 266–276, 2013.
- [70] O. Abedinia, N. Amjady, A. Ghasemi, and Z. Hejrati, “Solution of economic load dispatch problem via hybrid particle swarm optimization with time-varying acceleration coefficients and bacteria foraging algorithm techniques,” *International Transactions on Electrical Energy Systems*, vol. 23, no. 8, pp. 1504–1522, 2013.
- [71] M. N. Abdullah, A. H. A. Bakar, N. A. Rahim, H. Mokhlis, H. A. Illias, and J. J. Jamian, “Modified particle swarm optimization with time varying acceleration coefficients for economic load dispatch with generator constraints,” *Journal of Electrical Engineering & Technology*, vol. 9, no. 1, pp. 15–26, 2014.
- [72] M. C. Chih, C.-J. Lin, M.-S. Chern, and T.-Y. Ou, “Particle swarm optimization with time-varying acceleration coefficients for the multidimensional knapsack problem,” *Applied Mathematical Modelling*, vol. 38, no. 4, pp. 1338–1350, 2014.
- [73] H. Dhahri and A. M. Alimi, “Opposition-based particle swarm optimization for the design of beta basis function neural network,” in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '10)*, pp. 1–8, 2010.
- [74] H. Wang, Z. Wu, S. Rahnamayan, Y. Liu, and M. Ventresca, “Enhancing particle swarm optimization using generalized opposition-based learning,” *Information Sciences*, vol. 181, no. 20, pp. 4699–4714, 2011.
- [75] N. Dong, C.-H. Wu, W.-H. Ip, Z.-Q. Chen, C.-Y. Chan, and K.-L. Yung, “An opposition-based chaotic GA/PSO hybrid algorithm and its application in circle detection,” *Computers & Mathematics with Applications*, vol. 64, no. 6, pp. 1886–1902, 2012.
- [76] W. F. Gao, S. Y. Liu, and L. L. Huang, “Particle swarm optimization with chaotic opposition-based population initialization and stochastic search technique,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 11, pp. 4316–4327, 2012.
- [77] N. K. Khan, A. R. Baig, and M. A. Iqbal, “Opposition-based discrete PSO using natural encoding for classification rule discovery,” *International Journal of Advanced Robotic Systems*, vol. 9, 2012.
- [78] M. Kaucic, “A multi-start opposition-based particle swarm optimization algorithm with adaptive velocity for bound constrained global optimization,” *Journal of Global Optimization*, vol. 55, no. 1, pp. 165–188, 2013.
- [79] Y. Dai, L. Liu, and S. Feng, “On the identification of coupled pitch and heave motions using opposition-based particle swarm optimization,” *Mathematical Problems in Engineering*, vol. 2014, Article ID 784049, 10 pages, 2014.
- [80] D. M. Muñoz, C. H. Llanos, L. D. S. Coelho, and M. Ayala-Rincón, “Hardware opposition-based PSO applied to mobile robot controllers,” *Engineering Applications of Artificial Intelligence*, vol. 28, pp. 64–77, 2014.
- [81] C. Guochu, “Simplified particle swarm optimization algorithm based on particles classification,” in *Proceedings of the 6th International Conference on Natural Computation (ICNC' 10)*, pp. 2701–2705, August 2010.
- [82] M. E. H. Pedersen and A. J. Chipperfield, “Simplifying particle swarm optimization,” *Applied Soft Computing Journal*, vol. 10, no. 2, pp. 618–628, 2010.
- [83] C. H. Martins, R. P. B. dos Santos, and F. L. Santos, “Simplified particle swarm optimization algorithm,” *Acta Scientiarum—Technology*, vol. 34, no. 1, pp. 21–25, 2012.
- [84] S. Panda, B. K. Sahu, and P. K. Mohanty, “Design and performance analysis of PID controller for an automatic voltage regulator system using simplified particle swarm optimization,” *Journal of the Franklin Institute*, vol. 349, no. 8, pp. 2609–2625, 2012.
- [85] N. K. Vastrakar and P. K. Padhy, “Simplified PSO PI-PD controller for unstable processes,” in *Proceedings of the 4th International Conference on Intelligent Systems, Modelling & Simulation (ISMS '13)*, pp. 350–354, IEEE, Bangkok, Thailand, January 2013.
- [86] W.-C. Yeh, “New parameter-free simplified swarm optimization for artificial neural network training and its application in the prediction of time series,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 4, pp. 661–665, 2013.
- [87] S. Wang and J. M. Watada, “Two-stage fuzzy stochastic programming with Value-at-Risk criteria,” *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 1044–1056, 2011.
- [88] B. Jiang, N. Wang, and L. P. Wang, “Particle swarm optimization with age-group topology for multimodal functions and data clustering,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 11, pp. 3134–3145, 2013.
- [89] Y. Marinakis and M. Marinaki, “Particle swarm optimization with expanding neighborhood topology for the permutation flowshop scheduling problem,” *Soft Computing*, vol. 17, no. 7, pp. 1159–1173, 2013.
- [90] J. Rada-Vilela, M. J. Zhang, and W. Seah, “A performance study on synchronicity and neighborhood size in particle swarm optimization,” *Soft Computing*, vol. 17, no. 6, pp. 1019–1030, 2013.
- [91] H. Wang, H. Sun, C. Li, S. Rahnamayan, and J.-S. Pan, “Diversity enhanced particle swarm optimization with neighborhood search,” *Information Sciences*, vol. 223, pp. 119–135, 2013.
- [92] X. Fu, W. Liu, B. Zhang, and H. Deng, “Quantum behaved particle swarm optimization with neighborhood search for numerical optimization,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 469723, 10 pages, 2013.
- [93] Q. J. Ni and J. M. Deng, “A new logistic dynamic particle swarm optimization algorithm based on random topology,” *The Scientific World Journal*, vol. 2013, Article ID 409167, 8 pages, 2013.
- [94] Z. Beheshti, S. M. Shamsuddin, and S. Sulaiman, “Fusion global-local-topology particle swarm optimization for global optimization problems,” *Mathematical Problems in Engineering*, vol. 2014, Article ID 907386, 19 pages, 2014.
- [95] W. H. Lim and N. A. M. Isa, “Particle swarm optimization with increasing topology connectivity,” *Engineering Applications of Artificial Intelligence*, vol. 27, pp. 80–102, 2014.
- [96] C. B. Kalayci and S. M. Gupta, “A particle swarm optimization algorithm with neighborhood-based mutation for sequence-dependent disassembly line balancing problem,” *International Journal of Advanced Manufacturing Technology*, vol. 69, no. 1–4, pp. 197–209, 2013.

- [97] L.-Y. Chuang, S.-W. Tsai, and C.-H. Yang, "Catfish particle swarm optimization," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '08)*, IEEE, St. Louis, Mo, USA, September 2008.
- [98] R. F. Shi and X. J. Liu, "A hybrid improved particle swarm optimization based on dynamic parameters control and metropolis accept rule strategy," in *Proceedings of the 3rd International Conference on Genetic and Evolutionary Computing (WGECC '09)*, T. L. Huang, L. F. Li, and M. Zhao, Eds., pp. 649–653, IEEE Computer Society, Los Alamitos, Calif, USA, October 2009.
- [99] J. H. Zhang, Y. Wang, R. Wang, and G. Hou, "Bidding strategy based on adaptive particle swarm optimization for electricity market," in *Proceedings of the 8th World Congress on Intelligent Control and Automation (WCICA '10)*, pp. 3207–3210, IEEE, Jinan, China, July 2010.
- [100] 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.
- [101] Y. X. Shen, G. Y. Wang, and C. M. Tao, "Particle swarm optimization with novel processing strategy and its application," *International Journal of Computational Intelligence Systems*, vol. 4, no. 1, pp. 100–111, 2011.
- [102] L. Lin, Z. Ji, S. He, and Z. Zhu, "A crown jewel defense strategy based particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '12)*, June 2012.
- [103] S. M. Wang and J. Watada, "A hybrid modified PSO approach to VaR-based facility location problems with variable capacity in fuzzy random uncertainty," *Information Sciences*, vol. 192, pp. 3–18, 2012.
- [104] Z. Li, Z. Tian, Y. Xie, R. Huang, and J. Tan, "A knowledge-based heuristic particle swarm optimization approach with the adjustment strategy for the weighted circle packing problem," *Computers & Mathematics with Applications*, vol. 66, no. 10, pp. 1758–1769, 2013.
- [105] Y. C. Lu, J. C. Jan, S. L. Hung, and G. H. Hung, "Enhancing particle swarm optimization algorithm using two new strategies for optimizing design of truss structures," *Engineering Optimization*, vol. 45, no. 10, pp. 1251–1271, 2013.
- [106] C. L. C. Mattos, G. A. Barreto, and F. R. P. Cavalcanti, "An improved hybrid particle swarm optimization algorithm applied to economic modeling of radio resource allocation," *Electronic Commerce Research*, vol. 14, no. 1, pp. 51–70, 2014.
- [107] G. Wu, W. Pedrycz, M. Ma, D. Qiu, H. Li, and J. Liu, "A particle swarm optimization variant with an inner variable learning strategy," *The Scientific World Journal*, vol. 2014, Article ID 713490, 15 pages, 2014.
- [108] W. H. Lim and N. A. Mat Isa, "An adaptive two-layer particle swarm optimization with elitist learning strategy," *Information Sciences*, vol. 273, pp. 49–72, 2014.
- [109] Y. Shimizu, T. Sakaguchi, and T. Miura, "Parallel computing for huge scale logistics optimization through binary PSO associated with topological comparison," *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, vol. 8, no. 1, Article ID JAMDSM0005, 2014.
- [110] I. Fister Jr., M. Perc, K. Ljubić, S. M. Kamal, and A. Iglesias, "Particle swarm optimization for automatic creation of complex graphic characters," *Chaos, Solitons & Fractals*, vol. 73, pp. 29–35, 2015.
- [111] R. J. Kuo and C. W. Hong, "Integration of genetic algorithm and particle swarm optimization for investment portfolio optimization," *Applied Mathematics and Information Sciences*, vol. 7, no. 6, pp. 2397–2408, 2013.
- [112] W. C. Chen and D. Kurniawan, "Process parameters optimization for multiple quality characteristics in plastic injection molding using Taguchi method, BPNN, GA, and hybrid PSO-GA," *International Journal of Precision Engineering and Manufacturing*, vol. 15, no. 8, pp. 1583–1593, 2014.
- [113] M. Nazir, A. Majid-Mirza, and S. Ali-Khan, "PSO-GA based optimized feature selection using facial and clothing information for gender classification," *Journal of Applied Research and Technology*, vol. 12, no. 1, pp. 145–152, 2014.
- [114] K. Vidhya and K. R. S. Kumar, "Channel estimation of MIMO-OFDM system using PSO and GA," *Arabian Journal for Science and Engineering*, vol. 39, no. 5, pp. 4047–4056, 2014.
- [115] Y. Xiao, J. Xiao, F. Lu, and S. Wang, "Ensemble ANNs-PSO-GA approach for day-ahead stock E-exchange prices forecasting," *International Journal of Computational Intelligence Systems*, vol. 7, no. 2, pp. 272–290, 2014.
- [116] P. Ghamisi and J. A. Benediktsson, "Feature selection based on hybridization of genetic algorithm and particle swarm optimization," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 2, pp. 309–313, 2015.
- [117] H. Z. Tang, Y. Xiao, H. Huang, and X. Guo, "A novel dynamic particle swarm optimization algorithm based on improved artificial immune network," in *Proceedings of the 10th IEEE International Conference on Signal Processing (ICSP '10)*, B. Z. Yuan, Q. Q. Ruan, and X. F. Tang, Eds., pp. 103–106, New York, NY, USA, October 2010.
- [118] 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.
- [119] A. A. Ibrahim, A. Mohamed, H. Shareef, and S. P. Ghoshal, "Optimal power quality monitor placement in power systems based on particle swarm optimization and artificial immune system," in *Proceedings of the 3rd Conference on Data Mining and Optimization (DMO '11)*, pp. 141–145, June 2011.
- [120] R. J. Kuo, C. M. Chen, T. W. Liao, and F. C. Tien, "Hybrid of artificial immune system and particle swarm optimization-based support vector machine for radio frequency identification-based positioning system," *Computers and Industrial Engineering*, vol. 64, no. 1, pp. 333–341, 2013.
- [121] Z.-H. Liu, J. Zhang, S.-W. Zhou, X.-H. Li, and K. Liu, "Coevolutionary particle swarm optimization using AIS and its application in multiparameter estimation of PMSM," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1921–1935, 2013.
- [122] S. Darzi, T. S. Kiong, M. T. Islam, M. Ismail, S. Kibria, and B. Salem, "Null steering of adaptive beamforming using linear constraint minimum variance assisted by particle swarm optimization," *The Scientific World Journal*, vol. 2014, Article ID 724639, 10 pages, 2014.
- [123] Z. C. Li, D. J. Zheng, and H. J. Hou, "A hybrid particle swarm optimization algorithm based on nonlinear simplex method and tabu search," in *Advances in Neural Networks—ISNN 2010*, L. Q. Zhang, B. L. Lu, and J. Kwok, Eds., vol. 6063 of *Lecture Notes in Computer Science*, pp. 126–135, Springer, Berlin, Germany, 2010.
- [124] S. Nakano, A. Ishigame, and K. Yasuda, "Consideration of particle swarm optimization combined with tabu search," *Electrical Engineering in Japan*, vol. 172, no. 4, pp. 31–37, 2010.

- [125] T. Zhang, Y. J. Zhang, Q. P. Zheng, and P. M. Pardalos, "A hybrid particle swarm optimization and tabu search algorithm for order planning problems of steel factories based on the make-to-stock and make-to-order management architecture," *Journal of Industrial and Management Optimization*, vol. 7, no. 1, pp. 31–51, 2011.
- [126] R. Ktari and H. Chabchoub, "Essential particle swarm optimization queen with tabu search for MKP resolution," *Computing*, vol. 95, no. 9, pp. 897–921, 2013.
- [127] J. Wang, J. Lu, Z. Bie, S. You, and X. Cao, "Long-term maintenance scheduling of smart distribution system through a PSO-TS algorithm," *Journal of Applied Mathematics*, vol. 2014, Article ID 694086, 12 pages, 2014.
- [128] S.-M. Chen and C.-Y. Chien, "Solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques," *Expert Systems with Applications*, vol. 38, no. 12, pp. 14439–14450, 2011.
- [129] R. Xiao, W. Chen, and T. Chen, "Modeling of ant colony's labor division for the multi-project scheduling problem and its solution by pso," *Journal of Computational and Theoretical Nanoscience*, vol. 9, no. 2, pp. 223–232, 2012.
- [130] M. S. Kiran, E. Özceylan, M. Gündüz, and T. Paksoy, "A novel hybrid approach based on Particle Swarm Optimization and Ant Colony Algorithm to forecast energy demand of Turkey," *Energy Conversion and Management*, vol. 53, no. 1, pp. 75–83, 2012.
- [131] C.-L. Huang, W.-C. Huang, H.-Y. Chang, Y.-C. Yeh, and C.-Y. Tsai, "Hybridization strategies for continuous ant colony optimization and particle swarm optimization applied to data clustering," *Applied Soft Computing Journal*, vol. 13, no. 9, pp. 3864–3872, 2013.
- [132] R. Rahmani, R. Yusof, M. Seyedmahmoudian, and S. Mekhilef, "Hybrid technique of ant colony and particle swarm optimization for short term wind energy forecasting," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 123, pp. 163–170, 2013.
- [133] W. Elloumi, N. Baklouti, A. Abraham, and A. M. Alimi, "The multi-objective hybridization of particle swarm optimization and fuzzy ant colony optimization," *Journal of Intelligent and Fuzzy Systems*, vol. 27, no. 1, pp. 515–525, 2014.
- [134] S. M. Sait, A. T. Sheikh, and A. H. El-Maleh, "Cell assignment in hybrid CMOS/nanodevices architecture using a PSO/SA hybrid algorithm," *Journal of Applied Research and Technology*, vol. 11, no. 5, pp. 653–664, 2013.
- [135] H. Jiang and L. Zou, "A hybrid PSO-SA optimizing approach for SVM models in classification," *International Journal of Biomathematics*, vol. 6, no. 5, Article ID 1350036, 18 pages, 2013.
- [136] T. Niknam, M. R. Narimani, and M. Jabbari, "Dynamic optimal power flow using hybrid particle swarm optimization and simulated annealing," *International Transactions on Electrical Energy Systems*, vol. 23, no. 7, pp. 975–1001, 2013.
- [137] F. Khoshahval, A. Zolfaghari, H. Minuchehr, and M. R. Abbasi, "A new hybrid method for multi-objective fuel management optimization using parallel PSO-SA," *Progress in Nuclear Energy*, vol. 76, pp. 112–121, 2014.
- [138] H. Du, P. H. Lou, and W. H. Ye, "Application of hybrid particle swarm optimization in resource constrained multi-project scheduling," *Journal of the Chinese Society of Mechanical Engineers*, vol. 35, no. 5, pp. 371–379, 2014.
- [139] Y. Zhang, S. Wang, Y. Sun, G. Ji, P. Phillips, and Z. Dong, "Binary structuring elements decomposition based on an improved recursive dilation-union model and RSAPSO method," *Mathematical Problems in Engineering*, vol. 2014, Article ID 272496, 12 pages, 2014.
- [140] J. Geng, M.-W. Li, Z.-H. Dong, and Y.-S. Liao, "Port throughput forecasting by MARS-RSVR with chaotic simulated annealing particle swarm optimization algorithm," *Neurocomputing*, vol. 147, pp. 239–250, 2015.
- [141] 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, Cancún, Mexico, June 2013.
- [142] T. K. Sharma, M. Pant, and A. Abraham, "Blend of local and global variant of PSO in ABC," in *Proceedings of the World Congress on Nature and Biologically Inspired Computing (NaBIC '13)*, pp. 113–119, August 2013.
- [143] M. S. Kiran and M. Gündüz, "A recombination-based hybridization of particle swarm optimization and artificial bee colony algorithm for continuous optimization problems," *Applied Soft Computing Journal*, vol. 13, no. 4, pp. 2188–2203, 2013.
- [144] L. N. Vitorino, S. F. Ribeiro, and C. J. A. Bastos-Filho, "A mechanism based on artificial bee colony to generate diversity in particle swarm optimization," *Neurocomputing*, vol. 148, pp. 39–45, 2015.
- [145] G. Maione and A. Punzi, "Combining differential evolution and particle swarm optimization to tune and realize fractional-order controllers," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 19, no. 3, pp. 277–299, 2013.
- [146] Y. G. Fu, M. Y. Ding, C. P. Zhou, and H. P. Hu, "Route planning for unmanned aerial vehicle (UAV) on the sea using hybrid differential evolution and quantum-behaved particle swarm optimization," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 6, pp. 1451–1465, 2013.
- [147] Vasundhara, D. Mandal, R. Kar, and S. P. Ghoshal, "Digital FIR filter design using fitness based hybrid adaptive differential evolution with particle swarm optimization," *Natural Computing*, vol. 13, no. 1, pp. 55–64, 2014.
- [148] X. Yu, J. Cao, H. Shan, L. Zhu, and J. Guo, "An adaptive hybrid algorithm based on particle swarm optimization and differential evolution for global optimization," *The Scientific World Journal*, vol. 2014, Article ID 215472, 16 pages, 2014.
- [149] G.-G. Wang, A. Hossein Gandomi, X.-S. Yang, and A. Hossein Alavi, "A novel improved accelerated particle swarm optimization algorithm for global numerical optimization," *Engineering Computations*, vol. 31, no. 7, pp. 1198–1220, 2014.
- [150] A. Yadav and K. Deep, "An efficient co-swarm particle swarm optimization for non-linear constrained optimization," *Journal of Computational Science*, vol. 5, no. 2, pp. 258–268, 2014.
- [151] W. Xu, Z. Geng, Q. Zhu, and X. Gu, "Optimal grade transition for polyethylene reactors based on simultaneous strategies and trust region particle swarm optimization," *Industrial and Engineering Chemistry Research*, vol. 52, no. 9, pp. 3363–3372, 2013.
- [152] S. R. Mohanty, N. Kishor, and P. K. Ray, "Robust H-infinite loop shaping controller based on hybrid PSO and harmonic search for frequency regulation in hybrid distributed generation system," *International Journal of Electrical Power and Energy Systems*, vol. 60, pp. 302–316, 2014.
- [153] W. Guo, W. Li, Q. Zhang, L. Wang, Q. Wu, and H. Ren, "Biogeography-based particle swarm optimization with fuzzy

- elitism and its applications to constrained engineering problems,” *Engineering Optimization*, vol. 46, no. 11, pp. 1465–1484, 2014.
- [154] C. Y. Qiu, C. L. Wang, and X. Q. Zuo, “A novel multi-objective particle swarm optimization with K-means based global best selection strategy,” *International Journal of Computational Intelligence Systems*, vol. 6, no. 5, pp. 822–835, 2013.
- [155] W.-J. Chen, W.-C. Su, and Y.-L. Yang, “Application of constrained multi-objective hybrid quantum particle swarm optimization for improving performance of an ironless permanent magnet linear motor,” *Applied Mathematics & Information Sciences*, vol. 8, no. 6, pp. 3111–3120, 2014.
- [156] A. Ghanei, E. Assareh, M. Biglari, A. Ghanbarzadeh, and A. R. Noghrehabadi, “Thermal-economic multi-objective optimization of shell and tube heat exchanger using particle swarm optimization (PSO),” *Heat and Mass Transfer*, vol. 50, no. 10, pp. 1375–1384, 2014.
- [157] C. Duan, X. Wang, S. Shu, C. Jing, and H. Chang, “Thermodynamic design of Stirling engine using multi-objective particle swarm optimization algorithm,” *Energy Conversion and Management*, vol. 84, pp. 88–96, 2014.
- [158] M. R. Amiryousefi, M. Mohebbi, F. Khodaiyan, and M. G. Ahsaei, “Multi-objective optimization of deep-fat frying of ostrich meat plates using multi-objective particle swarm optimization (MOPSO),” *Journal of Food Processing and Preservation*, vol. 38, no. 4, pp. 1472–1479, 2014.
- [159] S. Ganguly, “Multi-objective planning for reactive power compensation of radial distribution networks with unified power quality conditioner allocation using particle swarm optimization,” *IEEE Transactions on Power Systems*, vol. 29, no. 4, pp. 1801–1810, 2014.
- [160] E. Z. Zhang, Y. F. Wu, and Q. W. Chen, “A practical approach for solving multi-objective reliability redundancy allocation problems using extended bare-bones particle swarm optimization,” *Reliability Engineering and System Safety*, vol. 127, pp. 65–76, 2014.
- [161] R. Perera, E. Sevillano, A. Arteaga, and A. de Diego, “Identification of intermediate debonding damage in FRP-plated RC beams based on multi-objective particle swarm optimization without updated baseline model,” *Composites Part B: Engineering*, vol. 62, pp. 205–217, 2014.
- [162] S. Cheng, M.-Y. Chen, and P. J. Fleming, “Improved multi-objective particle swarm optimization with preference strategy for optimal DG integration into the distribution system,” *Neurocomputing*, vol. 148, pp. 23–29, 2015.
- [163] M. Daneshyari and G. G. Yen, “Constrained multiple-swarm particle swarm optimization within a cultural framework,” *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 42, no. 2, pp. 475–490, 2012.
- [164] M. H. Afshar, “Extension of the constrained particle swarm optimization algorithm to optimal operation of multi-reservoirs system,” *International Journal of Electrical Power & Energy Systems*, vol. 51, pp. 71–81, 2013.
- [165] G. Koulinas, L. Kotsikas, and K. Anagnostopoulos, “A particle swarm optimization based hyper-heuristic algorithm for the classic resource constrained project scheduling problem,” *Information Sciences*, vol. 277, pp. 680–693, 2014.
- [166] J. J. Shan and Y. Ren, “Low-thrust trajectory design with constrained particle swarm optimization,” *Aerospace Science and Technology*, vol. 36, pp. 114–124, 2014.
- [167] C. C. Yeh and Y. C. Chien, “A particle swarm optimization-like algorithm for constrained minimal spanning tree problems,” *Journal of Marine Science and Technology*, vol. 22, no. 3, pp. 341–351, 2014.
- [168] N. Singh, R. Arya, and R. K. Agrawal, “A novel approach to combine features for salient object detection using constrained particle swarm optimization,” *Pattern Recognition*, vol. 47, no. 4, pp. 1731–1739, 2014.
- [169] P. Paliwal, N. P. Patidar, and R. K. Nema, “Determination of reliability constrained optimal resource mix for an autonomous hybrid power system using Particle Swarm Optimization,” *Renewable Energy*, vol. 63, pp. 194–204, 2014.
- [170] L. Z. Cui, Z. Ling, J. Poon et al., “A parallel model of independent component analysis constrained by a 5-parameter reference curve and its solution by multi-target particle swarm optimization,” *Analytical Methods*, vol. 6, no. 8, pp. 2679–2686, 2014.
- [171] Y. Y. Shou, Y. Li, and C. T. Lai, “Hybrid particle swarm optimization for preemptive resource-constrained project scheduling,” *Neurocomputing*, vol. 148, pp. 122–128, 2015.
- [172] M. Chen and S. A. Ludwig, “Discrete particle swarm optimization with local search strategy for rule classification,” in *Proceedings of the 4th World Congress on Nature and Biologically Inspired Computing (NaBIC '12)*, pp. 162–167, November 2012.
- [173] M. Shen, Z.-H. Zhan, W.-N. Chen, Y.-J. Gong, J. Zhang, and Y. Li, “Bi-velocity discrete particle swarm optimization and its application to multicast routing problem in communication networks,” *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 7141–7151, 2014.
- [174] C.-L. Chen, S.-Y. Huang, Y.-R. Tzeng, and C.-L. Chen, “A revised discrete particle swarm optimization algorithm for permutation flow-shop scheduling problem,” *Soft Computing*, vol. 18, no. 11, pp. 2271–2282, 2014.
- [175] Q. Cai, M. Gong, B. Shen, L. Ma, and L. Jiao, “Discrete particle swarm optimization for identifying community structures in signed social networks,” *Neural Networks*, vol. 58, pp. 4–13, 2014.
- [176] A. H. Kashan, B. Karimi, and A. Noktehdan, “A novel discrete particle swarm optimization algorithm for the manufacturing cell formation problem,” *The International Journal of Advanced Manufacturing Technology*, vol. 73, no. 9–12, pp. 1543–1556, 2014.
- [177] N. X. Xu, J. S. Gao, and X. G. Feng, “Study on the optimal design of frequency selective surfaces based on the discrete particle swarm optimization,” *Acta Physica Sinica*, vol. 63, no. 13, Article ID 138401, 2014.
- [178] R. Garg and A. K. Singh, “Multi-objective workflow grid scheduling using ϵ -fuzzy dominance sort based discrete particle swarm optimization,” *Journal of Supercomputing*, vol. 68, no. 2, pp. 709–732, 2013.
- [179] X. Zong, S. Xiong, and Z. Fang, “A conflict-congestion model for pedestrian-vehicle mixed evacuation based on discrete particle swarm optimization algorithm,” *Computers and Operations Research*, vol. 44, pp. 1–12, 2014.
- [180] R. Ezzeldin, B. Djebedjian, and T. Saafan, “Integer discrete particle swarm optimization of water distribution networks,” *Journal of Pipeline Systems Engineering and Practice*, vol. 5, no. 1, article 154, 2014.
- [181] T. T. Zhai and Z. F. He, “Instance selection for time series classification based on immune binary particle swarm optimization,” *Knowledge-Based Systems*, vol. 49, pp. 106–115, 2013.
- [182] K. N. V. D. Sarath and V. Ravi, “Association rule mining using binary particle swarm optimization,” *Engineering Applications of Artificial Intelligence*, vol. 26, no. 8, pp. 1832–1840, 2013.

- [183] M. A. Taha and D. I. Abu Al Nadi, "Spectrum sensing for cognitive radio using binary particle swarm optimization," *Wireless Personal Communications*, vol. 72, no. 4, pp. 2143–2153, 2013.
- [184] A. H. El-Maleh, A. T. Sheikh, and S. M. Sait, "Binary particle swarm optimization (BPSO) based state assignment for area minimization of sequential circuits," *Applied Soft Computing Journal*, vol. 13, no. 12, pp. 4832–4840, 2013.
- [185] A. Erturk, M. K. Gullu, D. Cesmeci, D. Gercek, and S. Erturk, "Spatial resolution enhancement of hyperspectral images using unmixing and binary particle swarm optimization," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 12, pp. 2100–2104, 2014.
- [186] Y. Zhang, S. Wang, P. Phillips, and G. Ji, "Binary PSO with mutation operator for feature selection using decision tree applied to spam detection," *Knowledge-Based Systems*, vol. 64, pp. 22–31, 2014.
- [187] H. T. Yin, P. Fu, and Z. Sun, "Face feature selection and recognition using separability criterion and binary particle swarm optimization algorithm," *Chinese Journal of Electronics*, vol. 23, no. 2, pp. 361–365, 2014.
- [188] J. Yang, H. Zhang, Y. Ling, C. Pan, and W. Sun, "Task allocation for wireless sensor network using modified binary particle swarm optimization," *IEEE Sensors Journal*, vol. 14, no. 3, pp. 882–892, 2014.
- [189] M. R. Ganesh, R. Krishna, K. Manikantan, and S. Ramachandran, "Entropy based Binary Particle Swarm Optimization and classification for ear detection," *Engineering Applications of Artificial Intelligence*, vol. 27, pp. 115–128, 2014.
- [190] A. R. Jordehi and J. Jasni, "Parameter selection in particle swarm optimisation: a survey," *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 25, no. 4, pp. 527–542, 2013.
- [191] S. Kumar and D. K. Chaturvedi, "Tuning of particle swarm optimization parameter using fuzzy logic," in *Proceedings of the International Conference on Communication Systems and Network Technologies (CSNT '11)*, pp. 174–179, June 2011.
- [192] W. Zhang, Y. Jin, X. Li, and X. Zhang, "A simple way for parameter selection of standard particle swarm optimization," in *Artificial Intelligence and Computational Intelligence Part III*, H. Deng, D. Miao, J. Lei, and F. L. Wang, Eds., pp. 436–443, Springer, Berlin, Germany, 2011.
- [193] H. Yang, "Particle swarm optimization with modified velocity strategy," in *Proceedings of the International Conference on Energy and Environmental Science (ICEES '11)*, X. Zhou, Ed., pp. 1074–1079, Elsevier Science, 2011.
- [194] J. Sun, W. Fang, X. Wu, V. Palade, and W. Xu, "Quantum-behaved particle swarm optimization: analysis of individual particle behavior and parameter selection," *Evolutionary Computation*, vol. 20, no. 3, pp. 349–393, 2012.
- [195] I. M. Yassin, M. N. Taib, R. Adnan, M. K. M. Salleh, and M. K. Hamzah, "Effect of swarm size parameter on Binary Particle Swarm optimization-based NARX structure selection," in *Proceedings of the IEEE Symposium on Industrial Electronics and Applications (ISIEA '12)*, pp. 219–223, Bandung City West Java, Indonesia, September 2012.
- [196] D.-S. Wang, H.-J. Wang, and J.-K. Zhang, "Selection of the PSO parameters for inverting of ellipsometry," in *Proceedings of the International Conference on Industrial Control and Electronics Engineering (ICICEE 2012)*, pp. 776–780, August 2012.
- [197] H. Hao, N. Hu, X. Xu, and W. Q. Ying, "Parameters selection and optimization of particle swarm optimization algorithm based on molecular force model," in *Measurement Technology and Engineering Researches in Industry*, P. Yarlagadda and Y. H. Kim, Eds., Parts 1–3, pp. 1370–1373, Trans Tech Publications, Stafa, Switzerland, 2013.
- [198] G. Xu, "An adaptive parameter tuning of particle swarm optimization algorithm," *Applied Mathematics and Computation*, vol. 219, no. 9, pp. 4560–4569, 2013.
- [199] P. Chauhan, K. Deep, and M. Pant, "Novel inertia weight strategies for particle swarm optimization," *Memetic Computing*, vol. 5, no. 3, pp. 229–251, 2013.
- [200] W. Zhang, D. Ma, J.-J. Wei, and H.-F. Liang, "A parameter selection strategy for particle swarm optimization based on particle positions," *Expert Systems with Applications*, vol. 41, no. 7, pp. 3576–3584, 2014.
- [201] M. Kanemasa and E. Aiyoshi, "Algorithm tuners for PSO methods and genetic programming techniques for learning tuning rules," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 9, no. 4, pp. 407–414, 2014.
- [202] K. Wang and J. H. Shen, "The convergence basis of particle swarm optimization," in *Proceedings of the International Conference on Industrial Control and Electronics Engineering (ICICEE '12)*, pp. 63–66, August 2012.
- [203] J. Sun, X. Wu, V. Palade, W. Fang, C.-H. Lai, and W. Xu, "Convergence analysis and improvements of quantum-behaved particle swarm optimization," *Information Sciences*, vol. 193, pp. 81–103, 2012.
- [204] T. Kurihara and K. Jin'no, "Analysis of convergence property of PSO and its application to nonlinear blind source separation," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '13)*, pp. 976–981, IEEE, Cancun, Mexico, June 2013.
- [205] C.-C. Lin, "Dynamic router node placement in wireless mesh networks: a PSO approach with constriction coefficient and its convergence analysis," *Information Sciences*, vol. 232, pp. 294–308, 2013.
- [206] Y. Zhang, D.-W. Gong, X.-Y. Sun, and N. Geng, "Adaptive bare-bones particle swarm optimization algorithm and its convergence analysis," *Soft Computing*, vol. 18, no. 7, pp. 1337–1352, 2014.
- [207] W. T. Lin, Z. Lian, X. Gu, and B. Jiao, "A local and global search combined particle swarm optimization algorithm and its convergence analysis," *Mathematical Problems in Engineering*, vol. 2014, Article ID 905712, 11 pages, 2014.
- [208] S. Kim and L. Li, "Statistical identifiability and convergence evaluation for nonlinear pharmacokinetic models with particle swarm optimization," *Computer Methods and Programs in Biomedicine*, vol. 113, no. 2, pp. 413–432, 2014.
- [209] M. Waintraub, R. Schirru, and C. M. N. A. Pereira, "Multi-processor modeling of parallel Particle Swarm Optimization applied to nuclear engineering problems," *Progress in Nuclear Energy*, vol. 51, no. 6, pp. 680–688, 2009.
- [210] S.-C. Yu, "Elucidating multiprocessors flow shop scheduling with dependent setup times using a twin particle swarm optimization," *Applied Soft Computing Journal*, vol. 21, pp. 578–589, 2014.
- [211] Y. K. Hung and W. C. Wang, "Accelerating parallel particle swarm optimization via GPU," *Optimization Methods and Software*, vol. 27, no. 1, pp. 33–51, 2012.
- [212] B. Rymut, B. Kwolek, and T. Krzeszowski, "GPU-accelerated human motion tracking using particle filter combined with PSO," in *Advanced Concepts for Intelligent Vision Systems, ACIVS 2013*, J. Blanc-Talon, A. Kasinski, W. Philips, D. Popescu,

- and P. Scheunders, Eds., pp. 426–437, Springer, Berlin, Germany, 2013.
- [213] J. Kumar, L. Singh, and S. Paul, “GPU based parallel cooperative particle swarm optimization using C-CUDA: a case study,” in *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE '13)*, July 2013.
- [214] O. Awwad, A. Al-Fuqaha, G. Ben Brahim, B. Khan, and A. Rayes, “Distributed topology control in large-scale hybrid RF/FSO networks: SIMT GPU-based particle swarm optimization approach,” *International Journal of Communication Systems*, vol. 26, no. 7, pp. 888–911, 2013.
- [215] R.-B. Chen, Y.-W. Hsu, Y. Hung, and W. Wang, “Discrete particle swarm optimization for constructing uniform design on irregular regions,” *Computational Statistics & Data Analysis*, vol. 72, pp. 282–297, 2014.
- [216] J. Liu, X. G. Luo, and X. M. F. Zhang, “Job scheduling algorithm for cloud computing based on particle swarm optimization,” in *Nanotechnology and Precision Engineering, Pts 1 and 2*, Z. Y. Jiang and Y. H. Kim, Eds., pp. 957–960, Trans Tech Publications, Stäfa, Switzerland, 2013.
- [217] Y. J. Xu and T. You, “Minimizing thermal residual stresses in ceramic matrix composites by using Iterative Map Reduce guided particle swarm optimization algorithm,” *Composite Structures*, vol. 99, pp. 388–396, 2013.
- [218] F. Ramezani, J. Lu, and F. Hussain, “Task scheduling optimization in cloud computing applying multi-objective particle swarm optimization,” in *Service-Oriented Computing*, S. Basu, C. Pautasso, L. Zhang, and X. Fu, Eds., vol. 8274 of *Lecture Notes in Computer Science*, pp. 237–251, Springer, Berlin, Germany, 2013.
- [219] K. Govindarajan, T. S. Somasundaram, and V. S. Kumar, “Particle swarm optimization (PSO)-based clustering for improving the quality of learning using cloud computing,” in *Proceedings of the IEEE 13th International Conference on Advanced Learning Technologies (ICALT '13)*, pp. 495–497, IEEE, Beijing, China, July 2013.
- [220] F. Ramezani, J. Lu, and F. K. Hussain, “Task-based system load balancing in cloud computing using particle swarm optimization,” *International Journal of Parallel Programming*, vol. 42, no. 5, pp. 739–754, 2014.
- [221] S. Ganguly, N. C. Sahoo, and D. Das, “Multi-objective particle swarm optimization based on fuzzy-Pareto-dominance for possibilistic planning of electrical distribution systems incorporating distributed generation,” *Fuzzy Sets and Systems*, vol. 213, pp. 47–73, 2013.
- [222] S. Komsiyah, “Computational methods of Gaussian particle swarm optimization (GPSO) and lagrange multiplier on economic dispatch issues (case study on electrical system of Java-Bali IV area),” in *Proceedings of the International Conference on Advances Science and Contemporary Engineering (ICASCE '13)*, vol. 68, 2014.
- [223] J. W. Feng, F. Tian, P. Jia, Q. He, Y. Shen, and S. Fan, “Improving the performance of electronic nose for wound infection detection using orthogonal signal correction and particle swarm optimization,” *Sensor Review*, vol. 34, no. 4, pp. 389–395, 2014.
- [224] E. Pekşen, T. Yas, and A. Kiyak, “1-D DC resistivity modeling and interpretation in anisotropic media using particle swarm optimization,” *Pure and Applied Geophysics*, vol. 171, no. 9, pp. 2371–2389, 2014.
- [225] J. Yang, L. F. He, and S. Y. Fu, “An improved PSO-based charging strategy of electric vehicles in electrical distribution grid,” *Applied Energy*, vol. 128, pp. 82–92, 2014.
- [226] I. M. de Mendonça, I. C. S. Junior, and A. L. M. Marcato, “Static planning of the expansion of electrical energy transmission systems using particle swarm optimization,” *International Journal of Electrical Power and Energy Systems*, vol. 60, pp. 234–244, 2014.
- [227] T.-D. Liu, T.-E. Fan, G.-F. Shao, J.-W. Zheng, and Y.-H. Wen, “Particle swarm optimization of the stable structure of tetrahedral Pt-based bimetallic nanoparticles,” *Physics Letters A*, vol. 378, no. 40, pp. 2965–2972, 2014.
- [228] U. Aich and S. Banerjee, “Modeling of EDM responses by support vector machine regression with parameters selected by particle swarm optimization,” *Applied Mathematical Modelling*, vol. 38, no. 11-12, pp. 2800–2818, 2014.
- [229] C.-J. Chou, C.-Y. Lee, and C.-C. Chen, “Survey of reservoir grounding system defects considering the performance of lightning protection and improved design based on soil drilling data and the particle swarm optimization technique,” *IEEE Transactions on Electrical and Electronic Engineering*, vol. 9, no. 6, pp. 605–613, 2014.
- [230] B. A. Lee, B. S. Kim, M. S. Ko, K. Y. Kim, and S. Kim, “Electrical resistance imaging of two-phase flow with a mesh grouping technique based on particle swarm optimization,” *Nuclear Engineering and Technology*, vol. 46, no. 1, pp. 109–116, 2014.
- [231] P. Thakral and A. K. Bakhshi, “Computational atomistic blueprinting of novel conducting copolymers using particle swarm optimization,” *Journal of Computer-Aided Molecular Design*, vol. 28, no. 2, pp. 111–122, 2014.
- [232] I. Fister, X.-S. Yang, K. Ljubič, D. Fister, and J. Brest, “Towards the novel reasoning among particles in PSO by the use of RDF and SPARQL,” *The Scientific World Journal*, vol. 2014, Article ID 121782, 10 pages, 2014.
- [233] J. Aghaei, K. M. Muttaqi, A. Azizvahed, and M. Gitizadeh, “Distribution expansion planning considering reliability and security of energy using modified PSO (Particle Swarm Optimization) algorithm,” *Energy*, vol. 65, pp. 398–411, 2014.
- [234] A. Selakov, D. Cvijetinović, L. Milović, S. Mellon, and D. Bekut, “Hybrid PSO-SVM method for short-term load forecasting during periods with significant temperature variations in city of Burbank,” *Applied Soft Computing Journal*, vol. 16, pp. 80–88, 2014.
- [235] Y. Shirvany, Q. Mahmood, F. Edelvik, S. Jakobsson, A. Hedstrom, and M. Persson, “Particle swarm optimization applied to EEG source localization of somatosensory evoked potentials,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 22, no. 1, pp. 11–20, 2014.
- [236] D. H. Tungadio, B. P. Numbi, M. W. Siti, and A. A. Jimoh, “Particle swarm optimization for power system state estimation,” *Neurocomputing*, vol. 148, pp. 175–180, 2015.
- [237] Y. F. Cai and S. X. Yang, “An improved PSO-based approach with dynamic parameter tuning for cooperative multi-robot target searching in complex unknown environments,” *International Journal of Control*, vol. 86, no. 10, pp. 1720–1732, 2013.
- [238] K. Kolomvatsos and S. Hadjieftymiades, “On the use of particle swarm optimization and Kernel density estimator in concurrent negotiations,” *Information Sciences*, vol. 262, pp. 99–116, 2014.
- [239] S. K. Pandey, S. R. Mohanty, N. Kishor, and J. P. S. Catalão, “Frequency regulation in hybrid power systems using particle swarm optimization and linear matrix inequalities based robust controller design,” *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 887–900, 2014.

- [240] G. Štimac, S. Braut, and R. Žigulić, “Comparative analysis of PSO algorithms for PID controller tuning,” *Chinese Journal of Mechanical Engineering*, vol. 27, no. 5, pp. 928–936, 2014.
- [241] N. Nedic, D. Prsic, L. Dubonjic, V. Stojanovic, and V. Djordjevic, “Optimal cascade hydraulic control for a parallel robot platform by PSO,” *International Journal of Advanced Manufacturing Technology*, vol. 72, no. 5–8, pp. 1085–1098, 2014.
- [242] W.-D. Chang and C.-Y. Chen, “PID controller design for MIMO processes using improved particle swarm optimization,” *Circuits, Systems, and Signal Processing*, vol. 33, no. 5, pp. 1473–1490, 2014.
- [243] X. Xiang, R. Mutlu, G. Alici, and W. Li, “Control of conducting polymer actuators without physical feedback: simulated feedback control approach with particle swarm optimization,” *Smart Materials and Structures*, vol. 23, no. 3, Article ID 035014, 2014.
- [244] K. A. Danapalasingam, “Robust autonomous helicopter stabilizer tuned by particle swarm optimization,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 28, no. 1, Article ID 1459002, 2014.
- [245] M. J. Mahmoodabadi, M. Taherkhorsandi, and A. Bagheri, “Optimal robust sliding mode tracking control of a biped robot based on ingenious multi-objective PSO,” *Neurocomputing*, vol. 124, pp. 194–209, 2014.
- [246] Y. Zhong, X. Huang, P. Meng, and F. Li, “PSO-RBF neural network PID control algorithm of electric gas pressure regulator,” *Abstract and Applied Analysis*, vol. 2014, Article ID 731368, 7 pages, 2014.
- [247] J.-W. Perng, G.-Y. Chen, and S.-C. Hsieh, “Optimal pid controller design based on PSO-RBFNN for wind turbine systems,” *Energies*, vol. 7, no. 1, pp. 191–209, 2014.
- [248] Y.-C. Huang and Y.-H. Li, “Experiments of iterative learning control system using particle swarm optimization by new bounded constraints on velocity and positioning,” *Engineering Computations*, vol. 31, no. 2, pp. 250–266, 2014.
- [249] M. G. Nisha and G. N. Pillai, “Nonlinear model predictive control of MIMO system with relevance vector machines and particle swarm optimization,” *Control Engineering and Applied Informatics*, vol. 16, no. 2, pp. 58–66, 2014.
- [250] M. Yousefi, M. Mosalanejad, G. Moradi, and A. Abdipour, “Dual band planar hybrid coupler with enhanced bandwidth using particle swarm optimization technique,” *IEICE Electronics Express*, vol. 9, no. 12, pp. 1030–1035, 2012.
- [251] X.-B. Sun, Z.-M. Li, C.-L. Zhao, and Z. Zhou, “Cognitive UWB pulse waveform design based on particle swarm optimization,” *Ad-Hoc & Sensor Wireless Networks*, vol. 16, no. 1-3, pp. 215–227, 2012.
- [252] H. Yongqiang, L. Wentao, and L. Xiaohui, “Particle swarm optimization for antenna selection in MIMO system,” *Wireless Personal Communications*, vol. 68, no. 3, pp. 1013–1029, 2013.
- [253] C.-C. Chiu, M.-H. Ho, and S.-H. Liao, “PSO and APSO for optimizing coverage in indoor UWB communication system,” *International Journal of RF and Microwave Computer-Aided Engineering*, vol. 23, no. 3, pp. 300–308, 2013.
- [254] A. A. Minasian and T. S. Bird, “Particle swarm optimization of microstrip antennas for wireless communication systems,” *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 12, pp. 6214–6217, 2013.
- [255] M. Zubair and M. Moinuddin, “Joint optimization of microstrip patch antennas using particle swarm optimization for UWB systems,” *International Journal of Antennas and Propagation*, vol. 2013, Article ID 649049, 8 pages, 2013.
- [256] Y. G. Kim and M. J. Lee, “Scheduling multi-channel and multi-timeslot in time constrained wireless sensor networks via simulated annealing and particle swarm optimization,” *IEEE Communications Magazine*, vol. 52, no. 1, pp. 122–129, 2014.
- [257] A. Yazgan and I. Hakki Cavdar, “A comparative study between LMS and PSO algorithms on the optical channel estimation for radio over fiber systems,” *Optik*, vol. 125, no. 11, pp. 2582–2586, 2014.
- [258] R. I. Rabady and A. Ababneh, “Global optimal design of optical multilayer thin-film filters using particle swarm optimization,” *Optik*, vol. 125, no. 1, pp. 548–553, 2014.
- [259] G. Das, P. K. Pattnaik, and S. K. Padhy, “Artificial Neural Network trained by Particle Swarm Optimization for non-linear channel equalization,” *Expert Systems with Applications*, vol. 41, no. 7, pp. 3491–3496, 2014.
- [260] S. Scott-Hayward and E. Garcia-Palacios, “Channel time allocation PSO for gigabit multimedia wireless networks,” *IEEE Transactions on Multimedia*, vol. 16, no. 3, pp. 828–836, 2014.
- [261] A. Omidvar and K. Mohammadi, “Particle swarm optimization in intelligent routing of delay-tolerant network routing,” *EURASIP Journal on Wireless Communications and Networking*, vol. 2014, article 147, 8 pages, 2014.
- [262] P. Kuila and P. K. Jana, “Energy efficient clustering and routing algorithms for wireless sensor networks: particle swarm optimization approach,” *Engineering Applications of Artificial Intelligence*, vol. 33, pp. 127–140, 2014.
- [263] Z. H. Liu and X. L. Wang, “A PSO-based algorithm for load balancing in virtual machines of cloud computing environment,” in *Advances in Swarm Intelligence: Third International Conference, ICSI 2012, Shenzhen, China, June 17–20, 2012 Proceedings, Part I*, Y. Tan, Y. Shi, and Z. Ji, Eds., vol. 7331 of *Lecture Notes in Computer Science*, pp. 142–147, Springer, Berlin, Germany, 2012.
- [264] Z. H. Che, T.-A. Chiang, and Z.-G. Che, “Using analytic network process and turbo particle swarm optimization algorithm for non-balanced supply chain planning considering supplier relationship management,” *Transactions of the Institute of Measurement and Control*, vol. 34, no. 6, pp. 720–735, 2012.
- [265] V. Hajipour and S. H. R. Pasandideh, “Proposing an adaptive particle swarm optimization for a novel Bi-objective queuing facility location model,” *Economic Computation and Economic Cybernetics Studies and Research*, vol. 46, no. 3, pp. 223–240, 2012.
- [266] F. J. Cabrerizo, E. Herrera-Viedma, and W. Pedrycz, “A method based on PSO and granular computing of linguistic information to solve group decision making problems defined in heterogeneous contexts,” *European Journal of Operational Research*, vol. 230, no. 3, pp. 624–633, 2013.
- [267] K. Prescilla and A. I. Selvakumar, “Modified Binary particle swarm optimization algorithm application to real-time task assignment in heterogeneous multiprocessor,” *Microprocessors and Microsystems*, vol. 37, no. 6-7, pp. 583–589, 2013.
- [268] H. B. Duan, X. X. Wei, and Z. N. Dong, “Multiple UCAVs cooperative air combat simulation platform based on PSO, ACO, and game theory,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 28, no. 11, pp. 12–19, 2013.
- [269] F. Belmecheri, C. Prins, F. Yalaoui, and L. Amodeo, “Particle swarm optimization algorithm for a vehicle routing problem with heterogeneous fleet, mixed backhauls, and time windows,” *Journal of Intelligent Manufacturing*, vol. 24, no. 4, pp. 775–789, 2013.
- [270] W. B. Hu, H. Liang, C. Peng, B. Du, and Q. Hu, “A hybrid chaos-particle swarm optimization algorithm for the vehicle routing

- problem with time window," *Entropy*, vol. 15, no. 4, pp. 1247–1270, 2013.
- [271] A. Al Badawi and A. Shatnawi, "Static scheduling of directed acyclic data flow graphs onto multiprocessors using particle swarm optimization," *Computers and Operations Research*, vol. 40, no. 10, pp. 2322–2328, 2013.
- [272] F. P. Goksal, I. Karaoglan, and F. Altiparmak, "A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery," *Computers & Industrial Engineering*, vol. 65, no. 1, pp. 39–53, 2013.
- [273] W. Zhang, H. Xie, B. Cao, and A. M. K. Cheng, "Energy-aware real-time task scheduling for heterogeneous multiprocessors with particle swarm optimization algorithm," *Mathematical Problems in Engineering*, vol. 2014, Article ID 287475, 9 pages, 2014.
- [274] P. N. Kechagiopoulos and G. N. Beligiannis, "Solving the urban transit routing problem using a particle swarm optimization based algorithm," *Applied Soft Computing Journal*, vol. 21, pp. 654–676, 2014.
- [275] L. Ming, H. Hai, Z. Aimin, S. Yingde, L. Zhao, and Z. Xingguo, "Modeling of mechanical properties of as-cast Mg-Li-Al alloys based on PSO-BP algorithm," *China Foundry*, vol. 9, no. 2, pp. 119–124, 2012.
- [276] J. Q. Chen, Y. Tang, R. Ge, Q. An, and X. 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.
- [277] 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.
- [278] J. Chen, Y. Tang, and X. Huang, "Application of surrogate based particle swarm optimization to the reliability-based robust design of composite pressure vessels," *Acta Mechanica Solida Sinica*, vol. 26, no. 5, pp. 480–490, 2013.
- [279] Y. L. Zhang, D. Gallipoli, and C. Augarde, "Parameter identification for elasto-plastic modelling of unsaturated soils from pressuremeter tests by parallel modified particle swarm optimization," *Computers and Geotechnics*, vol. 48, pp. 293–303, 2013.
- [280] G. H. Wang, J. Chen, T. Cai, and B. Xin, "Decomposition-based multi-objective differential evolution particle swarm optimization for the design of a tubular permanent magnet linear synchronous motor," *Engineering Optimization*, vol. 45, no. 9, pp. 1107–1127, 2013.
- [281] T. Lazrag, M. Kacem, P. Dubujet, J. Sghaier, and A. Bellagi, "Determination of unsaturated hydraulic properties using drainage gravity test and particle swarm optimization algorithm," *Journal of Porous Media*, vol. 16, no. 11, pp. 1025–1034, 2013.
- [282] C.-H. Lee, K.-S. Shih, C.-C. Hsu, and T. Cho, "Simulation-based particle swarm optimization and mechanical validation of screw position and number for the fixation stability of a femoral locking compression plate," *Medical Engineering & Physics*, vol. 36, no. 1, pp. 57–64, 2014.
- [283] J. J. Lake, A. E. Duwel, and R. N. Candler, "Particle swarm optimization for design of slotted MEMS resonators with low thermoelastic dissipation," *Journal of Microelectromechanical Systems*, vol. 23, no. 2, pp. 364–371, 2014.
- [284] A. R. Vosoughi and S. Gerist, "New hybrid FE-PSO-CGAs sensitivity base technique for damage detection of laminated composite beams," *Composite Structures*, vol. 118, pp. 68–73, 2014.
- [285] S. L. M. Ribeiro, T. A. A. Silva, L. M. G. Vieira, T. H. Panzera, K. Boba, and F. Scarpa, "Geometric effects of sustainable auxetic structures integrating the particle swarm optimization and finite element method," *Materials Research*, vol. 17, no. 3, pp. 747–757, 2014.
- [286] P. Kitak, A. Glotic, and I. Ticar, "Heat transfer coefficients determination of numerical model by using particle swarm optimization," *IEEE Transactions on Magnetics*, vol. 50, no. 2, 2014.
- [287] R. Kalatehjari, A. S. A. Rashid, N. Ali, and M. Hajihassani, "The contribution of particle swarm optimization to three-dimensional slope stability analysis," *The Scientific World Journal*, vol. 2014, Article ID 973093, 12 pages, 2014.
- [288] B. Ashuri and M. Tavakolan, "Fuzzy enabled hybrid genetic algorithm-particle swarm optimization approach to solve TCRO problems in construction project planning," *Journal of Construction Engineering and Management*, vol. 138, no. 9, pp. 1065–1074, 2012.
- [289] A. Bozorgi-Amiri, M. S. Jabalameli, M. Alinaghian, and M. Heydari, "A modified particle swarm optimization for disaster relief logistics under uncertain environment," *International Journal of Advanced Manufacturing Technology*, vol. 60, no. 1–4, pp. 357–371, 2012.
- [290] J. Sadoghi Yazdi, F. Kalantary, and H. Sadoghi Yazdi, "Calibration of soil model parameters using particle swarm optimization," *International Journal of Geomechanics*, vol. 12, no. 3, pp. 229–238, 2012.
- [291] B. Bolat, O. Altun, and P. Cortés, "A particle swarm optimization algorithm for optimal car-call allocation in elevator group control systems," *Applied Soft Computing Journal*, vol. 13, no. 5, pp. 2633–2642, 2013.
- [292] K. S. J. Babu and D. P. Vijayalakshmi, "Self-adaptive PSO-GA hybrid model for combinatorial water distribution network design," *Journal of Pipeline Systems Engineering and Practice*, vol. 4, no. 1, pp. 57–67, 2013.
- [293] S. C. Mohan, A. Yadav, D. Kumar Maiti, and D. Maity, "A comparative study on crack identification of structures from the changes in natural frequencies using GA and PSO," *Engineering Computations*, vol. 31, no. 7, pp. 1514–1531, 2014.
- [294] M. Asadnia, L. H. C. Chua, X. S. Qin, and A. Talei, "Improved particle swarm optimization-based artificial neural network for Rainfall-Runoff modeling," *Journal of Hydrologic Engineering*, vol. 19, no. 7, pp. 1320–1329, 2014.
- [295] B. Yan, S. Goto, A. Miyamoto, and H. Zhao, "Imaging-based rating for corrosion states of weathering steel using wavelet transform and PSO-SVM techniques," *Journal of Computing in Civil Engineering*, vol. 28, no. 3, Article ID 04014008, 2014.
- [296] Y. Wang and L. Li, "A PSO algorithm for constrained redundancy allocation in multi-state systems with bridge topology," *Computers & Industrial Engineering*, vol. 68, no. 1, pp. 13–22, 2014.
- [297] J. Sadeghi, S. Sadeghi, and S. T. Niaki, "Optimizing a hybrid vendor-managed inventory and transportation problem with fuzzy demand: an improved particle swarm optimization algorithm," *Information Sciences*, vol. 272, pp. 126–144, 2014.
- [298] Ž. Kanović, V. Bugarski, and T. Bačkalić, "Ship lock control system optimization using GA, PSO and ABC: a comparative review," *Promet—Traffic & Transportation*, vol. 26, no. 1, pp. 23–31, 2014.

- [299] P. Mandal, H. Zareipour, and W. D. Rosehart, "Forecasting aggregated wind power production of multiple wind farms using hybrid wavelet-PSO-NNs," *International Journal of Energy Research*, vol. 38, no. 13, pp. 1654–1666, 2014.
- [300] K. H. Chao, "An extension theory-based maximum power tracker using a particle swarm optimization algorithm," *Energy Conversion and Management*, vol. 86, pp. 435–442, 2014.
- [301] G. G. Chen, L. Liu, P. Song, and Y. Du, "Chaotic improved PSO-based multi-objective optimization for minimization of power losses and L index in power systems," *Energy Conversion and Management*, vol. 86, pp. 548–560, 2014.
- [302] C. Hu, G. Jain, P. Zhang, C. Schmidt, P. Gomadam, and T. Gorka, "Data-driven method based on particle swarm optimization and k-nearest neighbor regression for estimating capacity of lithium-ion battery," *Applied Energy*, vol. 129, pp. 49–55, 2014.
- [303] I. Tabet, K. Touafek, N. Bellel, N. Bouarroudj, A. Khelifa, and M. Adouane, "Optimization of angle of inclination of the hybrid photovoltaic-thermal solar collector using particle swarm optimization algorithm," *Journal of Renewable and Sustainable Energy*, vol. 6, no. 5, Article ID 053116, 2014.
- [304] S. Bahrami, R. A. Hooshmand, and M. Parastegari, "Short term electric load forecasting by wavelet transform and grey model improved by PSO (particle swarm optimization) algorithm," *Energy*, vol. 72, pp. 434–442, 2014.
- [305] A. Askarzadeh, "Comparison of particle swarm optimization and other metaheuristics on electricity demand estimation: a case study of Iran," *Energy*, vol. 72, pp. 484–491, 2014.
- [306] M. Sharafi and T. Y. ElMekkawy, "Multi-objective optimal design of hybrid renewable energy systems using PSO-simulation based approach," *Renewable Energy*, vol. 68, pp. 67–79, 2014.
- [307] P. García-Triviño, F. Llorens-Iborra, C. A. García-Vázquez, A. J. Gil-Mena, L. M. Fernández-Ramírez, and F. Jurado, "Long-term optimization based on PSO of a grid-connected renewable energy/battery/hydrogen hybrid system," *International Journal of Hydrogen Energy*, vol. 39, no. 21, pp. 10805–10816, 2014.
- [308] S. Biao, H. Chang hua, Y. Xin hua, and H. Chuan, "Mutation particle swarm optimization algorithm for solving the optimal operation model of thermal power plants," *Journal of Renewable and Sustainable Energy*, vol. 6, no. 4, Article ID 043118, 2014.
- [309] J.-H. Chen, H.-T. Yau, and W. Hung, "Design and study on sliding mode extremum seeking control of the chaos embedded particle swarm optimization for maximum power point tracking in wind power systems," *Energies*, vol. 7, no. 3, pp. 1706–1720, 2014.
- [310] M. Seyedmahmoudian, S. Mekhilef, R. Rahmani, R. Yusof, and A. Asghar Shojaei, "Maximum power point tracking of partial shaded photovoltaic array using an evolutionary algorithm: a particle swarm optimization technique," *Journal of Renewable and Sustainable Energy*, vol. 6, no. 2, Article ID 023102, 2014.
- [311] M. M. Aman, G. B. Jasmon, A. H. A. Bakar, and H. Mokhlis, "A new approach for optimum simultaneous multi-DG distributed generation Units placement and sizing based on maximization of system loadability using HPSO (hybrid particle swarm optimization) algorithm," *Energy*, vol. 66, pp. 202–215, 2014.
- [312] C.-L. Xiao and H.-X. Huang, "Optimal design of heating system for rapid thermal cycling mold using particle swarm optimization and finite element method," *Applied Thermal Engineering*, vol. 64, no. 1-2, pp. 462–470, 2014.
- [313] K. L. Lian, J. H. Jhang, and I. S. Tian, "A maximum power point tracking method based on perturb-and-observe combined with particle swarm optimization," *IEEE Journal of Photovoltaics*, vol. 4, no. 2, pp. 626–633, 2014.
- [314] S. N. Qasem and S. M. Shamsuddin, "Radial basis function network based on time variant multi-objective particle swarm optimization for medical diseases diagnosis," *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 1427–1438, 2011.
- [315] Y. Zhang, S. Wang, and L. Wu, "A novel method for magnetic resonance brain image classification based on adaptive chaotic PSO," *Progress in Electromagnetics Research*, vol. 109, pp. 325–343, 2010.
- [316] X. D. Guo, C. Wang, and R. G. Yan, "An electromagnetic localization method for medical micro-devices based on adaptive particle swarm optimization with neighborhood search," *Measurement*, vol. 44, no. 5, pp. 852–858, 2011.
- [317] P.-C. Chang, J.-J. Lin, and C.-H. Liu, "An attribute weight assignment and particle swarm optimization algorithm for medical database classifications," *Computer Methods and Programs in Biomedicine*, vol. 107, no. 3, pp. 382–392, 2012.
- [318] L.-F. Chen, C.-T. Su, K.-H. Chen, and P.-C. Wang, "Particle swarm optimization for feature selection with application in obstructive sleep apnea diagnosis," *Neural Computing & Applications*, vol. 21, no. 8, pp. 2087–2096, 2012.
- [319] W.-T. Sung and Y.-C. Chiang, "Improved particle swarm optimization algorithm for android medical care iot using modified parameters," *Journal of Medical Systems*, vol. 36, no. 6, pp. 3755–3763, 2012.
- [320] I. Cruz-Aceves, J. G. Aviña-Cervantes, J. M. López-Hernández, and S. E. González-Reyna, "Multiple active contours driven by particle swarm optimization for cardiac medical image segmentation," *Computational and Mathematical Methods in Medicine*, vol. 2013, Article ID 132953, 13 pages, 2013.
- [321] A. Subasi, "Classification of EMG signals using PSO optimized SVM for diagnosis of neuromuscular disorders," *Computers in Biology and Medicine*, vol. 43, no. 5, pp. 576–586, 2013.
- [322] Y. Zhang, S. Wang, G. Ji, and Z. Dong, "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.
- [323] V. Mangat and R. Vig, "Novel associative classifier based on dynamic adaptive PSO: application to determining candidates for thoracic surgery," *Expert Systems with Applications*, vol. 41, no. 18, pp. 8234–8244, 2014.
- [324] D. Mandal, A. Chatterjee, and M. Maitra, "Robust medical image segmentation using particle swarm optimization aided level set based global fitting energy active contour approach," *Engineering Applications of Artificial Intelligence*, vol. 35, pp. 199–214, 2014.
- [325] Y.-Z. Hsieh, M.-C. Su, and P.-C. Wang, "A PSO-based rule extractor for medical diagnosis," *Journal of Biomedical Informatics*, vol. 49, pp. 53–60, 2014.
- [326] S. Ganapathy, R. Sethukkarasi, P. Yogesh, P. Vijayakumar, and A. Kannan, "An intelligent temporal pattern classification system using fuzzy temporal rules and particle swarm optimization," *Academy Proceedings in Engineering Sciences*, vol. 39, no. 2, pp. 283–302, 2014.
- [327] P. Thakral, V. Arora, S. Kukreti, and A. K. Bakhshi, "Insilico engineering of intrinsically conducting copolymers using particle swarm optimization algorithm," *Indian Journal of Chemistry—Section A Inorganic, Physical, Theoretical and Analytical Chemistry*, vol. 52, no. 3, pp. 317–326, 2013.
- [328] H. Parastar, H. Ebrahimi-Najafabadi, and M. Jalali-Heravi, "Multivariate curve resolution-particle swarm optimization:

- a high-throughput approach to exploit pure information from multi-component hyphenated chromatographic signals,” *Analytica Chimica Acta*, vol. 772, pp. 16–25, 2013.
- [329] M. Khajeh, M. Kaykhai, and A. Sharafi, “Application of PSO-artificial neural network and response surface methodology for removal of methylene blue using silver nanoparticles from water samples,” *Journal of Industrial and Engineering Chemistry*, vol. 19, no. 5, pp. 1624–1630, 2013.
- [330] Y. Wu, B. Liu, M. Li, and K. Tang, “Prediction of CO₂ solubility in polymers by radial basis function artificial neural network based on chaotic self-adaptive particle swarm optimization and fuzzy clustering method,” *Chinese Journal of Chemistry*, vol. 31, no. 12, pp. 1564–1572, 2013.
- [331] M. Khajeh and K. Dastafkan, “Removal of molybdenum using silver nanoparticles from water samples: particle swarm optimization-artificial neural network,” *Journal of Industrial and Engineering Chemistry*, vol. 20, no. 5, pp. 3014–3018, 2014.
- [332] P. Thakral, V. Kapoor, and A. K. Bakhshi, “*In silico* architecturing of novel hetero-aromatic bicyclic copolymers using particle swarm optimization algorithm,” *Indian Journal of Chemistry Section A: Inorganic, Bio-Inorganic, Physical, Theoretical & Analytical Chemistry*, vol. 53, no. 1, pp. 9–16, 2014.
- [333] S. Y. Sun and J. W. Li, “Parameter estimation of methanol transformation into olefins through improved particle swarm optimization with attenuation function,” *Chemical Engineering Research & Design*, vol. 92, no. 11, pp. 2083–2094, 2014.
- [334] A. N. Skvortsov, “Estimation of rotation ambiguity in multivariate curve resolution with charged particle swarm optimization (cPSO-MCR),” *Journal of Chemometrics*, vol. 28, no. 10, pp. 727–739, 2014.
- [335] M. A. Khansary and A. H. Sani, “Using genetic algorithm (GA) and particle swarm optimization (PSO) methods for determination of interaction parameters in multicomponent systems of liquid-liquid equilibria,” *Fluid Phase Equilibria*, vol. 365, pp. 141–145, 2014.
- [336] R. Nasimi and R. Irani, “Identification and modeling of a yeast fermentation bioreactor using hybrid particle swarm optimization-artificial neural networks,” *Energy Sources, Part A: Recovery, Utilization and Environmental Effects*, vol. 36, no. 14, pp. 1604–1611, 2014.
- [337] S. Saraswathi, S. Sundaram, N. Sundararajan, M. Zimmermann, and M. Nilsen-Hamilton, “ICGA-PSO-ELM approach for accurate multiclass cancer classification resulting in reduced gene sets in which genes encoding secreted proteins are highly represented,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 2, pp. 452–463, 2011.
- [338] N. Mansour, F. Kanj, and H. Khachfe, “Particle swarm optimization approach for protein structure prediction in the 3D HP model,” *Interdisciplinary Sciences: Computational Life Sciences*, vol. 4, no. 3, pp. 190–200, 2012.
- [339] M. Karabulut and T. Ibriki, “A Bayesian Scoring Scheme based Particle Swarm Optimization algorithm to identify transcription factor binding sites,” *Applied Soft Computing Journal*, vol. 12, no. 9, pp. 2846–2855, 2012.
- [340] Y. J. Liu, X. Shi, G. Huang, B. Li, and L. Zhao, “Classification of diffuse large B cell lymphoma gene expression data based on two-layer particle swarm optimization,” in *Proceedings of the 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '13)*, pp. 422–427, Shenyang, China, July 2013.
- [341] M. Salahi, A. Jamalian, and A. Taati, “Global minimization of multi-funnel functions using particle swarm optimization,” *Neural Computing and Applications*, vol. 23, no. 7-8, pp. 2101–2106, 2013.
- [342] Z. Du, Y. Zhu, and W. Liu, “Combining quantum-behaved PSO and K2 algorithm for enhancing gene network construction,” *Current Bioinformatics*, vol. 8, no. 1, pp. 133–137, 2013.
- [343] L.-Y. Chuang, H.-Y. Lane, Y.-D. Lin, M.-T. Lin, C.-H. Yang, and H.-W. Chang, “Identification of SNP barcode biomarkers for genes associated with facial emotion perception using particle swarm optimization algorithm,” *Annals of General Psychiatry*, vol. 13, article 15, 2014.
- [344] M. Mandal and A. Mukhopadhyay, “A graph-theoretic approach for identifying non-redundant and relevant gene markers from microarray data using multiobjective binary PSO,” *PLoS ONE*, vol. 9, no. 3, Article ID e90949, 2014.
- [345] K.-H. Chen, K.-J. Wang, M.-L. Tsai et al., “Gene selection for cancer identification: a decision tree model empowered by particle swarm optimization algorithm,” *BMC Bioinformatics*, vol. 15, no. 1, article 49, 2014.
- [346] P. J. García Nieto, E. García-Gonzalo, J. R. Alonso Fernández, and C. Díaz Muñoz, “Hybrid PSO-SVM-based method for long-term forecasting of turbidity in the Nalón river basin: a case study in Northern Spain,” *Ecological Engineering*, vol. 73, pp. 192–200, 2014.
- [347] J. T. Mentzer, W. DeWitt, J. S. Keebler et al., “Defining supply chain management,” *Journal of Business Logistics*, vol. 22, no. 2, pp. 1–25, 2001.
- [348] L. Altamirano, M. C. Riff, I. Araya, and L. Trilling, “Anesthesiology nurse scheduling using particle swarm optimization,” *International Journal of Computational Intelligence Systems*, vol. 5, no. 1, pp. 111–125, 2012.
- [349] Z. G. Zeng, J. Xu, S. Wu, and M. Shen, “Antithetic method-based particle swarm optimization for a queuing network problem with fuzzy data in concrete transportation systems,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 29, no. 10, pp. 771–800, 2014.

Research Article

Recognition of Mixture Control Chart Pattern Using Multiclass Support Vector Machine and Genetic Algorithm Based on Statistical and Shape Features

Min Zhang and Wenming Cheng

School of Mechanical Engineering, Southwest Jiaotong University, Chengdu 610031, China

Correspondence should be addressed to Min Zhang; zhmzhangmin16@126.com

Received 4 July 2014; Revised 31 October 2014; Accepted 5 November 2014

Academic Editor: Yudong Zhang

Copyright © 2015 M. Zhang and W. Cheng. 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.

Control charts have been widely utilized for monitoring process variation in numerous applications. Abnormal patterns exhibited by control charts imply certain potentially assignable causes that may deteriorate the process performance. Most of the previous studies are concerned with the recognition of single abnormal control chart patterns (CCPs). This paper introduces an intelligent hybrid model for recognizing the mixture CCPs that includes three main aspects: feature extraction, classifier, and parameters optimization. In the feature extraction, statistical and shape features of observation data are used in the data input to get the effective data for the classifier. A multiclass support vector machine (MSVM) applies for recognizing the mixture CCPs. Finally, genetic algorithm (GA) is utilized to optimize the MSVM classifier by searching the best values of the parameters of MSVM and kernel function. The performance of the hybrid approach is evaluated by simulation experiments, and simulation results demonstrate that the proposed approach is able to effectively recognize mixture CCPs.

1. Introduction

In today's manufacturing and service industries, control charts are particularly important tools to improve product quality and monitor production process. Various kinds of control charts have been developed by different quality attributes and control targets. Recognizing control chart patterns (CCPs) is one of the most prevalently used techniques to detect process disturbances, equipment malfunctions, or other special events. In general, six basic CCPs are commonly exhibited by control charts, including normal (NOR), cyclic (CC), increasing trend (IT), decreasing trend (DT), upward shift (US), and downward shift (DS). Figure 1 shows these six types of control chart patterns [1]. Over the past two decades, attention has been given to improve the recognition accuracies of these basic CCPs using normalized original data. Automatic CCPs recognition was an active research area in last decade but has not yet been realized fully.

There are numerous research papers on CCPs organization. Most of the previous studies are concerned with the recognition of single abnormal CCPs [2–4]. However, in

practice, the observed process data may be mixture CCPs, which may be combined with two or three basic patterns. Compared to the basic patterns, the mixture patterns are more difficult to recognize and result in serious performance degradation for patterns recognition. So it is a challenging task to identify mixture patterns effectively. Only a few studies have reported on mixture patterns recognition [5–8]. Guh and Tannock [5] use the back-propagation neural network to recognize the mixture CCPs. H. Yang and S. Yang [6] propose an efficient statistical correlation coefficient method for the recognition of mixture CCPs. Chen et al. [7] integrate wavelet method and back-propagation neural network for online recognition of mixture CCPs. Lu et al. [8] propose a hybrid system that uses independent component analysis and supports vector machine to recognition mixture CCPs.

Feature extraction plays an important role in CCPs recognition. Most of the existing literatures use normalized original data as the inputs. These data normally generate large structures and are not very effective for complicated recognition problems. A smaller data size can lead to faster

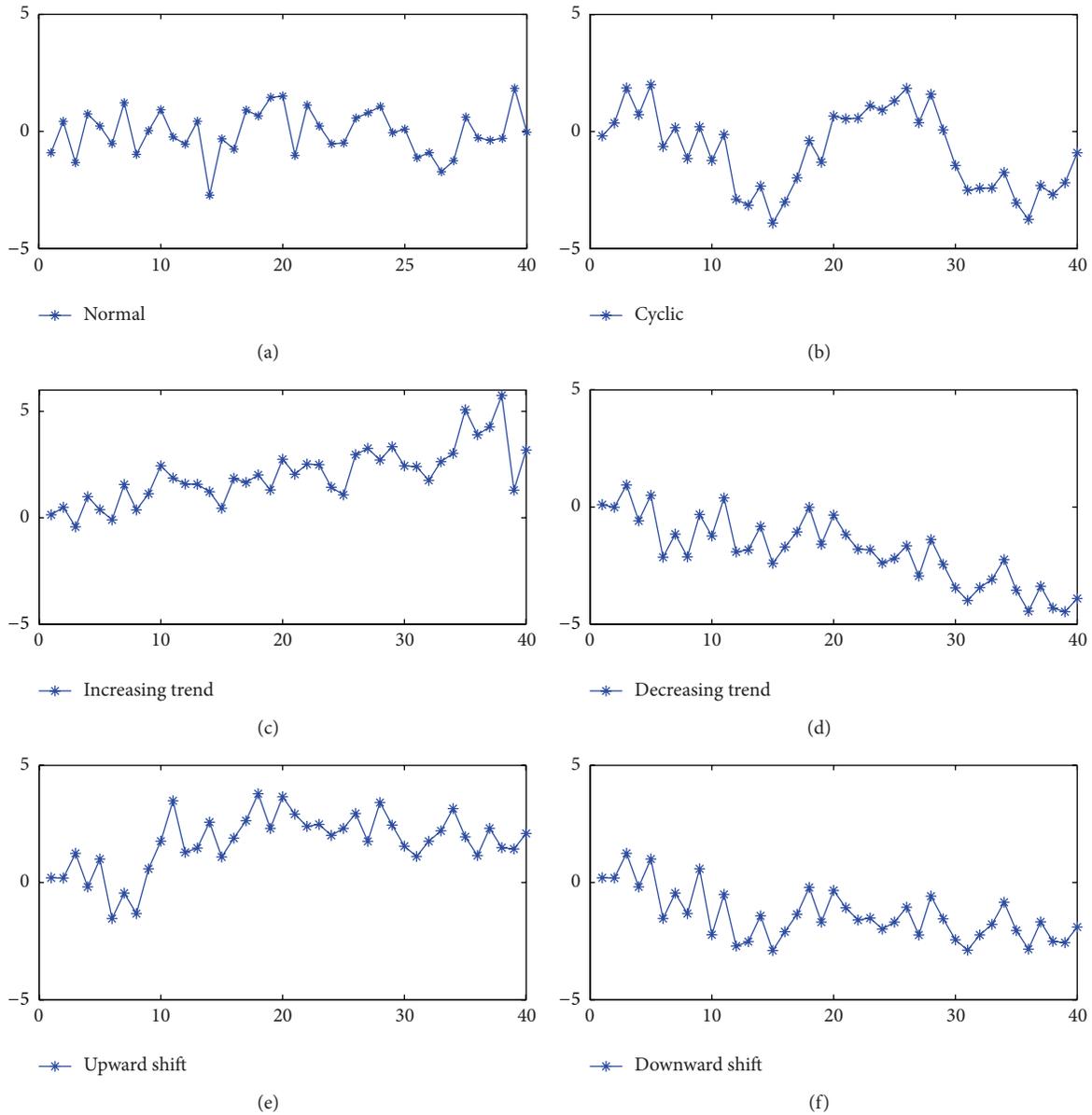


FIGURE 1: Six basic control chart patterns: (a) normal (NOR), (b) cyclic (CYC), (c) increasing trend (IT), (d) decreasing trend (DT), (e) upward shift (US), and (f) downward shift (DS).

training and more efficiency. Regarding this, researchers have proposed various methods to extract features for CCPs recognition [2, 9, 10]. Ranaee et al. [2] use both shape features and statistical features as the data inputs. The results show that this method is good for control chart recognition. Hassan et al. [9] introduce feature-based control chart pattern recognition. Six statistical features were proposed: mean, variance, skewness, mean-square value, autocorrelation, and cusum. It is intended to improve the performance of CCPs recognizer by smaller size features. Gauri and Chakraborty [10] present the improved feature extraction from a large number of potentially useful features using a CART-based approach. And other feature extraction methods are proposed for eliminating the duplicated information, like independent

component analysis (ICA) [11], fisher discriminate analysis (FDA) [12], and principal component analysis (PCA) [13, 14]. The feature extraction efforts cited above did not approach a suitable set of features. In this paper, thirteen features that consist of both statistical and shape features of the CCPs are initially chosen. It is a well-established dimensionality reduction technique, which can be employed to compress the noise and correlated measurements, so that makes the data into a simpler and smaller informative subspace for measurement data sets.

Traditionally, CCPs were analyzed and interpreted manually. Until the end of the 1980s, expert systems were employed for control chart patterns recognition [15, 16]. With the development of computer technology, machine

learning techniques have been widely adopted in automatic process monitoring. In particular, artificial neural networks (ANNs) are the most frequently used in control chart patterns recognition [17–20]. The use of ANNs has overcome some drawbacks in the traditional expert system method. Artificial neural networks utilize a multilayer perception with back propagation training to classify unnatural patterns and show higher accuracy. In subsequent studies, many other methods like decision tree, fuzzy clustering, and wavelet analysis are combined with ANNs to recognize CCPs [19, 20].

However, ANNs also suffer from several weaknesses, such as the need for a large amount of training data, bad generalization ability, the risk of model over-fitting, difficulty to obtain stable solution, and getting into a local extremum easily. The application of ANNs is limited due to these weaknesses. Support vector machine (SVM), based on statistical learning theory, is proposed to recognize CCPs because of its excellent performance in the practical application. It mainly used the principle of structural risk minimization, which makes it have greater generalization ability when there is a small sample, and is superior to the principle of the empirical risk minimization principle as artificial neural networks [7, 21, 22]. The biggest problems encountered in setting up the SVM model are how to select the kernel function and its parameters values. The parameter set of the penalty parameter and kernel function parameter should be optimized.

The purpose of this study is to develop an intelligent hybrid CCPs recognition model that can be used for mixture CCPs to improve the recognition accuracy. This paper considers the six basic and four mixture CCPs and generates their statistical and shape features as the inputs and multiclass support vector machine (MSVM) as classifier. At the same time, genetic algorithm (GA) is chosen as an optimization tool to optimize the MSVM parameters. This model will improve CCPs recognition performance.

2. Modeling for Control Chart Patterns Recognition

The aim of this model is to recognize CCPs effectively and automatically. Figure 2 shows the schematic diagram representing the procedure of the CCPs recognition, in which three modules are in series: feature extraction, classifier, and parameters optimization (F-MSVM-GA).

In the feature extraction module, statistical and shape features of observation data are used as the data inputs for the classifier. As we know, every control chart pattern has different properties, and features represent the properties of various CCPs. If some effective features are chosen to reflect the pattern, it is easier to recognize the abnormal patterns. Original data as the inputs usually have large data and are not very effective for complicated recognition problems. In this paper, statistical and shape features of CCPs as the feature extraction method are utilized to get the suitable data. In classifier module, an MSVM classifier is developed for recognizing the basic and mixture patterns. In order to achieve satisfactory recognition performance,

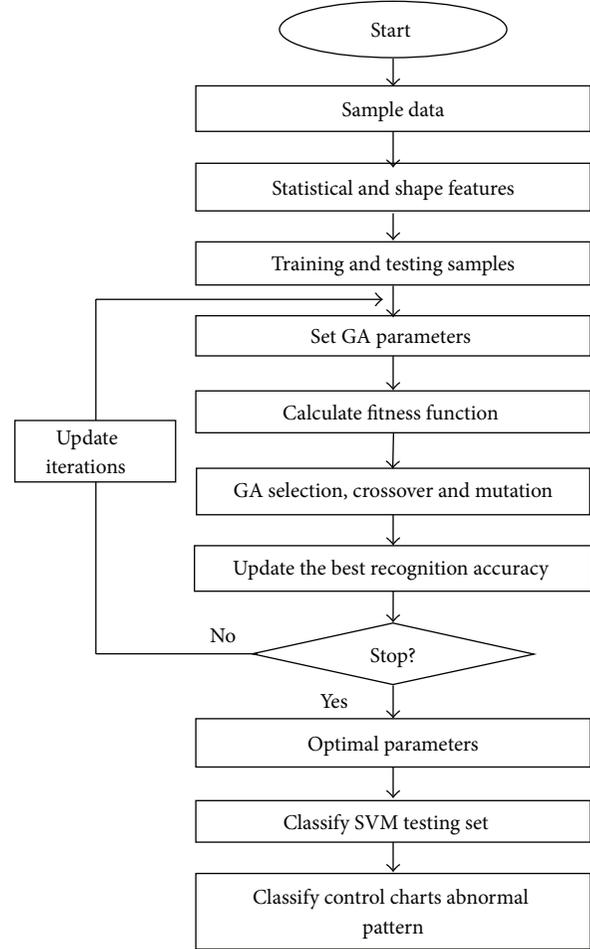


FIGURE 2: Flow chart of F-MSVM-GA model.

the MSVM classifier needs to be properly designed, trained, and tested. However, using MSVM has some difficulties, like how to select the optimal kernel function type and the most appropriate hyperparameters values for MSVM training and testing stages. Therefore, genetic algorithm is applied for finding the optimum values of hyperparameters, that is, the kernel parameter and classifier parameters in parameters optimization module.

2.1. Statistical and Shape Features. The patterns can be described in the original data. The statistical features and shape features can be got from the original data. It is efficient to simplify the data number and get the useful information. In this paper, eight statistical features and five shape features are chosen to reflect the patterns; these thirteen features are, respectively, shown below [2].

2.1.1. Statistical Features. They are as follows: mean, standard deviation, mean-square value, autocorrelation, positive cusum, negative cusum, skewness, and kurtosis.

(1) *Mean.* The mean for normal and cyclic pattern is around zero, while that for other patterns is different from zero.

Therefore, it may be a good candidate to differentiate normal and cyclic patterns from other patterns:

$$\bar{x} = \frac{1}{T} \sum_{i=1}^T x_i. \quad (1)$$

(2) *Standard Deviation*. Standard deviation of sample data, each mode performance is different

$$\sigma = \sqrt{\frac{1}{T} \sum_{i=1}^T (x_i - \bar{x})^2}. \quad (2)$$

(3) *Mean-Square Value*. Consider

$$\bar{x}^2 = \frac{x_1^2 + x_2^2 + \dots + x_N^2}{N}. \quad (3)$$

(4) *Average Autocorrelation*. This paper takes the average of correlation degree between property values for each sample:

$$R_{xx}(m) = \sum_{n=0}^{N-m-1} x_{n+m} y_n, \quad A_R = \text{mean}(R_{xx}). \quad (4)$$

(5) *Positive Cusum*. Sample data points are greater than the average and then cumulate the gap between data and their average:

$$C_i^+ = \max[0, (x_i - \bar{x}) + C_{i-1}^+]. \quad (5)$$

(6) *Negative Cusum*. Sample data points are smaller than the average and then cumulate the gap between data and their average:

$$C_i^- = \max[0, (\bar{x} - x_i) + C_{i-1}^-]. \quad (6)$$

(7) *Skewness*. It provides information regarding the degree of asymmetry:

$$a_3 = \frac{\sum_{i=1}^T (x_i - \bar{x})^3}{N\sigma^3}. \quad (7)$$

(8) *Kurtosis*. It measures the relative peakness or flatness of its distribution:

$$a_4 = \frac{\sum_{i=1}^T (x_i - \bar{x})^4}{N\sigma^4}. \quad (8)$$

2.1.2. *Shape Features*. They are as follows: slope, N1, N2, APML, and APLS.

(1) *Slope*. The slope of the least-square line: the slope for normal and cyclic pattern is around zero, while that for other patterns is greater than zero. Therefore, it may be a good candidate to differentiate normal and cyclic patterns from other patterns:

$$\text{slope}(\beta_1) = \frac{\sum_{i=1}^T (t_i - \bar{t})(x_i - \bar{x})}{\sum_{i=1}^T (t_i - \bar{t})^2}, \quad (9)$$

$$\beta_0 = \bar{x} - \beta_1 \bar{t}.$$

(2) *N1*. The number of mean crossing: it is almost zero for shift and trend patterns but very high for normal patterns; cyclic pattern is the intermediate pattern. The feature differences can distinguish the normal and cyclic from shift and trend patterns:

$$\text{if } (x_i - \bar{x})(x_{i+1} - \bar{x}) < 0, \quad n1 = 0; \text{ else } n1 = 1. \quad (10)$$

(3) *N2*. The number of least-square line crossing: this feature is the highest for normal and trend patterns, intermediate for shift patterns, and the lowest for cyclic patterns. Thus it can be used for separation of normal and trend patterns from others:

$$\text{if } (x_i - \hat{x}_i)(x_{i+1} - \hat{x}_{i+1}) < 0, \quad n2 = 0; \text{ else } n2 = 1. \quad (11)$$

(4) *APML*. The area between the pattern and its mean line: this feature is the lowest for normal pattern; therefore, it differentiates the normal pattern from others:

$$\text{ACML} = C_i^+ + C_i^-. \quad (12)$$

(5) *APLS*. The area between the pattern and its least-square line: normal and trend patterns have lower values than shift and cyclic patterns. Thus it can be used to distinguish normal and trend patterns from shift and cyclic patterns:

$$\text{APSL} = |x_i - \hat{x}_i|. \quad (13)$$

2.2. *Support Vector Machine*. Basic SVM is invented by Vapnik of the AT&T Bell lab team. It is created based on the VC dimension theory and structural risk minimization of statistical learning theory. So that gets the best solution between model complexities and learning ability according to the limited sample information. The basic SVM deals with two-class problems. However, it can be extended to Multiclass SVM [7, 21–23].

An SVM performs classification tasks by constructing optimal separating hyperplanes (OSHs). An OSH maximizes the margin between the two nearest data points belonging to two separate classes. Suppose that the training set, (x_i, y_i) , $i = 1, 2, \dots, n$, $x_i \in R^m$, $y_i \in \{-1, 1\}$, can be separated by the hyperplane $w \cdot x_i + b = 0$, where n is the number of sample observations and m is the dimension of each observation and w is the weight vector and b is the bias. If this hyperplane separates the data from two classes with maximal margin width $2/\|w\|^2$ and all the points on the boundary are named the support vector, the SVM solves the following optimization problem:

$$\min_w \quad \frac{1}{2} \|w\|^2$$

$$\text{s.t. } y_i (w^T x_i + b) \geq 1. \quad (14)$$

This is a convex quadratic programming (QP) problem, and Lagrange multipliers $(\alpha_i, i = 1, 2, \dots, n; \alpha_i \geq 0)$ are used to solve it. And, for input data with a high noise level, an SVM using soft margins can be expressed with the introduction of

the nonnegative slack variables ($\xi_i, i = 1, 2, \dots, n$). Equation (14) is transformed into the following constrained form:

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j (x_i \cdot x_j) \alpha_i \alpha_j - \sum_{j=1}^n \alpha_j \\ \text{s.t.} \quad & \sum_{i=1}^n y_i \alpha_i = 0 \\ & 0 \leq \alpha_i \leq C, \quad i = 1, \dots, n. \end{aligned} \quad (15)$$

In (15), C is the penalty factor; it determines the penalty degree of the error. It can be viewed as a tuning parameter, which can be utilized to control the trade-off between maximizing the margin and the classification error.

An MSVM method is adopted in the classifier stage. There are two methods: one-against-all (OAA) or one-against-one (OAO). Suppose that it has an N -class pattern recognition problem; N independent SVMs are constructed and each of them is trained to separate one class of samples from all others. When testing the system after all the SVMs are trained, a sample is input to all the SVMs. Suppose that this sample belongs to class $N1$; ideally only the SVM trained to separate class $N1$ from the others can have a positive response. Another method is called one-against-one (OAO) method. For an N -class problem, SVMs are constructed and each of them is trained to separate one class from another class. Again, the decision of a testing sample is based on the voting results of these SVMs. In this paper, OAO is adopted for patterns recognition [24].

In the nonlinearly separable cases, which cannot be linearly separated in the input space, the SVM uses the kernel method to transform the original input space into a high dimensional feature space, where an optimal linear separating hyperplane can be found. Although there are several types of kernel function, the most widely used kernel function is the radial basis function (RBF), which is defined as

$$K(x, x_i) = \exp \left\{ -\frac{\|x - x_i\|^2}{\sigma^2} \right\} = \exp \{-\gamma \|x - x_i\|^2\}. \quad (16)$$

The largest problems encountered in MSVM are how to select the penalty parameter (C) and kernel function parameters value (γ). The GA is used to search for the best value of parameters in MSVM classifier.

2.3. Genetic Algorithms (GA). GA is a powerful tool in the field of global optimization. It has better search efficiency, robustness, and parallel compared with traditional optimization algorithms. Genetic algorithms belong to the larger class of evolutionary algorithms, which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover.

In this paper, OAO and RBF kernel function are adopted for MSVM; the performance of an MSVM is mainly impacted by the setting of parameters of two parameters (C and γ). The GA is used to search for the best value of parameters in

MSVM classifier. The particle has two dimensions: C and γ ; the accuracy of training set is selected as the fitness function. The steps are as follows [25].

Step 1. Set GA parameters, like the number of population, evolutionary generation, crossover and mutation probability, and parameter ranges.

Step 2. Optimize coding parameters and initialize the population.

Step 3. Optimize decoding parameters and calculate the recognition rate, decode chromosomes of population, select the training set of model, and use training set recognition rate as the fitness function of the GA algorithm, so that we obtain the optimal MSVM parameters (C and γ).

Step 4. Genetic manipulation (selection, crossover, and mutation): each chromosome does the selection, crossover, and mutation based on the fitness, thus excluding low fitness chromosomes and leaving high fitness chromosomes. The new group members are outstanding in the previous generation groups, which are better than the previous generation. GA performs iteratively until meeting some predetermined optimized targets.

Step 5. Get optimal parameters: decode the best chromosome, use the optimal parameters to train the training data in support vector machine classifier, and ultimately get the optimized support vector machine classifier.

3. Simulation and Results Analysis

3.1. Data Generation. In order to analyze the CCPs recognition, Monte Carlo method is used to get the sample data. The following equation is applied to generate the data points for six basic patterns; different parameters are shown in Table 1 [8]:

$$X(t) = \mu + x(t) + d(t), \quad t = 1, 2, \dots, T, \quad (17)$$

where $X(t)$ means the value of sample data at time t ; μ is the mean of data; $x(t) = r \times \sigma$, r is the random value of standard normal distributed between -1 and 1 , σ is the standard deviation of normal distribution, and $d(t)$ is the abnormal value. We chose $\mu = 0$, $\sigma = 1$ and use the 40 data points of observation window as inputs of the feature extraction model. Every pattern generates 100 sample data.

However, the observed process data may be mixture control chart patterns in practice, which is combined with two or three basic patterns. Figure 3 shows four kinds of mixture CCPs, which are combined with cyclic, increasing trend and decreasing shift. We know that the principles of increasing/decreasing trend or upward/downward shift are similar, so the increasing trend and downward shift are chosen for the mixture CCPs. And sample data of mixture CCPs can be generated by different parameters in Table 3. Six basic CCPs (see Figure 1) and four mixture CCPs (see Figure 3) are, respectively, used for training and testing the proposed F-MSVM-GA method in this study (Table 2).

TABLE 1: The parameters of six basic patterns.

Control chart patterns	Pattern equations	Parameters
Normal (NOR)	$X(t) = \mu + r \times \sigma$	Mean $\mu = 0$, standard deviation $\sigma = 1$
Cyclic (CYC)	$X(t) = \mu + r \times \sigma + \alpha \sin(2\pi t/t')$	Period $t' = (8\sigma \text{ to } 16\sigma)$, amplitude $\alpha = (1.5\sigma \text{ to } 2.5\sigma)$
Increasing (decreasing) trend (IT/DT)	$X(t) = \mu + r \times \sigma \pm g \times t$	Gradient $g = (0.05\sigma \text{ to } 0.15\sigma)$
Increasing (decreasing) shift (US/DS)	$X(t) = \mu + r \times \sigma \pm k \times s (k = 1 \text{ if } t > P, \text{ else } k = 0)$	Shift magnitude $s = (1.5\sigma \text{ to } 2.5\sigma)$, shift position $P = 2$

TABLE 2: The parameters of mixture patterns.

Control chart patterns	Pattern equations	Parameters
Cyclic + increasing trend (CT)	$X(t) = \mu + r \times \sigma + \alpha \sin(2\pi t/t') + g \times t$	$t' = (8\sigma \text{ to } 16\sigma)$, $\alpha = (1.5\sigma \text{ to } 2.5\sigma)$, $g = (0.05\sigma \text{ to } 0.15\sigma)$
Cyclic + decreasing shift (CS)	$X(t) = \mu + r \times \sigma + \alpha \sin(2\pi t/t') - k \times s$	$t' = (8\sigma \text{ to } 16\sigma)$, $\alpha = (1.5\sigma \text{ to } 2.5\sigma)$, $s = (1.5\sigma \text{ to } 2.5\sigma)$, $P = 2$
Increasing trend + decreasing shift (TS)	$X(t) = \mu + r \times \sigma + g \times t - k \times s$	$g = (0.05\sigma \text{ to } 0.15\sigma)$, $s = (1.5\sigma \text{ to } 2.5\sigma)$, $P = 2$
Cyclic + increasing trend + decreasing shift (CTS)	$X(t) = \mu + r \times \sigma + \alpha \sin(2\pi t/t') + g \times t - k \times s$	$t' = (8\sigma \text{ to } 16\sigma)$, $\alpha = (1.5\sigma \text{ to } 2.5\sigma)$, $g = (0.05\sigma \text{ to } 0.15\sigma)$, $s = (1.5\sigma \text{ to } 2.5\sigma)$, $P = 2$

TABLE 3: Parameters in GA.

Parameters name	Value
Maximum number of iterations	100
Population size	20
Crossover probability	0.4
Mutation probability	0.01

3.2. *Parameters of MSVM and GA.* The performance of MSVM is influenced by its parameters. As Section 2.2 analysis, MSVM based on RBF kernel function is chosen in this study. Related parameters C and γ for this kernel were varied in the fixed ranges $[0.1, 100]$ and $[0.01, 10]$, so as to cover high or small regulations of the classifier and fat or thin kernels, respectively. In the GA optimization module, there are several coefficients, whose values can be adjusted to produce better performances during training in this study, are summarized in Table 3.

4. Performance Analyses

In this section, we measure the performance of the proposed recognizer. For this purpose, we have previously generated 10 patterns, 100 of each type; every sample has 40 data points of observation windows. And we have used about 50% of the sample for training the classifier and the rest for testing. The testing samples can be used to estimate the performance of recognizer for each pattern and then compute the average recognition accuracy of CCPs. Several performances are done to verify the effectiveness of the proposed model.

4.1. *Performance of Recognizer in Optimization.* First, we have applied MSVM classifier with different features.

Table 4 indicates the recognition accuracy (RA) of proposed F_MSVM_GA model on the 13 statistical and shape features and GA optimization algorithm. In order to demonstrate the superior performance of the proposed F_MSVM_GA scheme, MSVM using 13 features as inputs without GA optimization (called F_MSVM) is constructed; the performance results are shown in Table 5.

As reported in Tables 4 and 5, the average recognition accuracies of F_MSVM_GA and F_MSVM are 97.4% and 91.4%. The proposed F_MSVM_GA model has better recognition performance for the mixture CCPs, especially in TS and CTS. Genetic algorithm searches for the best combination of MSVM classifier parameters to gain the fitness maximum, so as to improve recognition rate of testing samples.

4.2. *Performance of Recognizer in Different Features.* Feature extraction can lead to faster training and more efficiency in CCPs recognition. Thirteen statistical and shape features are utilized as the inputs in this paper. In order to explain its effectiveness, MSVM classifier using the original 40 data points as the inputs (called D_MSVM) is constructed. Table 6 shows the recognition accuracy of mixture CCPs.

The average recognition accuracies of D_MSVM (78.0%) and F_MSVM (91.4%) show that feature extraction method plays an important role in improving the recognition accuracy. From the data, we can find that mixture control chart patterns are difficult to recognize due to the complex relation, but the result is much better after using statistical and shape features extraction method.

4.3. *Comparison of the Proposed Method with BP Method.* Artificial neural networks (ANNs) are the most frequently used in control chart patterns recognition. It utilized a

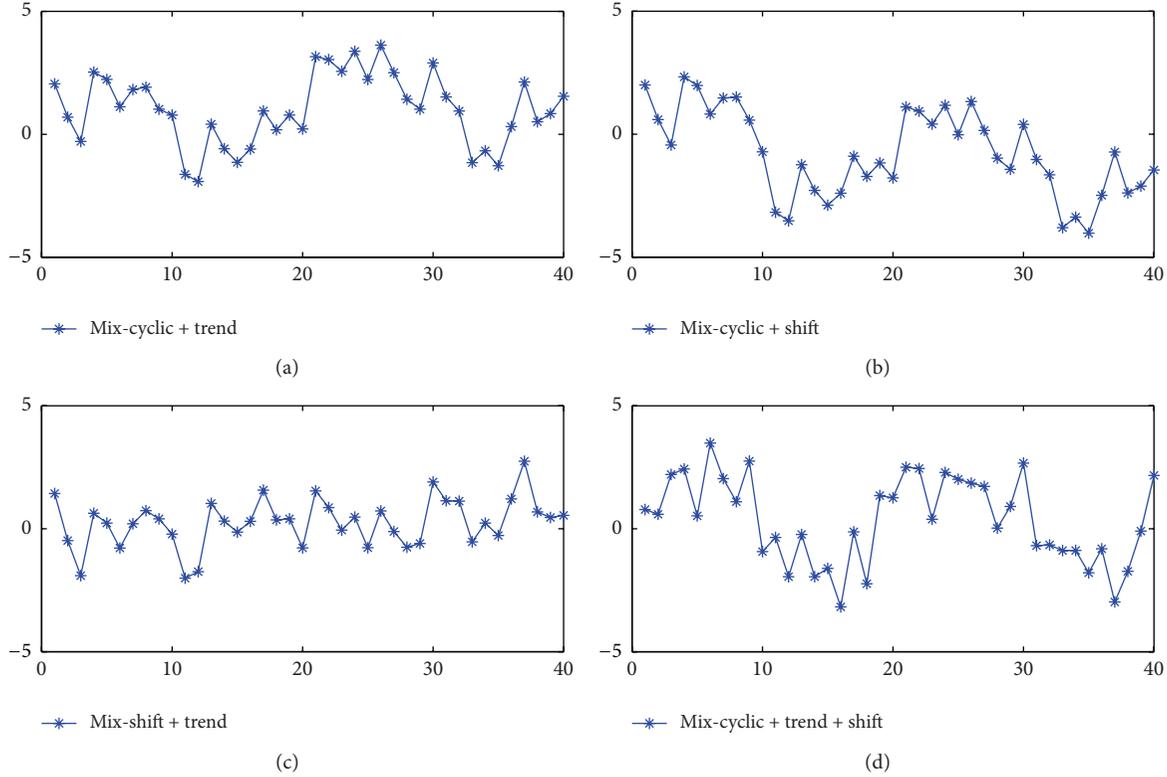


FIGURE 3: Four mixture CCPs: (a) mix-cyclic + trend (CT), (b) mix-cyclic + shift (CS), (c) mix-trend shift (TS), and (d) mix-cyclic + trend + shift (CTS).

TABLE 4: Recognition accuracies with the proposed F_MSVM_GA model.

True pattern	Identified pattern (%)									
	NOR	CYC	IT	DT	US	DS	CT	CS	TS	CTS
NOR	100	0	0	0	0	0	0	0	0	0
CYC	0	98	0	0	0	0	0	0	0	2
IT	0	0	98	0	0	0	2	0	0	0
DT	0	0	0	100	0	0	0	0	0	0
US	0	0	0	0	100	0	0	0	0	0
DS	0	0	0	0	0	98	0	2	0	0
CT	0	0	2	0	0	0	98	0	0	0
CS	0	0	0	0	0	0	0	100	0	0
TS	2	0	0	0	0	0	0	0	96	2
CTS	0	10	0	0	0	0	0	0	2	88
Average	97.6									

multilayer perception with back-propagation training to classify abnormal patterns. Back-propagation (BP) method is a common method of training artificial neural networks used in conjunction with an optimization method such as gradient descent. We define that the number of input neurons is 50 and the number of hidden layers is 5. Table 7 shows the performance results.

Compare MSVM models; the accuracy of BP method is only 65.4%, much lower than the other MSVM methods. The reason is that BP neural network quite depends on the quantity and quality of the sample data, but only 50 training

samples are considered in this study; it belongs to the small sample noise problem.

We have compared the proposed model with other approaches. This comparison can be seen in Figure 4, and six basic CCPs and mixture CCPs are, respectively, numbered from 1 to 10.

5. Conclusion

Control charts are the most useful tools in statistical process control, and mixture control chart patterns are more and

TABLE 5: Recognition accuracies with the F_MSVM model.

True pattern	Identified pattern (%)									
	NOR	CYC	IT	DT	US	DS	CT	CS	TS	CTS
NOR	100	0	0	0	0	0	0	0	0	0
CYC	2	86	0	0	0	0	0	0	6	6
IT	0	0	96	0	0	0	4	0	0	0
DT	0	0	0	100	0	0	0	0	0	0
US	0	0	0	0	100	0	0	0	0	0
DS	0	0	0	2	0	96	0	2	0	0
CT	0	0	6	0	0	0	94	0	0	0
CS	0	0	0	2	0	0	0	98	0	0
TS	6	0	12	0	0	0	0	0	82	0
CTS	0	34	0	0	0	0	0	0	4	62
Average	91.4									

TABLE 6: Recognition accuracies with the D_MSVM model.

True pattern	Identified pattern (%)									
	NOR	CYC	IT	DT	US	DS	CT	CS	TS	CTS
NOR	70	4	2	24	0	0	0	0	0	0
CYC	2	76	2	0	0	0	14	0	0	6
IT	14	0	80	0	2	0	4	0	0	0
DT	0	4	0	88	0	0	0	0	2	6
US	0	0	0	0	98	0	2	0	0	0
DS	0	0	0	2	0	92	0	4	2	0
CT	2	22	2	0	2	0	72	0	0	0
CS	0	0	0	2	0	2	0	74	0	22
TS	0	0	0	14	0	12	0	0	64	10
CTS	0	4	0	12	0	2	0	16	0	66
Average	78.0									

TABLE 7: Recognition accuracies with the BP model.

True pattern	Identified pattern (%)									
	NOR	CYC	IT	DT	US	DS	CT	CS	TS	CTS
NOR	90	8	0	2	0	0	0	0	0	0
CYC	4	88	4	2	2	0	0	0	0	0
IT	0	0	56	32	10	2	0	0	0	0
DT	0	6	24	32	12	26	0	0	0	0
US	0	0	8	28	34	30	0	0	0	0
DS	0	0	0	2	10	74	10	2	2	0
CT	0	0	0	0	8	24	66	0	0	2
CS	0	0	0	0	0	0	22	78	0	0
TS	0	0	0	0	0	0	2	10	84	4
CTS	0	2	0	2	6	2	0	14	16	58
Average	65.2									

more widely used in manufacturing and service processes. Recognizing the mixture CCPs plays an important role in finding the abnormal quality problems. In this study, a hybrid method by integrating statistical and shape features extraction, MSVM, and GA are presented for recognizing the mixture CCPs. The proposed method initially uses statistical

and shape features to get effective input data; then the combination of MSVM and GA is applied to recognize the mixture patterns. GA is to optimize the parameters of MSVM kernel parameters. Six basic CCPs and four mixture CCPs are used in this study for evaluating the performance of the proposed method. From the experiments, the simulation

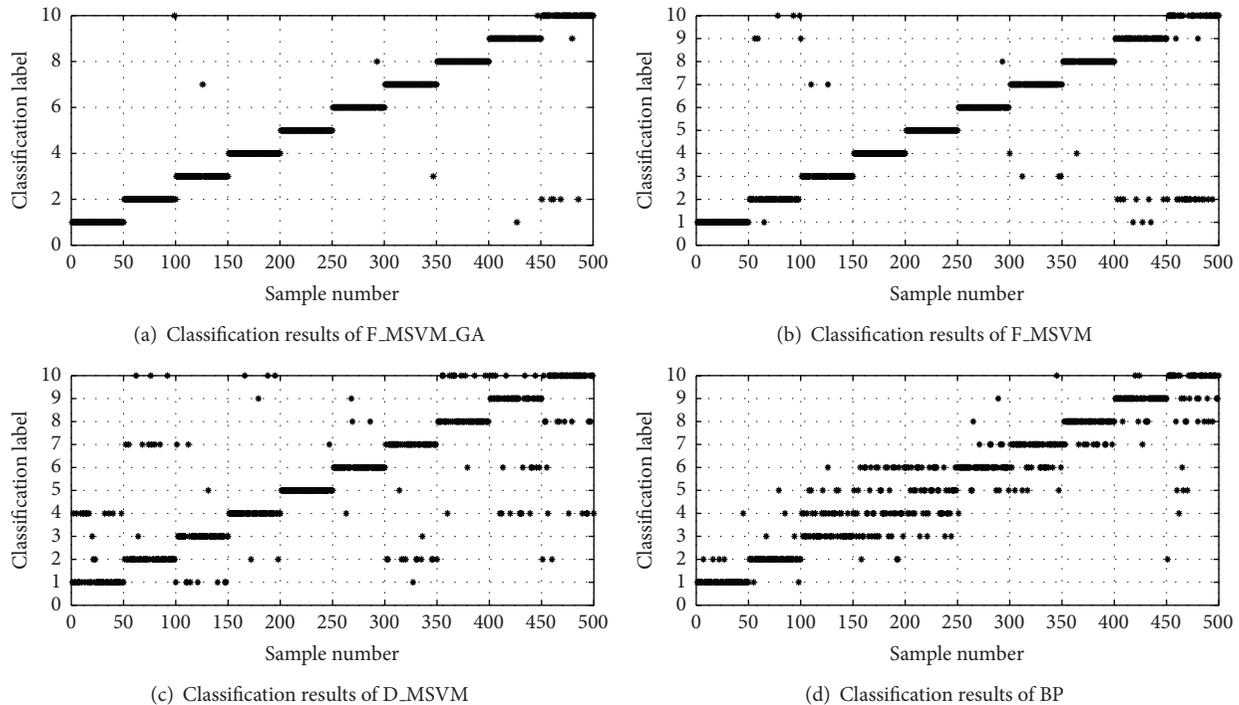


FIGURE 4: Mixture CCPs recognition accuracy results of the four methods: (a) F_MSVM_GA, (b) F_MSVM, (c) D_MSVM, and (d) BP.

results indicate that the intelligent hybrid method can achieve the highest average recognition accuracies in the tested methods.

The future work will be focused on the following aspects: (1) employing statistical and shape features method as feature extraction method which we will compare with other excellent feature extraction methods, (2) comparing GA with other intelligent algorithms, including particle swarm optimization, simulated annealing algorithm, and ant colony optimization, (3) researching the fundamental principles of mixture CCPs with the help of mathematicians, and (4) seeking economic explanation of our method with the help of economists.

Conflict of Interests

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

Acknowledgments

This work is financially supported by National Natural Science Foundation of China (NSFC) under Grant no. 51175442 and the Fundamental Research Funds for the Central Universities under Grant no. 2682014BR022.

References

- [1] D. C. Montgomery, *Introduction to Statistical Quality Control*, John Wiley & Sons, New York, NY, USA, 2001.
- [2] V. Ranaee, A. Ebrahimzadeh, and R. Ghaderi, "Application of the PSOSVM model for recognition of control chart patterns," *ISA Transactions*, vol. 49, no. 4, pp. 577–586, 2010.
- [3] R.-S. Guh, "A hybrid learning-based model for on-line detection and analysis of control chart patterns," *Computers and Industrial Engineering*, vol. 49, no. 1, pp. 35–62, 2005.
- [4] S. K. Gauri and S. Chakraborty, "Feature-based recognition of control chart patterns," *Computers & Industrial Engineering*, vol. 51, no. 4, pp. 726–742, 2006.
- [5] R.-S. Guh and J. D. T. Tannock, "Recognition of control chart concurrent patterns using a neural network approach," *International Journal of Production Research*, vol. 37, no. 8, pp. 1743–1765, 1999.
- [6] J. H. Yang and M. S. Yang, "A control chart pattern recognition system using a statistical correlation coefficient method," *Computers and Industrial Engineering*, vol. 48, no. 2, pp. 205–221, 2005.
- [7] Z. Chen, S. Lu, and S. Lam, "A hybrid system for SPC concurrent pattern recognition," *Advanced Engineering Informatics*, vol. 21, no. 3, pp. 303–310, 2007.
- [8] C.-J. Lu, Y. E. Shao, and P.-H. Li, "Mixture control chart patterns recognition using independent component analysis and support vector machine," *Neurocomputing*, vol. 74, no. 11, pp. 1908–1914, 2011.
- [9] A. Hassan, M. S. Nabi Baksh, A. M. Shaharoun, and H. Jamaluddin, "Improved SPC chart pattern recognition using statistical features," *International Journal of Production Research*, vol. 41, no. 7, pp. 1587–1603, 2003.
- [10] S. K. Gauri and S. Chakraborty, "Recognition of control chart patterns using improved selection of features," *Computers and Industrial Engineering*, vol. 56, no. 4, pp. 1577–1588, 2009.

- [11] M. Kano, S. Tanaka, S. Hasebe, I. Hashimoto, and H. Ohno, "Monitoring independent components for fault detection," *AIChE Journal*, vol. 49, no. 4, pp. 969–976, 2003.
- [12] Q. P. He, S. J. Qin, and J. Wang, "A new fault diagnosis method using fault directions in Fisher discriminant analysis," *AIChE Journal*, vol. 51, no. 2, pp. 555–571, 2005.
- [13] G. N. Costache, P. Corcoran, and P. Puslecki, "Combining PCA-based datasets without retraining of the basis vector set," *Pattern Recognition Letters*, vol. 30, no. 16, pp. 1441–1447, 2009.
- [14] H. Wang, Z. Song, and P. Li, "Fault detection behavior and performance analysis of principal component analysis based process monitoring methods," *Industrial & Engineering Chemistry Research*, vol. 41, no. 10, pp. 2455–2464, 2002.
- [15] J. A. Swift and J. H. Mize, "Out-of-control pattern recognition and analysis for quality control charts using LISP-based systems," *Computers & Industrial Engineering*, vol. 28, no. 1, pp. 81–91, 1995.
- [16] C. S. Cheng and N. F. Hubele, "Design of a knowledge-based expert system for statistical process control," *Computers and Industrial Engineering*, vol. 22, no. 4, pp. 501–517, 1992.
- [17] H. B. Hwang and N. F. Hubele, "Back-propagation pattern recognizers for X-bar control charts: methodology and performance," *Computers and Industrial Engineering*, vol. 24, no. 2, pp. 219–235, 1993.
- [18] R.-S. Guh and Y.-R. Shiue, "On-line identification of control chart patterns using self-organizing approaches," *International Journal of Production Research*, vol. 43, no. 6, pp. 1225–1254, 2005.
- [19] C. H. Wang, R. S. Guo, M. H. Chiang, and J. Y. Wong, "Decision tree based control chart pattern recognition," *International Journal of Production Research*, vol. 46, no. 17, pp. 124–134, 2008.
- [20] C.-H. Wang and W. Kuo, "Identification of control chart patterns using wavelet filtering and robust fuzzy clustering," *Journal of Intelligent Manufacturing*, vol. 18, no. 3, pp. 343–350, 2007.
- [21] S.-Y. Yang, D.-H. Wu, and H.-T. Su, "Abnormal pattern recognition method for control chart based on principal component analysis and support vector machine," *Journal of System Simulation*, vol. 18, no. 5, pp. 1314–1318, 2006 (Chinese).
- [22] C. Wu and L. Zhao, "Control chart pattern recognition based on wavelet analysis and SVM," *China Mechanical Engineering*, vol. 21, no. 13, pp. 1572–1576, 2010 (Chinese).
- [23] V. Ranaee and A. Ebrahimzadeh, "Control chart pattern recognition using a novel hybrid intelligent method," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 2676–2686, 2011.
- [24] Y. Zhang and L. Wu, "Classification of fruits using computer vision and a multiclass support vector machine," *Sensors*, vol. 12, no. 9, pp. 12489–12505, 2012.
- [25] Y. Zhang, S. Wang, and G. Ji, "A rule-based model for bankruptcy prediction based on an improved genetic ant colony algorithm," *Mathematical Problems in Engineering*, vol. 2013, Article ID 753251, 10 pages, 2013.

Research Article

A Novel WLAN Client Puzzle against DoS Attack Based on Pattern Matching

**Ali Ordi, Mazdak Zamani, Norbik Bashah Idris,
Azizah Abdul Manaf, and Mohd Shahidan Abdullah**

Advanced Informatics School, Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia

Correspondence should be addressed to Ali Ordi; ali.ordi@gmail.com

Received 17 September 2014; Accepted 7 November 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Ali Ordi 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.

Despite the popularity of 802.11 based networks, they suffer several types of DoS attack, launched by an attacker whose aim is to make an access point (AP) unavailable to legitimate users. One of the most common DoS attacks on 802.11 based networks is to deplete the resources of the AP. A serious situation like this can occur when the AP receives a burst of connection requests. This paper addresses this common DoS attack and proposes a lightweight puzzle, based on pattern-matching. Using a pattern-matching technique, this model adequately resists resource-depletion attacks in terms of both puzzle generation and solution verification. Using a sensible series of contextual comparisons, the outcomes were modelled by a simulator, and the security definition and proofs are verified, among other results.

1. Introduction

Despite an unprecedented growth in popularity, using an open shared transmission medium makes wireless LANs (WLAN) extremely vulnerable to many attacks [1]. A series of security extensions to 802.11 have already been ratified to fix some of these vulnerabilities. However, these extensions primarily deal with vulnerabilities related to unauthorized access and confidentiality breaches. As our dependence on wireless access increases, it also becomes essential to consider the issue of availability as another important security requirement [2]. Denial-of-service (DoS) attacks strike against availability, attempting to prevent legitimate users from accessing the network. There are many 802.11-specific DoS vulnerabilities, which have been experimentally demonstrated in the literature of recent years.

Denial-of-service (DoS) attacks are a growing concern to networked services like the Internet. A DoS attack intends to deny access to shared services or resources by legitimate users [3]. A common form of WLAN DoS attack is a resource depletion attack, in which an attacker tries to overload

the Access Point's (AP) resources, such as its memory-hosted association table, rendering the AP unable to service honest clients. A potential way to deal with this problem is for a defending server to identify and segregate malicious traffic as quickly as possible. Other forms of DoS attack are jamming attacks, semantic attacks, and implementation specific attacks.

To deal with DoS attack, a number of methods have been proposed by researchers. Particularly for resource depletion or connection request flooding attack, in [4] a number of countermeasures both in the physical and MAC layers have been discussed. These solutions are cryptographic protection, security protocol repair, intrusion detection systems (IDS), decreasing the retry limit, identifying with signal strength info, and identifying through RF fingerprint.

Client puzzles, also known as proofs of work, have been shown to be a promising tool to thwart DoS attacks on network protocols, particularly on authentication protocols. A puzzle is issued by the server in reply to each request when the server is under attack. After receiving a puzzle, the client has to solve it in order to convince the server to grant

access to its resources. The main idea is that puzzle generation and solution verification should be easy for the server, while computing the puzzle solution should be somewhat hard, computationally speaking, for the client.

Many client puzzles have been proposed since they were first introduced by Dwork and Naor in 1992 [5]. An important recent development has been the analysis of client puzzles within the provable security framework [6, 7]. The main contribution of our paper is proposing a faster puzzle in verification and generation phases. Compared to other hash based puzzles, our work consumes at least 30% less resources in both generation and verification phases, all without the need for additional hardware. It also addresses the security definition of Chen et al. [6] and proves that our proposed puzzle is secure.

To emphasize the importance of CRF DoS attacks and to show how they are launched, we shall highlight parts of the 802.11 standard which present malicious users with opportunities to breach secure, fair, and efficient protocol operations.

The following section will describe how connection-request flooding attacks and spoofed disconnect attacks on 802.11 based networks occur. Section 3 will review the methods and approaches proposed by other researchers in the literature. However, more information related to other attacks and issues of network security can be found in [8–32]. More information related to patterns can be found in [33–36]. In Section 4, the details of the proposed puzzle will be discussed. The experimental results will be demonstrated in Section 5. Using security definition of Chen et al. [6] alongside the client puzzle protocol security properties, a security analysis of the proposed approach will be provided prior to the conclusion.

2. CRF DoS Attack on WLANs

Fundamentally, 802.11-based networks operate in two modes: Ad-hoc and Infrastructure. In addition, WLANs are deployed in three architectural models: Independent Basic Service Set (IBSS), Infrastructure BSS, and Extended BSS [37]. This paper will pay special attention to CRF DoS attacks on WLAN in Infrastructure mode, where a Basic Service Set (BSS) is orchestrated by the AP.

To get access eligibility, STAs must initially authenticate themselves to the AP. This is the same sort of operation as connecting a PC to a certain wired network outlet. In order to simplify the attachment of wireless stations (STA) to a network, the connection procedure in wireless networks has been designed without providing an authentication mechanism on MAC frame header fields, [38], particularly in open authentication mode. This security hole makes forging the source address of an MAC frame so easy that identifying the source of traffic is virtually impossible. Sending false connection requests is much cheaper than validating those requests. If the authentication server does not protect the limited-resource AP against false requests (whose aim is to exhaust available resources), the solution becomes challenging.

In addition, the authentication and association procedure has been designed as a stateful process. This means that AP is

required to allocate a certain amount of resources—normally memory capacity—for every connection request in order to track the current state, as shown in Figure 1.

Despite the significant benefits of the authentication and association procedure, there is a clear sign that this procedure could easily become a simple route to deny service [39]. An attacker can effortlessly launch a CRF DoS attack by forwarding a burst of bogus connection requests—whether probe, authentication, or association request frame—over a relatively short time, towards an unprotected AP. Consequently, the victimized AP runs out of resources (i.e. association table) quickly, so that legitimate requests remain unanswered [3, 40]. In addition, after making a wireless network disappear, a fake system belonging to an attacker may pose as the legitimate wireless infrastructure, which enables the attacker to launch a man-in-the-middle attack [41].

Even though a number of anti-CRF DoS schemes have already been proposed for wired networks, they are unsuitable to protect resource-limited wireless infrastructure [42].

3. Related Works

Over the past decades, a whole bunch of countermeasures have been proposed by researchers to mitigate or even eliminate the harmful effects of CRF DoS attacks on computer networks. In this section, we review the relevant literature on client puzzles, with an emphasis on hash-based puzzles.

Client Puzzle. The idea of client puzzle protocol is quite simple [43]. When server is not under attack, it follows its normal activity and accepts connection requests. When a server comes under attack, the server forwards a unique puzzle to each client wishing to attach to the network. Solving these puzzles imposes a certain amount of computation and storage cost on clients. If a client submits the valid solution to the server, the server will allocate the required resources to that client. As it turns out, a legitimate client has to undergo an insignificant computational cost when a server comes under attack, while an attacker needs large computational resources to make a noticeable interruption in the network service. The use of puzzles in cryptography was pioneered by Merkle [44], who used puzzles to establish a secret key between parties over an insecure channel. Client puzzles were first proposed by Dwork and Naor [5] as a countermeasure for email spam. The computational problems underlying most puzzles are either number-theoretic [5, 45, 46] or based on hash inversions [6, 47, 48]. Hash-based puzzles are very efficient—generation and verification typically requires only one or two hash function calls—but concrete realizations to date have been shown to be secure only in the random oracle model. Number-theoretic puzzles, on the other hand, have been shown to be secure in the standard model but have tended to be relatively inefficient, typically requiring the server to perform a large-integer modular exponentiation, making it unsuitable for resource-limited WLANs. Hence, some researchers focused on other payment schemes than CPU cycles, for example, memory-bound puzzles [49–51],

TABLE 1: DoS-resistant authentication based on extracting square roots.

Client	AP
	AP periodically decides D (difficulty), generates $n = p \times q$ (2 large prime numbers), sets a time limit Δt , and chooses L , R , and z Rand(z) $\in \{1, 2, \dots, w\}$, w is the size of n
Client solves X by brute force from equation $X^2 \equiv a \pmod{n}$ within Δt	
	AP verifies that R and L are recent AP computes $a' \equiv X^2 \pmod{n}$ AP verifies that a' satisfies the condition AP may now commit resources

knapsack puzzles [52], NST puzzles [53], bandwidth puzzles [54, 55], or human interaction with so-called CAPTCHAs [56].

Hash-Based Puzzle. Juels and Brainard (see [57]) employed a cryptographic client puzzle protocol to protect servers against CRF DoS attack.

Aura et al. [48] employed the client puzzle in an authentication protocol. They proposed a CPU-bound puzzle based on partial collision in a one-way hash function, protected by digital signature. The client solves X from the following equation by brute force and sends the signed solution X back to the server:

$$h(C, N_s, N_c, X) = \overbrace{000 \cdots 000}^{\text{The } k \text{ first bits of the hash}} Y. \quad (1)$$

The authors claimed that their method is a lightweight puzzle and consumes as little CPU-cycle as possible. However, using a public-key signature may increase significantly the likelihood of resource-exhaustion on the authenticator's side (AP or server), particularly when the network environment is WLAN [58]. Moreover, when using hash-based puzzles, some issues have to be taken into consideration, including the following:

- (i) The puzzle difficulty is not fine-grain controlled [59]. In other words, the difficulty grows exponentially while the number of bits needed for partial collision increases.
- (ii) Like other CPU-bound puzzles, this puzzle suffers from the CPU-power disparity issue [59]. CPU-power disparity issues occur where a puzzle depends on computational power. In such circumstances, clients equipped with high resources are able to solve puzzles in a shorter time than others. Attackers take full advantage of this security hole.
- (iii) This method is recognized as a parallelizable puzzle, so the attacker can solve the puzzle in a parallel manner using a number of employed clients.

To mitigate the overloading of Aura's puzzle on APs, Shi and Ma [60] designed a lightweight anti-DoS attack wireless authentication scheme on the basis of a hash function. They put a signature computation only after the puzzle verification.

Still, the AP has to generate a puzzle for every client's request and to store it. To handle a burst of connection requests containing spoofed or unauthenticated MAC addresses, the AP has to exhaust more resources to generate more puzzles. This may lead to memory exhaustion. Moreover, this puzzle still suffers from the hash reversal puzzle issues mentioned above.

To eliminate the Aura puzzle defect, Dong et al. [58] proposed a hash based puzzle to handle two key problems: avoiding the possible memory-exhaustion in the earlier puzzle and combating the CRF attack. They took advantage of beacon frames to distribute the puzzle parameters. This protects the AP against a second DoS attack on puzzle generation where AP faces a sudden burst of puzzle requests. However, this scheme does not provide a solution to the problems raised by hash-based puzzles. The authors in [61] proposed a CPU-bound client puzzle based on extracting square roots. Table 1 exposes the proposed client puzzle in detail.

By using beacon frames as the puzzle carrier, this puzzle keeps the AP away from a puzzle request flooding attack. Moreover, it is claimed that by choosing two large prime numbers p and q along with Δt , it is ensured that forging multiple solutions in a short time is impossible. Despite these valuable achievements, a number of drawbacks should be taken into consideration:

- (i) Using quadratic residue problems imposes a large computational load on an STA, which in the case of a PDA, tablet, or other limited-resource device can be a problem.
- (ii) Allocating 77 bytes of management frame to a puzzle wastes wireless bandwidth.
- (iii) Even though the verification is low cost, sending too many fake solutions can overburden the AP.
- (iv) The results show the puzzle difficulty of this puzzle is poorly granular.
- (v) Like other CPU-bound puzzles, this puzzle suffers from CPU-power disparity issue.

As the granularity of hash-based reversal puzzles is too coarse, Feng et al. [62] proposed the idea of a hint-based hash reversal puzzle to allow the granularity to be linear. The behind idea of the proposed puzzle is that the server gives

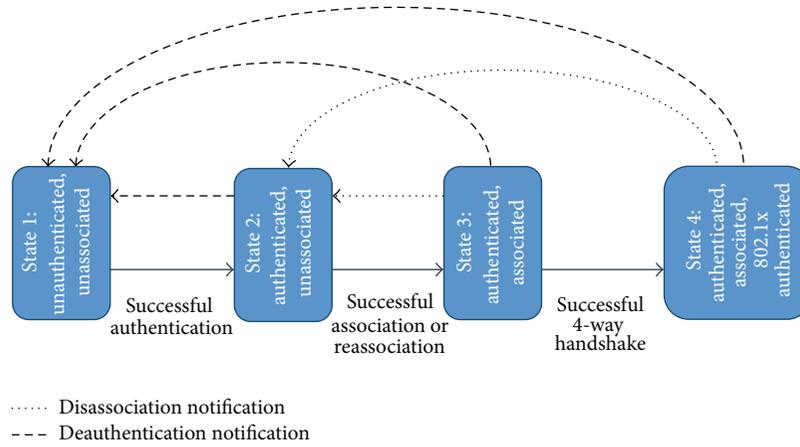


FIGURE 1: Authentication and association states.

the client a hint about the range within which the solution lies. For example, suppose a randomly generated number x is used as the input to the hash $h(x)$. To generate a puzzle with $O(D)$ difficulty, the server passes the client the hash and a hint, $x - u(0, D)$. Where $u(0, D)$ is a randomly chosen number uniformly distributed between 0 and D . Instead of checking every possible solution, the client starts at the hint and searches the range linearly for the answer. Apart from the last process, all remaining processes are similar to Jules and Brainard's puzzle. Hence, the simple puzzle generation and verification, as well as the linear granularity for fine grained control are the strengths of this solution. However, it is still susceptible to parallel processing attacks. Also it is suspicious of DoS puzzle attacks on the AP during puzzle-generation, where an attacker sends a burst of puzzle requests and the AP has to produce a unique puzzle for every request. Moreover, like other CPU-bound puzzle, a hint-based hash-reversal puzzle suffers from CPU speed disparity issue.

Similarly, Lei et al. [63] introduced the quasi-partial collision concept to provide a granularity of puzzle difficulty in a more controlled way. The results show a marked improvement on earlier puzzles; however, the hash-based CPU-bound problems still remain.

Number-Theoretic Puzzle. Rivest et al. [64] proposed a non-parallelizable time-lock puzzle based on repeated-squaring to enable time-release cryptography. The idea behind it is that a client has to allocate a certain amount of computational resource time to execute repeated squaring. First, a server calculates roughly how many squaring operations a client is able to run per second: S . Then it specifies the time required by a client to find the puzzle's solution: T . Considering this information, server computes how many times a client must run the squaring function to find the solution: $t = T \times S$. Finally, the server encrypts a message M into a cipher text C as follows: $C = M + X^{a^t} \bmod N$, given an integer X , an exponent a , a large integer t , and an appropriate semiprime modulus N . To acquire M from C , the client needs to compute $X^{a^t} \bmod N$ given X , a , t , and N in $\log(a^t) \approx t$ modular multiplications. This computation can be

performed efficiently using the trapdoor offered by Euler's function: $X^{a^t} \bmod N \equiv X^{a^t \bmod \varphi(N)} \bmod N$. Surprisingly, time-lock puzzles are able to provide a precise and fixed amount of work. To verify the puzzle, the server checks the solution through the trapdoor offered by Euler's function in $O(\log(N))$ modular multiplications.

Although this nonparallelizable puzzle eliminates the CPU-power disparity problem, there are some fundamental drawbacks which make this puzzle impractical in the real world.

- (i) Generating and storing big prime numbers may be resource-exhausting on the server side. As a result, time-lock puzzles meet the inefficiency threshold in puzzle generation in the order of microseconds [65].
- (ii) To verify the puzzle, the server has to allocate a significant amount of resources to compute $e = a^t \pmod{\varphi(N)}$ and $b = X^e \pmod{N}$ for every received puzzle solution, which is undesirable in WLANs.
- (iii) The server has to find out some information, such as CPU-power, from the clients in order to calculate the precise puzzle parameters. Attaining this information from a heterogeneous environment of clients, such as in WLANs, is almost impossible.

To reduce the costs of puzzle verification of Rivest's puzzle, Karame and Ćapkun [45] adapted Rivest's puzzle by employing an RSA key pair with small private exponent. This puzzle is based on the assumption that it is computationally intractable to compute a small private exponent d when the public exponent e is larger by several orders of magnitude than the modulus N . The server must still perform a modular exponentiation but, given a semiprime modulus N , the number of multiplications is decreased by a factor of $|N|/k$ where k is a security parameter. For example, a factor 12.8 for a 1024 bit modulus results in 120 modular multiplications instead of 1536.

In spite of this significant reduction in the verification and generation cost, still the verification costs remain too high to provide viable DoS protection for WLANs where

resources are much limited [66]. In particular, the puzzle setup is extremely costly. Also, this method suffers from difficult granularity issues. Furthermore, this puzzle wastes the network's bandwidth due to employing large numbers.

Recently, Rangasamy et al. [67] proposed a modular exponentiation-based client puzzle which can be seen as an efficient alternative to Rivest et al.'s time-lock puzzle. Unlike the Rivest et al. and Karame-Ćapkun puzzle, Rangasamy et al.'s puzzle does not require the server to perform any online exponentiations. In fact, the server has to perform a total of two hash operations and a few modular multiplications for the puzzle generation and verification. Although it is a significant improvement over the Karame-Ćapkun puzzle construction, the security of the puzzle does not rely on the standard security assumptions.

Kuppusamy et al. [68] also proposed a DLPuz puzzle based on the interval-discrete logarithm problem. This puzzle is claimed to be efficient while remaining secure in the standard model. Among other number theoretic puzzles, DLPuz puzzles demonstrate an extremely efficient verification algorithm, although the results show a costly puzzle generation which is prone to DoS attack.

4. PatPuz: A Superfast Hash-Based WLAN Puzzle

This section describes our new client puzzle construction PatPuz, which is based on finding a pattern key through reversing a one-way hash function. First, we will review the definition of a client puzzle and then present our construction.

The PatPuz will be able to manage connection requests so that they lose their importance as a valuable target for launching a DoS attack. Furthermore, this puzzle resists bogus puzzle solution flood attack where attackers strive to overload an AP during the puzzle verification stage.

Notation. If n is an integer, then we use $|n|$ to denote the length in bits of n . The puzzle makes use of a family of keyed hash function $\mathcal{H} := \{H_k\}_{k \in \mathcal{K}}$, where each H_k is a function mapping G to $\{0, 1\}^\ell$ and k is security parameter. We let $x \leftarrow_R S$ denote an uniformly random choosing of x from S where S is a set. We write $x \leftarrow \mathcal{A}(y)$ to assign the output of algorithm \mathcal{A} to x when run with the input y . If k is a security parameter, then $\text{negl}(k)$ denotes a function that is negligible in k , namely asymptotically smaller than the inverse of any polynomial in k . Let $P_{x,z}(R)$ be a pattern function and $V(p_n)$ denotes the p_n -th bit of R , then the following patterns are applied on R :

- (i) Extract n values of length ℓ from $h(x)$ as shown in Figure 4.
- (ii) R changes as follows:

If $z = 1$ then $V(p_{2n}) = V(p_{2n+1})$ otherwise $V(p_{2n}) \neq V(p_{2n+1})$.

If $x = x_0, x_1, \dots, x_i, \dots, x_j, \dots, x_n$ is a bit string then we let $x\langle i, j \rangle$ denote the substring x_i, \dots, x_j . We use $y = \sigma_{s,\alpha}(x)$ to denote $y = \sigma_s(x)\langle 1, \alpha \rangle$.

4.1. Client Puzzle Formal Definition. In [6], Chen et al. defines a client puzzle formally as follows.

Definition 1 (client puzzle). A client puzzle Puz is a tuple consisting of the following algorithms:

- (i) Setup(1^k): a p.p.t. setup algorithm that generates and returns a set of public parameters $params$ and a secret key s , the former of which includes a puzzle difficulty parameter space $QSpace$.
- (ii) GenPuz(s, Q, str): a p.p.t. puzzle generation algorithm which accepts a secret key s , difficulty parameter Q , and a session string str and returns a puzzle puz .
- (iii) FindSoln(puz, τ): a probabilistic puzzle solving algorithm that returns a potential solution $soln$ for puzzle puz after running time at most τ .
- (iv) VerAuth(s, puz): a d.p.t. puzzle authenticity verification algorithm that returns true or false.
- (v) VerSoln($s, str, puz, soln$): a d.p.t. puzzle solution verification algorithm that returns true or false.

For correctness, we require that if $(params; s) \leftarrow \text{Setup}(1^k)$ and $puz \leftarrow \text{GenPuz}(s, Q, str)$ then there exists $\tau \in \mathbb{N}$ such that $\text{VerSoln}(s, str, puz, soln)$ is true with probability 1 where $soln \leftarrow \text{FindSoln}(puz, \tau)$.

4.2. The PatPuz Puzzle. This scheme aims to provide a unique puzzle for every connection request received by an AP. The task of an STA is to find a pattern key through inverting a one-way hash function and to use that pattern key to create a unique pattern as the puzzle solution. Then, the AP only verifies those patterns. In many scenarios, it is essential that the GenPuz, VerAuth, and VerSoln algorithms be extremely efficient. In a denial-of-service setting, these algorithms are run online by the AP many times, and if they were expensive, then an attacker could induce a second resource depletion attack by asking for many puzzles to be generated or verified.

Setup(1^k). The various spaces are chosen; $sSpace \leftarrow \mathcal{K}$, $QSpace \leftarrow \mathbb{N}$, $strSpace \leftarrow \{0, 1\}^*$, $solnSpace \leftarrow \mathcal{X}$, $paternSpace \leftarrow \mathcal{P}$, and $puzSpace \leftarrow QSpace \times strSpace \times \{0, 1\}^k \times \mathcal{Y}$. The value s is chosen as $s \xleftarrow{\$} sSpace$ then output.

GenPuz(s, Q, str). A nonce as a timestamp for current cycle is selected $n_s \xleftarrow{\$} \{0, 1\}^k$. Let $str \leftarrow \text{MAC}_{AP}$. Next x is computed as $x \leftarrow F_{s,Q}(Q, n_s, str)$. The value $y \in \mathcal{Y}$ is computed as $y \leftarrow \varphi(x, n_s, Q)$ and the puzzle assigned to be $puz = (Q, str, n_s, y)$ and output.

FindSoln(puz, τ). One typical method for a legitimate client to implement the FindSoln algorithm is a brute-force search. While this algorithm is within the allowed number of clock cycles of execution, it randomly samples elements from the set of possible solutions without replacement and for each potential $x' \in \mathcal{X}$ computes $y' \leftarrow \varphi(x', n_s, Q)$. If $y' = y$ then it computes $R' \leftarrow \mathcal{P}_{x',z}(R)$ output R' then halts and

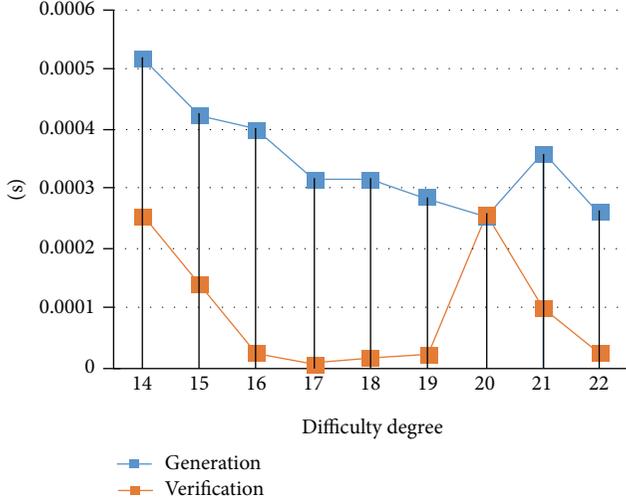


FIGURE 2: The curve relationship between difficulties and time-consuming on generating and verifying the puzzle.

otherwise continues with random sampling. If this algorithm reaches the last clock cycle of execution then it outputs an R' computed from a random element of the remaining unsampled preimage space.

$VerAuth(s, x, puz')$. For a puzzle $puz' = (Q', str', n'_s, x', y')$ this computes $x \leftarrow F_{s,Q}(Q', n'_s, str')$; then if $x = x'$ this output is true and otherwise the output is false.

$VerSoln(puz', R')$. Given a potential solution R' this checks if $R' \in \mathcal{P}_x$ and if so outputs are true and otherwise output is false.

Remark 2. In PatPuz, the AP has to store the short-term secrets x and P_x in order to speed up the verification phases. These secrets are expired when a cycle times out and the new values are replaced.

When the pattern sent by the STA passes the verification step, the AP will send an authentication response frame back to the STA and allocates the required resources.

5. Performance Experiment of PatPuz

NS2 simulates our scheme over a core i5 2.27 GHz/4G Ubuntu system. Since our goal is to balance the resource consumption between AP and STA, we focus on the time consumed on the puzzle generation and verification on the one hand and puzzle solving on the other. We will consider the puzzle difficulty Q to be less than 22 for simplicity reasons. The difficulty of a client puzzle gives a measure of the likelihood of an adversary finding a solution to a given puzzle within a given number of clock cycles of execution. Intuitively, the difficulty of solving a puzzle is ensured by the hardness of inverting the one-way function.

Figure 2 demonstrates the link between difficulties and time-consumption in both puzzle generation and solution verification. The obtained results are important for two

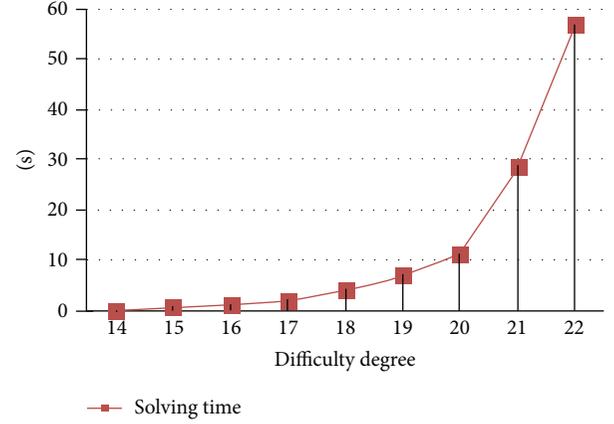


FIGURE 3: The curve relationship between difficulty degree and time-consuming on solving the puzzle.

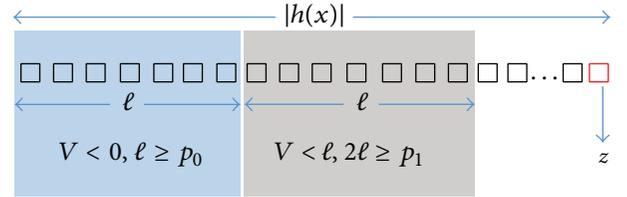


FIGURE 4: Extracting the values from $h(x)$ to calculate the correct positions for applying the pattern accordingly.

reasons. First of all, the time the AP spends to generate and verify a new puzzle is considerably shorter. It is less than 0.6 and 0.3 milliseconds in generation and verification procedure, respectively. Secondly, the fluctuation in generation and verification time is almost constant when the difficulty degree goes up or down. Therefore, increasing or decreasing the difficulty degree does not affect the time needed by the CPU for generating and verifying the puzzle.

Consequently, the proposed scheme eliminates any chance for an attacker to make puzzle generation and verification phases into a valuable target to launch DoS attacks by using fake solutions.

The most important impact of our scheme is to force STAs to cost their resources for every connection request. Therefore, the chance of attacker exhausting AP's resources by sending a burst of fake connection requests is very small. As shown in Figure 3, the solving time increases considerably, by nearly exponential growth, when difficulty of degree goes up.

6. Security Analysis of PatPuz

In this section, we analyse the PatPuz puzzle using the security model of Chen et al. [6]. Chen et al. introduced two security properties that a client puzzle should satisfy: unforgeability and difficulty. We shall give a brief description of these two properties. Intuitively, the unforgeability of PatPuz is ensured by the use of a pseudo-random function

and the difficulty of solving PatPuz puzzles is ensured by difficulty of inverting the one-way function.

6.1. Unforgeability. This experiment measures the ability of an adversary to produce a valid client puzzle and force a server to accept it as one that was not originally generated by a server in a probabilistic way. In general, unforgeability can easily be provided by using a message authentication code (MAC) or pseudo-random function to tag puzzles generated by the server, and this is what is done in PatPuz. First we review the formal definition of puzzle unforgeability in the next section. The results show that PatPuz is indeed unforgeable. We shall make use of a sequence of games [69] to prove the security properties.

Definition 3 (unforgeability [6]). Let k be a security parameter, \mathcal{A} a probabilistic algorithm, and Puz a client puzzle. Define the experiment $Exec_{\mathcal{A},Puz}^{UF}(k)$ as follows:

- (1) $(params, s) \leftarrow Setup(1^k)$.
- (2) Run $\mathcal{A}(params)$ with oracle access to $CreatePuz(\cdot)$ and $CheckPuz(\cdot)$, which are answered as follows:
 - (a) $CreatePuz(str, Q)$: $puz \leftarrow GenPuz(s, Q, str)$. Return Puz to \mathcal{A} .
 - (b) $CheckPuz(puz)$: if puz was not an output for any of the $CreatePuz(str)$ query made previously and $VerAuth(s, puz) = \text{true}$ then stop the experiment and output 1. Otherwise, return false to \mathcal{A} .
- (3) Output 0.

We say that \mathcal{A} wins the game if $Exec_{\mathcal{A},Puz}^{UF}(k) = 1$ and loses otherwise. The advantage of \mathcal{A} is defined as:

$$Adv_{\mathcal{A},Puz}^{UF}(k) = \Pr(Exec_{\mathcal{A},Puz}^{UF}(k) = 1). \quad (2)$$

A puzzle is said to be unforgeable if this advantage is negligible in k for probabilistic algorithms \mathcal{A} running in time polynomial in k .

In this unforgeability experiment, the adversary is allowed to query the $CreatePuz$ oracle by choosing puzzle difficulty level Q at will. This is to ensure that even after seeing puzzles with different difficulty levels, the adversary cannot create a valid looking puzzle.

Theorem 4 (unforgeability of PatPuz puzzle). *The PatPuz puzzle is unforgeable.*

Proof. We prove the theorem using a sequence of games. Let \mathcal{A} be a probabilistic algorithm with running time t . Let S_i be the event that \mathcal{A} wins in game G_i .

Game G_0 . Let G_0 be the original unforgeability game $Exp_{\mathcal{A},PatPuz}^{UF}(k)$. Then

$$\Pr(Exec_{\mathcal{A},OROD}^{UF}(k') = 1) = \Pr(S_0). \quad (3)$$

Game G_1 . In this game, we modify game G_0 by replacing the F_s with a truly random function \mathcal{R} to compute the pattern key x . This change has a negligible effect on adversary \mathcal{A} because of the pseudo-randomness of F_s . Hence,

$$|\Pr(S_0) - \Pr(S_1)| \leq Adv_{\mathcal{B}}^{F_s}(k) \leq \text{negl}(k), \quad (4)$$

where \mathcal{B} is an algorithm running in time $O(t)$, and the second inequality follows whenever F_s is a pseudo-random function.

Since the function \mathcal{R} in game 1 is truly random, the probability that an adversary without access to \mathcal{R} can guess an output is negligible:

$$\Pr(S_1) \leq \frac{1}{2^k}. \quad (5)$$

Combining (3) through (5), we obtain the final result that the adversary's success in forging a puzzle is negligible. \square

6.2. Difficulty. To prove our puzzle difficulty, we shall mainly focus on generating a valid pattern without having to find. Like the previous property, we shall first define the formal form of the puzzle difficulty presented by Chen et al., then we will show that our puzzle is a difficult puzzle.

Definition 5 (puzzle difficulty). Let k be a security parameter and let Q be a difficulty parameter which is kept fixed through the experiment. Let \mathcal{A} be a probabilistic algorithm and puz be a client puzzle. The game $Exec_{\mathcal{A},Puz}^{Q,DIFF}(k)$ is defined for each hardness parameter $Q \in \mathbb{N}$ as follows:

- (1) $(params, s) \leftarrow Setup(1^k)$
- (2) Run $\mathcal{A}(params)$ with oracle access to $CreatePuzSoln(\cdot)$ and $Test(\cdot)$, which are answered as follows:
 - (a) $CreatePuzSoln(str)$: $puz \leftarrow GenPuz(s, Q, str)$. Find a solution $soln$ such as $VerSoln(puz, soln) = \text{true}$. Return $(puz, soln)$ to \mathcal{A} .
 - (b) $Test(str^*)$: This query may ask once, at any point during the game. The challenger generates a puzzle $puz^* \leftarrow GenPuz(s, Q, str)$ and returns puz^* to \mathcal{A} . Then \mathcal{A} may continue to ask $CreatePuzSoln$ queries.
- (3) \mathcal{A} output a potential solution $soln^*$.
- (4) Output 1 if $VerSoln(puz^*, soln^*) = \text{true}$ and 0 otherwise.

We say that \mathcal{A} wins the game if $Exec_{\mathcal{A},Puz}^{Q,DIFF}(k) = 1$ and loses otherwise. We define the success of an adversary \mathcal{A} against puz as

$$Succ_{\mathcal{A},Puz}^{Q,DIFF}(k) = \Pr[Exec_{\mathcal{A},Puz}^{Q,DIFF}(k) = 1]. \quad (6)$$

Let $\epsilon_{k,Q}(t)$ be a family of functions monotonically increasing in t . A puzzle is $\epsilon_{k,Q}(\cdot)$ -difficult if, for all probabilistic algorithm \mathcal{A} running in time at most t ,

$$Succ_{\mathcal{A},Puz}^{Q,DIFF}(k) \leq \epsilon_{k,Q}(t). \quad (7)$$

Theorem 6 (difficulty of PatPuz puzzle). *Let k be a security parameter and let Q be a difficulty parameter. Let $\mathcal{H} := \{H_k\}_{k \in K}$ be a family function of keyed hash function, where each H_k is a function mapping $\{0, 1\}^{\ell_1}$ to $\{0, 1\}^{\ell_2}$, and let \mathcal{F} be a pseudo-random function. Then PatPuz puzzle is $\epsilon_{k,Q}(t)$ -difficult for all probabilistic algorithms \mathcal{A} running in time at most τ .*

Proof. Like previous properties, we shall employ a sequence of games to prove our theorem. In game G_0 the adversary tries to break our construction whereas in game G_1 the adversary works against an idealized version of our construction, where random function \mathcal{F} has been replaced with a truly random function. Let \mathcal{A} be a probabilistic algorithm with running time τ . Let S_i be the event that \mathcal{A} wins the game G_i .

Game G_0 . This game represents the original difficulty game $Exec_{\mathcal{A}, PatPuz}^{Q, DIFF}(k)$. For clarity, we shall write the full definition of this game:

- (1) The challenger \mathcal{C} first runs the Setup on 1^k and obtains $s \xleftarrow{\$} sSpace$. s is kept secret.
- (2) The adversary \mathcal{A} will then starts to ask CreatePuzSoln() queries. To answer each \mathcal{C} select a random nonce $s \xleftarrow{\$} \{0, 1\}^k$ and a random number $z \xleftarrow{\$} \{0, 1\}^*$. Then computes x through $x \leftarrow F_{s,Q}(Q, n_s, str)$. Next \mathcal{C} computes $y \leftarrow \varphi(x, n_s, Q)$ and $R' \leftarrow \mathcal{P}_{x',z}(R)$ then returns $puz \leftarrow (Q, str, n_s, y)$ and $soln \leftarrow R'$ to \mathcal{A} .
- (3) At any time during the game, \mathcal{A} is allowed to issue a Test() query for which \mathcal{C} generates a puzzle $paz^* = (Q, str, n_s^*, y^*)$ and returns paz^* to \mathcal{A} .
- (4) The adversary \mathcal{A} may continue to ask CreatePuzSoln() queries which the challenger \mathcal{C} answers as before.
- (5) Eventually, after τ clock cycles \mathcal{A} outputs a potential solution $soln^* = R^*$. If $VerSoln(puz^*, R^*) = \text{true}$, the adversary \mathcal{A} wins and then the challenger \mathcal{C} outputs 1 and terminates and otherwise outputs 0 and terminates.

Hence,

$$\Pr(Exec_{\mathcal{A}, PatPuz}^{Q, DIFF}(k) = 1) = \Pr(S_0). \quad (8)$$

Game G_1 . Now we transform game G_0 into game G_1 , replacing the pseudo-random function F with a truly random function \mathcal{R} . This change is indistinguishable due to the pseudo-randomness of F , so

$$|\Pr(S_0) - \Pr(S_1)| \leq \text{negl}(k). \quad (9)$$

Game G_2 . We now transform game G_1 into game G_2 , replacing the hash function $\varphi(\cdot)$ with a randomly chosen one, $\varphi^\dagger(\cdot): h \xleftarrow{\$} \{0, 1\}^\ell$. We assume that the family function of hash

function \mathcal{H} is entropy smoothing. This means that it is hard to distinguish $(k, H_k(\delta))$ from (k, h) , where k is a random element of K , δ is a random element of G , and h is a random element of $\{0, 1\}^\ell$. Algorithmically, game G_2 looks like this:

- (1) The challenger \mathcal{C} first runs the Setup on 1^k and obtains $s \xleftarrow{\$} sSpace$. s is kept secret.
- (2) The adversary \mathcal{A} will then start to ask CreatePuzSoln() queries. \mathcal{C} answers to queries as follows:
 - (a) CreatePuzSoln(): as in game G_1 except replacing φ^\dagger to compute y .
- (3) At any time during the game, \mathcal{A} is allowed to issue a Test() query for which \mathcal{C} do the same as G_1 except replacing φ^\dagger to compute y .
- (4) The adversary \mathcal{A} may continue to ask CreatePuzSoln() queries which the challenger \mathcal{C} answers as before.
- (5) Eventually, after τ clock cycles \mathcal{A} outputs a potential solution $soln^* = R^*$. If $VerSoln(puz^*, R^*) = \text{true}$, the adversary \mathcal{A} wins and then the challenger \mathcal{C} outputs 1 and terminates and otherwise outputs 0 and terminates.

In the above game, a truly random function \mathcal{R} is input to the function φ^\dagger to compute y . As a result the only way \mathcal{A} could find a correct solution R' would be for \mathcal{A} to invert the function φ^\dagger . Hence we have

$$\Pr(S_2) \leq Adv_{A,H}^{OWF} \leq \frac{1}{2^{n \times \ell \times |z|}}, \quad (10)$$

where S_2 is the event that \mathcal{A} wins the game G_2 , n is the number of pattern points, ℓ is the length of pattern points such that $2^\ell = |h(x)|$, and $Adv_{A,H}^{OWF}$ is defined as below. The second inequality follows the fact that the only way a protected (by our scheme) AP may be attacked is to reveal the pattern. An attacker can reach the correct pattern either through solving the puzzle or guessing the pattern. If he or she chooses the first way, the main goal of our scheme is satisfied. The probability of finding a correct R' is $1/2^{n \times \ell \times |z|}$ in which in case of $n = 6$, $\ell = 7$, and $|z| = 2$ the result would be $1/2^{43}$ which is negligible.

Definition 7. For an adversary \mathcal{A} we define its advantage against a function $\psi: \mathcal{X} \mapsto \mathcal{Y}$, where \mathcal{X} is fixed and finite, in terms of OWF as

$$\begin{aligned} Adv_{\mathcal{A}, \psi}^{OWF} &= \Pr x \left[x \xleftarrow{\$} \mathcal{X}; y \leftarrow \psi(x); \right. \\ &\quad \left. (\tilde{x} \leftarrow \mathcal{A}(y) \wedge \psi(\tilde{x}) = y) \right]. \end{aligned} \quad (11)$$

Let $\epsilon_i: \mathbb{N} \mapsto [0, 1]$ be a monotonically increasing function. Then the function ψ is an $\epsilon_i(\cdot)$ -OWF if for all adversaries \mathcal{A} it holds that $Adv_{\mathcal{A}, \psi}^{OWF} \leq \epsilon_i(\tau)$.

We also claim that

$$|\Pr(S_1) - \Pr(S_2)| = \epsilon_h(\tau), \quad (12)$$

where ϵ_h is the h-advantage of some efficient algorithm (which is negligible assuming \mathcal{H} is entropy smoothing).

To prove this claim, any difference between $\Pr(S_1)$ and $\Pr(S_2)$ can be parlayed into a corresponding h-advantage. The following algorithm D interpolates between game G_1 and game G_2 and so has h-advantage equal to $|\Pr(S_1) - \Pr(S_2)|$.

Algorithm D (k, h)

- (1) \mathcal{C} first runs the Setup on $1^k: s \xleftarrow{\$} sSpace$
- (2) Run \mathcal{A} with oracle access to CreatePuzSoln() and Test(), which are answered as follows:
 - (a) CreatePuzSoln():
 - (i) $n_s \xleftarrow{\$} \{0, 1\}^k$.
 - (ii) $R \xleftarrow{\$} \{0, 1\}^*$
 - (iii) $x \xleftarrow{\$} \{0, 1\}_Q^*$
 - (iv) compute y
 - (v) $R' \leftarrow P_{x,z}(R)$
 - (vi) Return $(puz, soln) \leftarrow (y, R')$
 - (b) Test(): $puz^* \leftarrow y^*$ and return puz^*
- (3) \mathcal{A} outputs a potential solution $soln^*$
- (4) \mathcal{A} may continue to ask CreatePuzSoln() queries which are answered as before.
- (5) Output 1 if $\text{VerSoln}(puz^*, soln^*) = \text{true}$ and 0 otherwise.

Based on this indistinguishability assumption, $|\Pr(S_1) - \Pr(S_2)|$ is negligible.

Combining (8) through (12) yields the desired result. \square

6.3. Security Properties. In [70, 71], some general criteria are listed to specify the properties a puzzle must meet to be considered as an effective and efficient anti-DoS approach. Simply put, these criteria prove how secure and powerful the proposed client puzzle protocol is. Here we review these properties with an emphasis on PatPuz puzzle.

Computation Guarantee. Since the hash function is considered to resist preimage and collision attack, the only way to solve a puzzle is to use a brute force method. Hence, STAs have to look up a range of 2^k possible solutions to find out the right pattern. Even though this range may be reduced to $2^{k/2}$ possible solutions due to a birthday attack [72], the client (and also attacker) still has to spend enough time to find the puzzle's solution.

Adjustability of Difficulty. In our scheme, the AP adjusts the difficulty degree by increasing or decreasing Q . Note that these variations have no effect on the time spent to generate or verify the puzzle.

Efficiency. Since the puzzle verification is done only by looking for a correct pattern in a puzzle solution R' —a

significantly low computational process—the proposed client puzzle protocol resists a puzzle verification attack where an attacker forwards too many bogus puzzle solutions. In addition, a protected AP is required to store only a long-term secret value s , to verify the received solutions. To make the verification phase more efficient, a short-term value, which is a Q -bit length, can be stored by AP.

Correlation Free. The PatPuz puzzle is correlation-free. That means knowing all previous puzzle solutions does not help to solve the current puzzle in any way. This property is upheld by the unpredictability feature of the random numbers generated in puzzle generation stage (s) and puzzle solving (R).

Stateless. The proposed client puzzle protocol does not require storing any client's or puzzle related information, except s , and x . Moreover, the memory allocated to x is cleared after changing the puzzle, meaning that the algorithm utilizes a fixed-size memory to handle the puzzle. Hence, a protected AP in our scheme will face no memory shortage in a relatively short time.

Tamper-Resistance. No STA (attacker) is able to learn x by examining the other STAs' solutions. Every STA, in our scheme, is required to apply the produced pattern over its own R which is basically random.

7. Conclusion and Future Work

The main consideration for implementing any security protocol in 802.11 based networks is how much cost the proposed algorithm imposes on both the CPU and memory to complete its tasks. This paper proposes a novel puzzle to meet the AP's constraints and protect wireless network against CRF DoS attacks.

The following items are satisfied by the proposed scheme:

- (i) **Low-cost generation.** The CPU load in our proposed scheme is very low during initial setup and puzzle generation. The simulation output shows that this phase takes less than 6 milliseconds.
- (ii) **Low-cost verification.** Like the generation phase, our proposed verification load is very low on the AP's CPU. Hence, the simulation output shows that the time an AP has to spend to accomplish the verification phase is less than 0.4 milliseconds.
- (iii) **Antireassign DoS attack.** Since both generation and verification in our scheme are very cost-effective, the proposed scheme eliminates a second DoS attack on the AP which can be posed by attack-prone security puzzles.
- (iv) The memory usage of PatPaz puzzle is fixed and very small, so much so as to be almost negligible. Therefore, the proposed puzzle will never suffer from memory exhaustion.
- (v) The proposed scheme also defines T_{exp} to limit the puzzle's solution life. Before T_{exp} expires, all received

puzzle solutions are discarded. Hence, launching an effective DoS attack becomes more challenging.

7.1. Future Work. This paper proposes a lightweight method based on a cryptographic client puzzle. Cryptographic puzzles are very low-cost in puzzle generation and verification; however, they pose some problems. First of all, they are naturally solved within a probabilistic time. Secondly, the puzzle may be solved through parallelization. Thirdly, varying the difficulty level in this approach is too coarse. That means the difficulty of solving an $n - 1$ bit puzzle is two times less than solving an n bit puzzle. Thus it is very troublesome to design and implement an appropriate difficulty degree which increases efficiency as much as possible.

Future work can be focused on designing a new puzzle which mitigates (or even eliminates) the aforementioned problems of cryptographic client puzzles. A future study can also focus on finding a smarter mechanism to perceive and apprehend DoS attacks, in order to adjust the puzzle difficulty efficiently.

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 express greatest appreciation to Universiti Teknologi Malaysia (UTM) for financial support of the research done in Advanced Informatics School (AIS).

References

- [1] M. Nasreldin, H. Aslan, M. El-Hennawy, and A. El-Hennawy, "WiMax security," in *Proceedings of the 22nd International Conference on Advanced Information Networking and Applications Workshops/Symposia (AINA '08)*, pp. 1335–1340, March 2008.
- [2] P. H. Yu and U. W. Pooch, "A secure dynamic cryptographic and encryption protocol for wireless networks," in *Proceedings of the EUROCON 2009 (EUROCON '09)*, IEEE, St.-Petersburg, Russia, 2009.
- [3] K. Bicakci and B. Tavli, "Denial-of-Service attacks and countermeasures in IEEE 802.11 wireless networks," *Computer Standards & Interfaces*, vol. 31, no. 5, pp. 931–941, 2009.
- [4] K. Bicakci and B. Tavli, "Denial-of-Service attacks and countermeasures in IEEE 802.11 wireless networks," *Computer Standards and Interfaces*, vol. 31, no. 5, pp. 931–941, 2009.
- [5] C. Dwork and M. Naor, *Pricing via Processing or Combatting Junk Mail*, Springer, Berlin, Germany, 1992.
- [6] L. Chen, P. Morrissey, and N. Smart, "Security notions and generic constructions for client puzzles," in *Proceedings of the 15th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT '09)*, 2009.
- [7] D. Stebila, L. Kuppusamy, J. Ranganamy, C. Boyd, and J. Gonzalez Nieto, "Stronger difficulty notions for client puzzles and denial-of-service-resistant protocols," in *Topics in cryptology—CT-RSA 2011*, vol. 6558 of *Lecture Notes in Computer Science*, pp. 284–301, Springer, Heidelberg, Germany, 2011.
- [8] M. Alizadeh, W. H. Hassan, M. Zamani, S. Karamizadeh, and E. Ghazizadeh, "Implementation and evaluation of lightweight encryption algorithms suitable for RFID," *Journal of Next Generation Information Technology*, vol. 4, no. 1, pp. 65–77, 2013.
- [9] M. Alizadeh, J. Shayan, M. Zamani, and T. Khodadadi, "Code analysis of lightweight encryption algorithms using in RFID systems to improve cipher performance," in *Proceedings of the IEEE Conference on Open Systems (ICOS '12)*, Kuala Lumpur, Malaysia, October 2012.
- [10] E. Amiri, M. Alizadeh, H. Keshavarz, M. Zamani, and T. Khodadadi, "Energy efficient routing in wireless sensor networks based on fuzzy ant colony optimization," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 768936, 17 pages, 2014.
- [11] M. Gharooni, M. Zamani, M. Mansourizadeh, and S. Abdullah, "A confidential RFID model to prevent unauthorized access," in *Proceedings of the 3rd International Conference on Information Science and Engineering*, Yangzhou, China, 2011.
- [12] E. Ghazizadeh, M. Zamani, J.-L. Ab Manan, and M. Alizadeh, "Trusted computing strengthens cloud authentication," *The Scientific World Journal*, vol. 2014, Article ID 260187, 17 pages, 2014.
- [13] E. Ghazizadeh, M. Zamani, J.-L. Ab Manan, and A. Pashang, "A survey on security issues of federated identity in the cloud computing," in *Proceedings of the 4th IEEE International Conference on Cloud Computing Technology and Science (CloudCom '12)*, pp. 562–565, Taipei, Taiwan, December 2012.
- [14] H. Shohreh, M. Zamani, and H. Roza, "Dynamic monitoring in ad hoc network," *Applied Mechanics and Materials*, vol. 229–231, pp. 1481–1486, 2012.
- [15] M. Janbeglou, M. Zamani, and S. Ibrahim, "Redirecting network traffic toward a fake DNS server on a LAN," in *Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT '10)*, pp. 429–433, IEEE, Chengdu, China, July 2010.
- [16] M. Janbeglou, M. Zamani, and S. Ibrahim, "Redirecting outgoing DNS requests toward a fake DNS server in a LAN," in *Proceedings of the IEEE International Conference on Software Engineering and Service Sciences (ICSESS '10)*, pp. 29–32, Beijing, China, July 2010.
- [17] M. Janbeglou, M. Zamani, and S. Ibrahim, "Improving the security of protected wireless internet access from insider attacks," *Advances in Information Sciences and Service Sciences*, vol. 4, no. 12, pp. 170–181, 2012.
- [18] F. Kiani, E. Amiri, M. Zamani, T. Khodadadi, and A. A. Manaf, "Efficient intelligent energy routing protocol in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2014, Article ID 618072, 13 pages, 2014.
- [19] T. K. Araghi, M. Zamani, and A. B. T. Abdul Mnaf, "Performance analysis in reactive routing protocols in wireless mobile Ad Hoc networks using DSR, AODV and AOMDV," in *Proceedings of the International Conference on Informatics and Creative Multimedia (ICICM '13)*, pp. 81–84, Kuala Lumpur, Malaysia, September 2013.
- [20] T. Koohpayeh Araghi, M. Zamani, A. Abdul Manaf, and S. Kohpayeh Araghi, "An access control framework in an Ad Hoc network infrastructure," in *Proceedings of the 1st International Conference on Communication and Computer Engineering*, Malacca, Malaysia, 2014.
- [21] K. Mohebbi, S. Ibrahim, M. Zamani, and M. Khezrian, "UltiMatch-NL: a Web service matchmaker based on multiple semantic filters," *PLoS ONE*, vol. 9, no. 8, pp. 1–21, 2014.

- [22] S. Nikbakhsh, A. B. A. Manaf, M. Zamani, and M. Janbeglou, "A novel approach for rogue access point detection on the client-side," in *Proceedings of the 26th IEEE International Conference on Advanced Information Networking and Applications*, pp. 684–687, Fukuoka, Japan, March 2012.
- [23] H. R. Zeidanloo, A. B. Manaf, P. Vahdani, F. Tabatabaei, and M. Zamani, "Botnet detection based on traffic monitoring," in *Proceedings of the International Conference on Networking and Information Technology (ICNIT '10)*, pp. 97–101, Manila, Philippines, June 2010.
- [24] H. R. Zeidanloo, M. J. Zadeh, P. V. Amoli, M. Safari, and M. Zamani, "A taxonomy of Botnet detection techniques," in *Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT '10)*, pp. 158–162, Chengdu, China, July 2010.
- [25] A. Sadeghian and M. Zamani, "Detecting and preventing DDOS attacks in Botnets by the help of self triggered black holes," in *Proceedings of the Asia-Pacific Conference on Computer Aided System Engineering*, Bali, Indonesia, 2014.
- [26] A. Sadeghian, M. Zamani, and A. Abdul Manaf, "A taxonomy of SQL injection detection and prevention techniques," in *Proceedings of the International Conference on Informatics and Creative Multimedia (ICICM '13)*, pp. 53–56, Kuala Lumpur, Malaysia, September 2013.
- [27] A. Sadeghian, M. Zamani, and S. M. Abdullah, "A taxonomy of SQL injection attacks," in *Proceedings of the International Conference on Informatics and Creative Multimedia (ICICM '13)*, pp. 269–273, Kuala Lumpur, Malaysia, September 2013.
- [28] A. Sadeghian, M. Zamani, and S. Ibrahim, "SQL injection is still alive: a study on SQL injection signature evasion techniques," in *Proceedings of the International Conference on Informatics and Creative Multimedia*, pp. 265–268, Kuala Lumpur, Malaysia, September 2013.
- [29] A. Sadeghian, M. Zamani, and A. A. Manaf, "SQL injection vulnerability general patch using header sanitization," in *Proceedings of the International Conference on Computer, Communication, and Control Technology*, Langkawi, Malaysia, 2014.
- [30] A. Sadeghian, M. Zamani, and B. Shanmugam, "Security threats in online social networks," in *Proceedings of the International Conference on Informatics and Creative Multimedia (ICICM '13)*, pp. 254–258, Kuala Lumpur, Malaysia, September 2013.
- [31] A. Hematian, S. Chuprat, A. A. Manaf, and N. Parsazadeh, "Zero-delay FPGA-based odd-even sorting network," in *Proceedings of the IEEE Symposium on Computers and Informatics (ISCI '13)*, pp. 128–131, April 2013.
- [32] Z. A. Davani and A. A. Manaf, "Enhancing key management of ZigBee network by steganography method," in *Proceedings of the 2nd International Conference on Informatics and Applications (ICIA '13)*, pp. 77–81, IEEE, Łódź, Poland, September 2013.
- [33] Y. Zhang, S. Wang, G. Ji, and Z. Dong, "Genetic pattern search and its application to brain image classification," *Mathematical Problems in Engineering*, vol. 2013, Article ID 580876, 8 pages, 2013.
- [34] 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.
- [35] Y. Zhang, L. Wu, Y. Huoc, and S. Wang, "A novel global optimization method- Genetic pattern search," *Applied Mechanics and Materials*, vol. 44–47, pp. 3240–3244, 2011.
- [36] Y. Zhang and L. Wu, "Pattern recognition via PCNN and Tsallis entropy," *Sensors*, vol. 8, no. 11, pp. 7518–7529, 2008.
- [37] M. Gast, *802.11 Wireless Networks the Definitive Guide*, O'Reilly, Sebastopol, Calif, USA, 2005.
- [38] H. Ordi, B. Mousavi, B. Shanmugam, M. R. Abbasy, and M. R. N. Torkaman, "A novel proof of work model based on pattern matching to prevent DoS attack," in *Communications in Computer and Information Science*, vol. 166, pp. 508–520, Springer, Berlin, Germany, 2011.
- [39] J. Bellardo and S. Savage, "802.11 denial-of-service attacks: real vulnerabilities and practical solutions," in *Proceedings of the 12th Conference on USENIX Security Symposium (SSYM '03)*, Washington, DC, USA, 2003.
- [40] C.-H. Liu and Y.-Z. Huang, "The analysis for DoS and DDOS attacks of WLAN," in *Proceedings of the 2nd International Conference on MultiMedia and Information Technology (MMIT '10)*, pp. 108–111, April 2010.
- [41] B. Thapa, *Robust Wireless Communication in Adversarial Settings [doctoral dissertation]*, Northeastern University, Boston, Mass, USA, 2011.
- [42] Q. Dong, L. Gao, and X. Li, "A new client-puzzle based DoS-resistant scheme of IEEE 802.11i wireless authentication protocol," in *Proceedings of the 3rd International Conference on BioMedical Engineering and Informatics (BMEI '10)*, pp. 2712–2716, October 2010.
- [43] L. Chibiao, X. Chugui, Q. Jinming, and L. Changjing, "Experimental and theoretical study of authentication request flooding attack on 802.11 WLAN," *Energy Procedia*, vol. 13, pp. 6800–6808, 2011.
- [44] R. C. Merkle, "Secure communications over insecure channels," *Communications of the ACM*, vol. 21, no. 4, pp. 294–299, 1978.
- [45] G. O. Karame and S. Čapkun, *Low-Cost Client Puzzles Based on Modular Exponentiation*, vol. 6345 of *Computer Security*, Springer, 2010.
- [46] B. Waters, A. Juels, J. A. Halderman, and E. W. Felten, "New client puzzle outsourcing techniques for DoS resistance," in *Proceedings of the 11th ACM Conference on Computer and Communications Security (CCS '04)*, pp. 246–256, Washington, DC, USA, October 2004.
- [47] Back, "Hashcash—a denial of service counter-measure," Tech. Rep., 2002.
- [48] T. Aura, P. Nikander, and J. Leiwo, "DOS-resistant authentication with client puzzles," in *Security Protocols*, vol. 2133 of *Lecture Notes in Computer Science*, pp. 170–177, Springer, Berlin, Germany, 2001.
- [49] D. S. H. Rosenthal, "On the cost distribution of a memory bound function," *Computing Research Repository (CoRR)*, vol. cs.CR/0311005, pp. 1–7, 2003.
- [50] C. Dwork, A. Goldberg, and M. Naor, "On memory-bound functions for fighting spam," in *Advances in Cryptology—CRYPTO 2003*, vol. 2729 of *Lecture Notes in Computer Science*, pp. 426–444, Springer, Santa Barbara, Calif, USA, 2003.
- [51] S. Doshi, F. Monrose, and A. D. Rubin, "Efficient memory bound puzzles using pattern databases," *Applied Cryptography and Network Security*, vol. 3989, pp. 98–113, 2006.
- [52] S. Tritilanunt, C. Boyd, E. Foo, and J. M. G. Nieto, "Toward non-parallelizable client puzzles," in *Cryptology and Network Security*, vol. 4856 of *Lecture Notes in Computer Science*, pp. 247–264, Springer, Berlin, Germany, 2007.
- [53] I. Martinovic, F. A. Zdarsky, M. Wilhelm, C. Wegmann, and J. B. Schmitt, "Wireless client puzzles in IEEE 802.11 networks: security by wireless," in *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec '08)*, pp. 36–45, April 2008.

- [54] Y. I. Jerschow, B. Scheuermann, and M. Mauve, "Counterflooding: DoS protection for public key handshakes in LANs," in *Proceedings of the 5th International Conference on Networking and Services (ICNS '09)*, pp. 376–382, April 2009.
- [55] M. Walfish, M. Vutukuru, H. Balakrishnan, and D. Karger, "DDoS defense by offense," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, New York, NY, USA, 2006.
- [56] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, "CAPTCHA: using hard AI problems for security," in *Advances in Cryptology—EUROCRYPT 2003*, vol. 2656 of *Lecture Notes in Computer Science*, pp. 294–311, Springer, Berlin, Germany, 2003.
- [57] Juels and J. Brainard, "Client Puzzle," EMC, 1999, http://www.rsa.com/rsalabs/staff/bios/ajuels/publications/client-puzzles/Client_puzzles.ppt.
- [58] Q. Dong, L. Gao, and X. Li, "A new client-puzzle based DoS-resistant scheme of IEEE 802.11i wireless authentication protocol," in *Proceedings of the 3rd International Conference on Biomedical Engineering and Informatics (BMEI '10)*, pp. 2712–2716, October 2010.
- [59] Q. Tang and A. Jeckmans, *On Non-Parallelizable Deterministic Client Puzzle Scheme with Batch Verification Modes*, Centre for Telematics and Information Technology, University of Twente, 2010.
- [60] T.-J. Shi and J.-F. Ma, "Design and analysis of a wireless authentication protocol against DoS attacks based on Hash function," in *Aerospace Electronics Information Engineering and Control*, 2006.
- [61] Q. Dong, L. Li, and X. Li, "Quadratic residue based client puzzle distributed by beacon frame in dos-resistant wireless access authentication," *Advances in Information Sciences and Service Sciences*, vol. 3, no. 11, pp. 79–86, 2011.
- [62] W.-C. Feng, E. Kaiser, and A. Luu, "The design and implementation of network puzzles," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, pp. 2372–2382, Miami, Fla, USA, March 2005.
- [63] Y. Lei, S. Pierre, and A. Quintero, "Client puzzles based on quasi partial collisions against DoS attacks in UMTS," in *Proceedings of the IEEE 64th Vehicular Technology Conference (VTC '06)*, pp. 1–5, IEEE, Montreal, Canada, September 2006.
- [64] R. L. Rivest, A. Shamir, and D. A. Wagner, "Time-lock puzzles and timed-release crypto," Tech. Rep. MIT/LCS/TR-684, MIT Laboratory for Computer Science, Cambridge, Mass, USA, 1996.
- [65] D. Hofheinz and D. Unruh, "Simulatable security and polynomially bounded concurrent composability," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 169–182, Berkeley/Oakland, Calif, USA, May 2006.
- [66] Y. I. Jerschow and M. Mauve, "Secure client puzzles based on random beacons," in *NETWORKING 2012*, vol. 7290 of *Lecture Notes in Computer Science*, pp. 184–197, Springer, Berlin, Germany, 2012.
- [67] J. Rangasamy, D. Stebila, C. Boyd, and J. G. Nieto, "Efficient modular exponentiation-based puzzles for denial-of-service protection," in *Proceedings of the International Conference on Information Security and Cryptology (ICISC '11)*, 2011.
- [68] L. Kuppusamy, J. Rangasamy, D. Stebila, C. Boyd, and J. G. Nieto, "Practical client puzzles in the standard model," in *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security (ASIACCS '12)*, pp. 42–43, May 2012.
- [69] V. Shoup, "Sequence of games: a tool for taming complexity in security proofs," Tech. Rep. 2004/332, IACR Cryptology ePrint Archive, 2004.
- [70] M. Abliz and T. Znati, "A guided tour puzzle for denial of service prevention," in *Proceedings of the 25th Annual Computer Conference Security Applications (ACSAC '09)*, pp. 279–288, December 2009.
- [71] B. Groza and B. Warinschi, "Cryptographic puzzles and DoS resilience, revisited," *Designs, Codes and Cryptography*, vol. 73, no. 1, pp. 177–207, 2014.
- [72] J. Patarin and A. Montreuil, "Benes and Butterfly schemes revisited," in *Proceedings of the 9th International Conference on Information Security and Cryptology (ICISC '05)*, vol. 3935, pp. 92–116, 2006.

Research Article

Dynamic Track Management in MHT for Pedestrian Tracking Using Laser Range Finder

Abdul Hadi Abd Rahman,¹ Hairi Zamzuri,¹ Saiful Amri Mazlan,¹
Mohd Azizi Abdul Rahman,¹ Yoshio Yamamoto,² and Saiful Bahri Samsuri¹

¹Vehicle System Engineering Research Lab, Universiti Teknologi Malaysia, Jalan Semarak, 54100 Kuala Lumpur, Malaysia

²Department of Precision Engineering, Tokai University, Hiratsuka 259-1292, Japan

Correspondence should be addressed to Hairi Zamzuri; hairi@ic.utm.my

Received 28 August 2014; Accepted 5 November 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Abdul Hadi Abd 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.

Real time pedestrian tracking could be one of the important features for autonomous navigation. Laser Range Finder (LRF) produces accurate pedestrian data but a problem occurs when a pedestrian is represented by multiple clusters which affect the overall tracking process. Multiple Hypothesis Tracking (MHT) is a proven method to solve tracking problem but suffers a large computational cost. In this paper, a multilevel clustering of LRF data is proposed to improve the accuracy of a tracking system by adding another clustering level after the feature extraction process. A Dynamic Track Management (DTM) is introduced in MHT with multiple motion models to perform a track creation, association, and deletion. The experimental results from real time implementation prove that the proposed multiclustering is capable of producing a better performance with less computational complexity for a track management process. The proposed Dynamic Track Management is able to solve the tracking problem with lower computation time when dealing with occlusion, crossed track, and track deletion.

1. Introduction

Detection and tracking moving object system (DATMO) is one of the most important and popular research areas in autonomous navigation. Previously, DATMO is widely used for system monitoring using camera either in indoor or outdoor usage under both static and dynamic environments but it is unable to provide accurate measurements for long distance objects. A reliable DATMO system contributes to significant improvement for other scenarios such as obstacle avoidance, path planning, and collision avoidance. The presence of Laser Range Finder (LRF) which is capable of providing accurate range information, wide coverage area, and a low time interval permits implementations in real time system. Pedestrian tracking in urban area is one of the useful implementations of DATMO for autonomous navigation. When dealing with crowded environment or urban area, a DATMO system tends to face difficulties to deal with noisy data provided by LRF. Data for different pedestrians

sometime interlace and disrupt a reliable feature extraction to represent the pedestrians. The identification of each pedestrian needs to be fused with other information such as geometrical shapes and movement histories. In order to realize DATMO in real time implementation, a suitable optimization is needed especially in computational time and detection rate.

LRF placement on vehicle or robot is very crucial to detect either waist or leg portion of pedestrians. Both methods have their own benefits and drawbacks. In leg detection situation, a walking model is used to associate two legs data in order to represent the same pedestrian. Meanwhile the waist part data may contain data of pedestrian's hands which may sometime cause occlusion for full waist data. Experimental works done by Zhao et al. [1] proved that waist data contained better representation for pedestrians. The shape of a target pedestrian that changed for one iteration to another due to changes in orientation and translation is better detected by using waist data compared to leg. The size of associated target pedestrian helps to filter less possible clusters [2].

Most researchers implemented Iterative Closest Point (ICP) for data clustering. New cluster is created when two adjacent beams are separated by a prefix threshold distance. There are two different methods to measure the threshold either using a prefix threshold value or a threshold function. Arras et al. [3] used ICP with prefix threshold for segmentation and adaption of AdaBoost for training a strong classifier to associate features of consecutive laser scans which correspond to pedestrians legs. It worked in a cluttered environment but was applied only from a static position. Some of the researchers used ICP with adaptive threshold value as in [4, 5] followed by Expectation-Maximization (EM) algorithm to separate or merge clusters. The experimental results showed that the true detection rates for all the experiments are very low for a single layer LRF. Klasing et al. [6] and Wenzl et al. [7] used fluctuations Δr to represent the circumference transformation effected by upper and lower limb movements. The position of an object is estimated by calculating the distance L between the LRF sensor and the object surface with the measurement error ΔL . The estimated distance between a LRF sensor and the object centroid is $Lr = L + r$, where r denotes the estimated mean radius.

There are numbers of approaches which have been conducted for feature extraction. Zhang et al. [8] proposed a mutation operator (MBPSO) based wrappers to reduce false positive error. It can solve other classification problems such as web-page, text, and image classification. Wrappers can perform better than filters with regard to classification performance indexes. False positive error can be reduced without compromising the sensitivity and accuracy values. Zhang et al. [9] has conducted stationary wavelet transform (SWT) to extract features in order to encountered problem occurring when using discrete wavelet transform (DWT) based classification. By applying DWT, one subject was recognized as two different subjects when the centres of the images are located at slightly different positions. SWT based features also allowed to be implemented in the MR image classification, denoising, compression, and fusion. The finding showed that SWT is superior to classical discrete wavelet transform associated with translation invariant property.

Multiple Hypothesis Tracking (MHT) algorithm preserves multiple hypotheses associating previous observations with targets in multiscan association. A new set of hypotheses is formed based on previous hypothesis after receiving new measurements at each iteration. The algorithm calculates for the highest posterior hypothesis to be chosen as the solution. MHT is categorized as a deferred logic method where a track creation or deletion of existing track is only done when the measurements are completely gathered. Hypotheses-based MHT suffers from the computational complexity to maintain and expand the previous hypotheses. To overcome this problem, several heuristic methods are applied but they sacrifice the Maximum A Posterior (MAP) estimation property. Ding and Chen [10] successfully verified the MHT technique via simulation for moving object tracking to be applied in unmanned vehicle operation. Coraluppi [11] provided simulation results demonstrating the advantages of multistage MHT processing compared to single-stage MHT, track-while-fuse processing. The simulation results

proved that multistage MHT offers a powerful and flexible paradigm to circumvent limitations in conventional MHT processing. Vasquez and Williams [12] proposed MHT with Integral Square Error Reduction (MISER) to represent the dependency which occurs between targets effected by the joint observation process and applied Integral Square Error (ISE) mixture reduction algorithm to control the hypothesis growth. Mohammed et al. [13] developed Fast-IMM (Fast Interacting Multiple Model) for tracking a single maneuvering target which decreases the computational burden but keeps a high accuracy. In the study by Blackman [14], it is proven that track-oriented MHT can easily maintain several hundred of tracks and expand into new hypotheses for difficult scenarios. Vu et al. [15] implemented an adaptive Interacting Multiple Models filter which consists of 16 motion models coupled with a Multiple Hypothesis Tracker to solve moving objects tracking.

In this paper, a multiclustering method is introduced to solve spurious data obtained from Laser Range Finder in an outdoor environment. For tracking part, a new Dynamic Track Management for MHT (DTM-MHT) is proposed to produce lower computational time for real time implementation in urban area. It is evaluated to deal with track creation and crossed track, as well as track deletion in real time implementation.

2. Multiclustering

Multiclustering is significantly important to reduce a total size of observation at each iteration before data tracking process is done. An efficient multiclustering provides more reliable observation data which will reduce complexity in track management process. Firstly, the laser data at each iteration are clustered using ICP with adaptive threshold. Then, filtering is accomplished by removing clusters with total point numbers less than 5. This simple filtering is important to reduce the computation (DTM) time by eliminating spurious data from measurement noise. Then the measurement of the width for each cluster is done for baseline before feature extraction. The width of the cluster is important to reduce the computation time for unwanted features other than pedestrian which has the width between 40 cm and 70 cm. The next step is to determine the geometrical feature of the detected object as mentioned in [16]. Feature extraction using an ellipse fit and a circle fit was used for all clusters with circle or arc features. The feature extraction results are then transformed into global Cartesian coordinates based on the vehicle position.

The multiclustering process is illustrated as in Figure 1. The Laser Range Finder (LRF) provides the polar coordinate (r, θ) for the body and arm parts of a pedestrian. L represents the threshold distance between body and arm. The rotation and translation of the pedestrians are then calculated according to the feedback from motion model (MM) selection in data association process. MM is attributed to the laser data at $t - 1$. With the selected motion model and the tracking result from the previous iteration, the region area for each pedestrian is computed and multiclustering of all

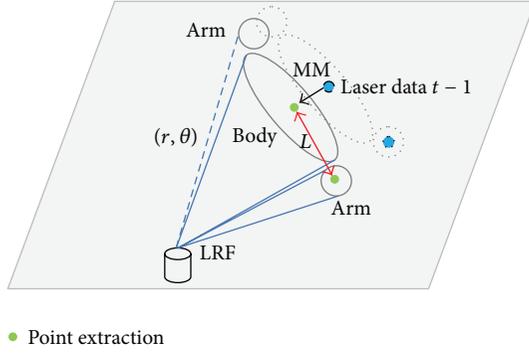


FIGURE 1: Multiclustering technique.

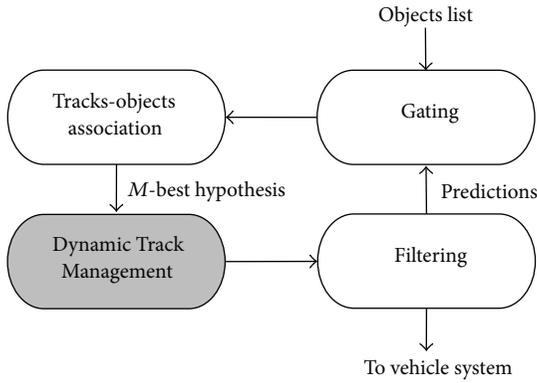


FIGURE 2: MHT process.

the neighbouring clusters is accomplished using the following equation:

$$\text{Nbr}(C_i^t) = \{C_j^{t-1} \mid \text{dist}(C_i^t, C_j^{t-1})\} < \text{Thresh}_{\text{nbr}}, \quad (1)$$

where C_i^t denotes the i th cluster at time t and $\text{dist}(\cdot)$ denotes the function for computing the distance between two clusters. Each pair of clusters C_j^{t-1}, C_j^{t-2} in $\text{Nbr}(C_i^t)$ is tested whether a merge case is generated according to the size of the overlapping area for a pedestrian.

3. Multiple Hypothesis Tracking

The flow for MHT is illustrated in Figure 2. The MHT forms different hypotheses for every possibility which includes detection of new tracks, false alarms, or existing tracks. First, the object list is calculated to determine candidates for association in the gating procedure. The results of the gating process will undergo a data association process to find the best possible track-object association. Next, the tracks are managed to deal with track creation, track deletion, and false measurement. Furthermore, the movement for each object for the next iteration is predicted in the filtering process. These processes are repeated for every iteration. The details of each step of MHT with proposed Dynamic Track Management are described in this section.

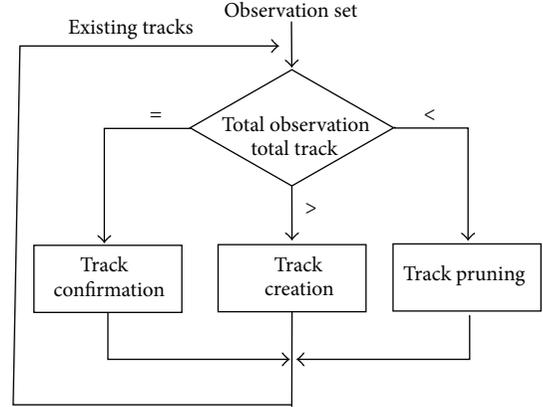


FIGURE 3: Dynamic Track Management.

3.1. Gating. The object list obtained from the feature extraction of raw laser data is used as an input for gating procedure. Gating is necessary for early detection of false measurements and new objects. Global Nearest Neighbour (GNN) algorithm is used to calculate the distance between the tracks and observations, which produces candidates for association. The result of the gating procedure is a set of compatible observation-track pairs of current hypothesis which is processed in data association step.

3.2. Data Association. Reids algorithm [17] is a hypothesis-based MHT implementation which keeps the past different hypotheses in the memory between consecutive time steps. When a new measurement falls within the gate region, it will form a list of possible assignments between measurements and exiting tracks which expanded the existing hypothesis. Each hypothesis contains a set of potential track-object assignments, which cause an inhibitory time consuming process in determining all the possible assignment combinations. If there is no measurement which is compatible with one of existing tracks, then a new track or a false alarm should be formed. The implementation of MHT considered in this paper is similar to running GNN-based multitarget trackers in parallel. Then N -scan pruning and M -best hypotheses techniques are used to reduce the number of hypotheses. M -best hypothesis is able to optimally determine the M -best assignments in polynomial time. It reduces the dimension and thus precludes solving duplicate assignment problems.

3.3. Dynamic Track Management. At this stage, Dynamic Track Management is proposed to manage track creation, confirmation, or deletion as shown in Figure 3. The implementation is based on track-oriented approach. New hypotheses are formed using the updated tracks and the new observation data on each iteration.

The process starts with an early detection to determine which type of management needs to be executed. It is done by subtracting the number of observation set from the number of existing tracks available. The previous probability for

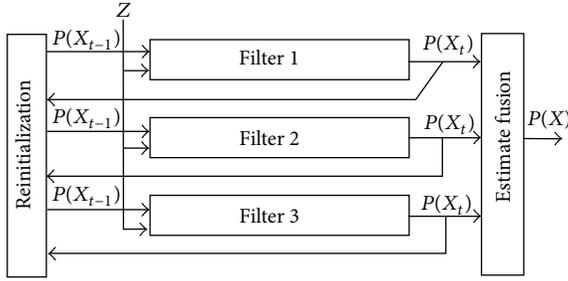


FIGURE 4: Basic IMM steps.

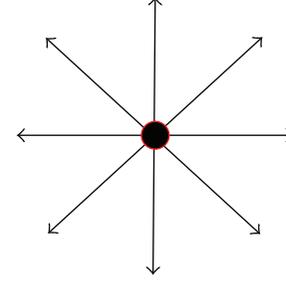


FIGURE 5: Motion models for IMM.

each track is considered in this step. The creation of new tracks only takes place when a new track creation hypothesis appears in the M -best hypotheses. The new tracks are confirmed after detected objects appear along the same track over several consecutive iterations to prevent track creation for spurious measurements. Evaluation of track scores using Sequential Probability Ratio Test (SPRT) is used to perform the track confirmation test. The selection of confirmation and deletion thresholds is associated with the tracking requirements through the parameters α for false track confirmation and β for true track deletion probability. A new hypothesis is reformed by using the new observations based on the tracks which survived after track pruning process. The survive tracks after pruning process are predicted using the new observations obtained and reformed into new hypotheses. Some tracks are deleted based on low probability or N -scan pruning. The track scores containing all the relevant statistical data for track pruning are maintained for the next track management process.

Non-detection hypothesis appears when occlusion of measurement occurs, that is, when an object is hidden by another object or disappears from perception sensor. When this happens, the track measurement for unassociated objects is updated according to the previous associated objects via prediction using Interacting Multiple Models (IMM) in a later step. The track will be deleted and would be unavailable to be occupied for any upcoming observations when a track is not updated for a certain iteration limit. Furthermore, the continued growth of the tracks is controlled by performing another pruning. Typically the growth is controlled by N -scan pruning technique by keeping only the N previous scans in the trees. The numbers of tracks trees were delimited to N which simply the execution of MHT algorithm.

3.4. Filtering. The basic Interacting Multiple Models (IMM) steps are illustrated in Figure 4. It consists of multiple filters which are used to estimate the next pedestrian's state. The pedestrian's movements were emulated by a set of possible directions using IMM models. The parameters for each motion model were predefined as it represents the corresponding filter. With an assumption of having the same velocities in eight possible directions, motion of an object will fall in one of the 8 motion models as shown in Figure 5.

Kalman filters are integrated in each motion model for prediction. The Kalman filter equations are given as follows:

$$\begin{aligned} x_k &= F_k x_{k-1} + G_k u_{k-1} + w_k, \\ Z_k &= H_k x_k + v_k, \end{aligned} \quad (2)$$

where matrices F_k , G_k , and H_k are time dependent and the noises w_k and v_k have covariance matrices Q_k and R_k , respectively. If the prior distribution $p(x_o)$ is also a Gaussian $N(x_o; \hat{x}_o, P_o)$, the distributions resulting from the prediction and update steps are also Gaussian. Therefore, the beliefs of the Kalman filter are completely specified by the first two moments of the distribution. Thus, the distribution $p(x_k | Z_{1:k-1}, u_{0:k-1}) = N(x_k; \hat{x}_{k|k-1}, P_{k|k-1})$ where the mean $x_{k|k-1}$ and covariance matrix $P_{k|k-1}$ are

$$\begin{aligned} \hat{x}_{k|k-1} &= F_k \hat{x}_{k-1} + G_k u_{k-1}, \\ P_{k|k-1} &= F_k P_{k-1} F_k^T + Q_k. \end{aligned} \quad (3)$$

In the same way, the mean \hat{x}_k and covariance matrix P_k resulting from the update step are

$$\begin{aligned} \hat{x}_k &= \hat{x}_{k|k-1} + W_k v_k, \\ P_k &= P_{k|k-1} - W_k S_k W_k^T, \end{aligned} \quad (4)$$

where the innovation v , the predicted measurement $\hat{Z}_{k|k-1}$, the innovation covariance S_k , and the filter gain W_k are

$$\begin{aligned} v &= (Z_k - \hat{Z}_{k|k-1}), \\ \hat{Z}_{k|k-1} &= H_k \hat{x}_{k|k-1}, \\ S_k &= H_k P_{k|k-1} H_k^T + R_k, \\ W_k &= P_{k|k-1} H_k^T S_k^{-1}. \end{aligned} \quad (5)$$

4. Experimental Setup

The developed DATMO algorithm with the DTM-MHT was evaluated on its capability to deal with various important situations in tracking part. Two most important aspects for evaluation were tracking computation time and position accuracy during tracking process. The capabilities of pro-

TABLE 1: Experiment parameters.

Parameter	Setting
Area	40 meters \times 20 meters
Scenario duration	770 iterations
Target number	1 to 6 objects
Sampling rate	0.2 s
Laser distance range	5 meters
Track creation	3 hits
Track kill	8 misses



FIGURE 6: Hokuyo LRF placed on iREV.

posed multiclustering were tested with real time implementation in an outdoor environment where a high level of noise was expected. It becomes more complex when LRF data are obtained from a moving vehicle. The proposed Dynamic Track Management was evaluated on the capability to handle missing observation cases due to occlusion, cross track, track creation, and track deletion. The tracking computation time was computed in every iteration for evaluation.

The details of the experimental setup are shown in Table 1. The experiment was conducted in area of 40 m \times 20 m. A custom-made buggy car called iREV (Instrumented Research Vehicle) developed in [18] was used to provide the trajectory data which contained the vehicle pose state (X , Y and the heading, θ). A Hokuyo Laser Range Finder (LRF) was placed in front of the vehicle at 1.2 m height from the ground to provide multiple pedestrian data on the waist part as shown in Figure 6. The data from LRF covers the area within 5 meters in 180° width. The resolution of the LRF was set to 0.333° to provide more data compared to 0.5° used in most previous works. This simply increases the accuracy in feature extraction process of multiple pedestrians.

Figure 7 shows three different scenarios for evaluations and analysis. Figure 7(a) involves 5 pedestrians appearing from different direction heading across the moving vehicle. There occurs an occlusion scenario where two pedestrians walk in parallel to each other causing blind spot from the laser perception. The second scenario is found in Figure 7(b) where 4 pedestrians are detected and moving closely to each other in the same direction labelled as “B.” The occlusion situations were labelled as label “A” and “C” while “D” represents one of the track deletion cases. Figure 7(c) was set up to deal with high measurement noise from laser data

labelled as “A” while “B” represents crossed track situation. In this experiment, each scenario is separated by a longer time interval. It is considered as a more realistic situation where pedestrians would not appear for a long time as the vehicle moves along its path.

The computational time was computed for evaluation of the overall tracking process which includes multiclustering process and DTM-MHT. The performance of tracking results for position of each pedestrian was then evaluated using Root Mean Squared Error (RMSE).

5. Results and Discussion

This section presents the results of the pedestrian tracking from a moving vehicle based on the experimental setup. As mentioned before, multiclustering is supposed to deal with spurious data from LRF sensor. The first stage clustering contained noise data and cluster candidates. Thus, choice of selecting threshold value is critical in selecting candidates for multiclustering process. The results from multiclustering are used for feature extraction process. The feature extraction results from multiclustering process were represented and labelled as PE in Figures 8(a), 8(b), and 8(c) using an ellipse fit technique. It is observed that some extraction points are fluctuated due to measurement noise. However, in unlike the tracking part, the occlusion problem could not be resolved in these feature extraction results.

The occlusion problems were solved using the motion model estimation. Use of motion models allows the prediction of next pedestrian movement. The selection of motion models are based on the highest probability obtained from previous hypotheses. Next, filtering with Kalman filter on the motion models information was done to complete the pedestrian tracking process. Figures 8(a), 8(b), and 8(c) presented the tracking path produced by the motion models. The motion model selection during *missed detection* situation produced estimation for certain period of time based on track kill threshold set in Table 1 before track deletion is done or maintained the track. Based on this motion model, filtering was done by using Kalman filter to predict the next movement of the entire pedestrians. The tracking results reflect successful implementation for DATMO algorithm in dealing with various scenarios as explained in the experimental setup.

In the first scenario, temporarily missed detection labelled as “A” and “B” in Figure 7(a) was solved as illustrated in Figure 8(a). The pedestrian that temporarily disappeared for a certain period of time and then reappeared was tracked via estimation using motion models and labelled as MM2 in Figure 8(a). The tracking results related to crossed track and occlusion situation are labelled as $Est4$ and $Est5$. All the tracks were successfully deleted after they disappeared for a certain period of time or static motion models were selected. The result for the second scenario is shown in Figure 8(b) where 4 pedestrians were tracked moving across the moving vehicle. Occlusion appeared due to pedestrians’ parallel movements. In this scenario, the proposed algorithm was evaluated to deal with close distance between tracked pedestrians. The tracking results for the third scenario are shown in Figure 8(c) to deal

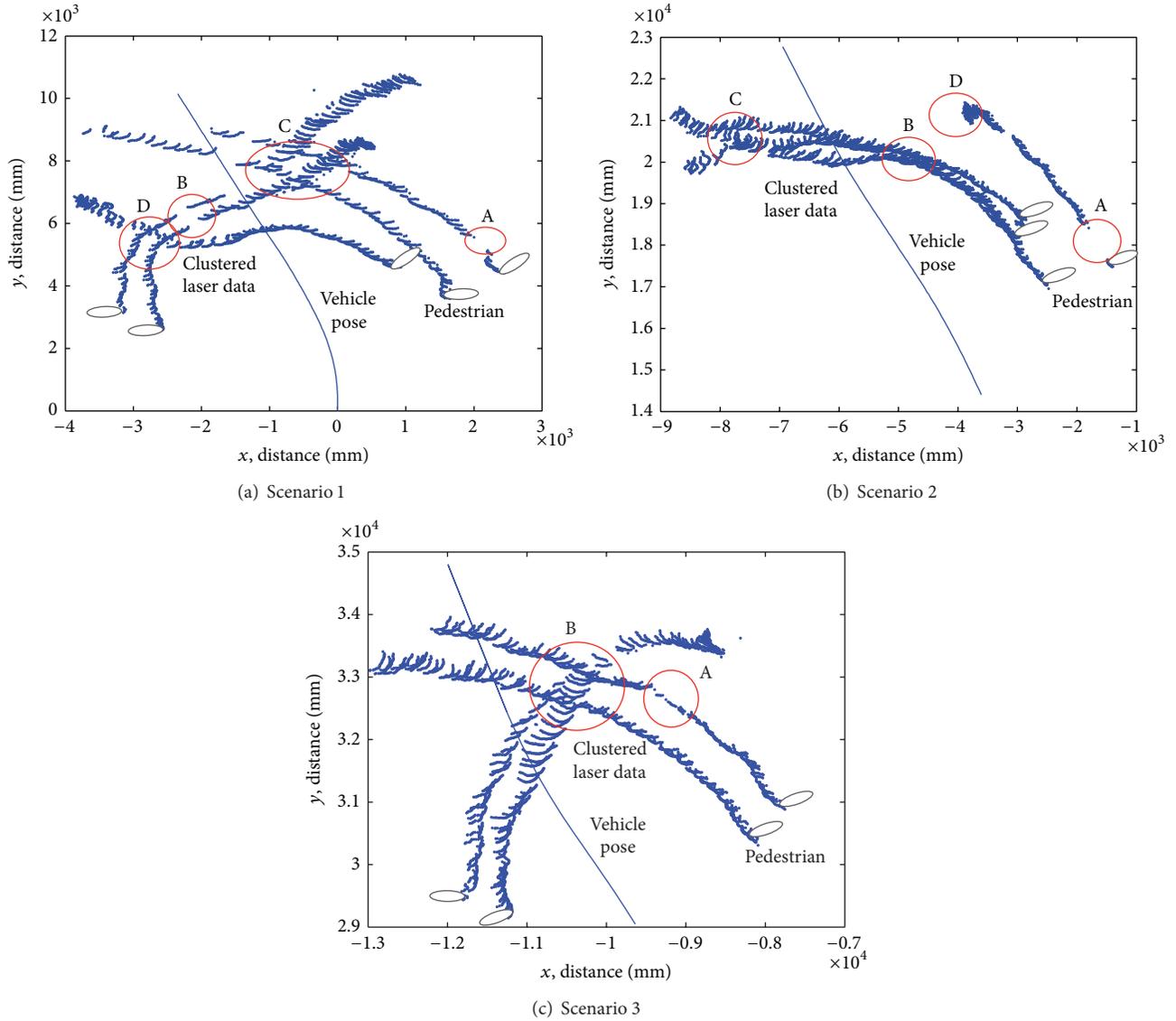


FIGURE 7: Data of multiple pedestrians in three different scenarios.

with data association during crossed track. It is observed that the tracking was having difficulties in solving tracking due to high measurement noise from laser data but still able to associate pedestrian to track correctly.

The overall performance of the tracking algorithm for the all scenarios is indicated by Root Mean Squared Error of positions for multiple pedestrians as tabulated in Table 2. The results reflected the tracking performance of the proposed algorithm. It is suggested that the algorithm was able to perform correct data association for all 5 pedestrians based in all given scenarios. RMSE values for position of pedestrians labelled 1, 3, and 5 were found lower compared to those for pedestrians 2 and 4. They were affected by consistent detections for all the pedestrians until they disappeared while the latter had missed the detections and thus affected the tracking results. The RMSE values for the positions of all pedestrians were less than 0.5 for all pedestrians in the second

TABLE 2: RMSE for position for each tracked pedestrian.

RMSE	Scenario 1		Scenario 2		Scenario 3	
	x (m)	y (m)	x (m)	y (m)	x (m)	y (m)
Pedestrian 1	0.131	0.281	0.106	0.186	0.125	0.447
Pedestrian 2	0.127	0.112	0.093	0.247	0.231	0.461
Pedestrian 3	0.138	0.224	0.086	0.237	0.231	0.49
Pedestrian 4	0.12	0.129	0.053	0.28	0.187	0.435
Pedestrian 5	0.142	0.134	—	—	—	—

and third scenarios, which reflected successful tracking for all pedestrians.

The computation time of the proposed DTM-MHT was computed to evaluate the complexity of the tracking process. Figure 9(a) shows the relation between track number and

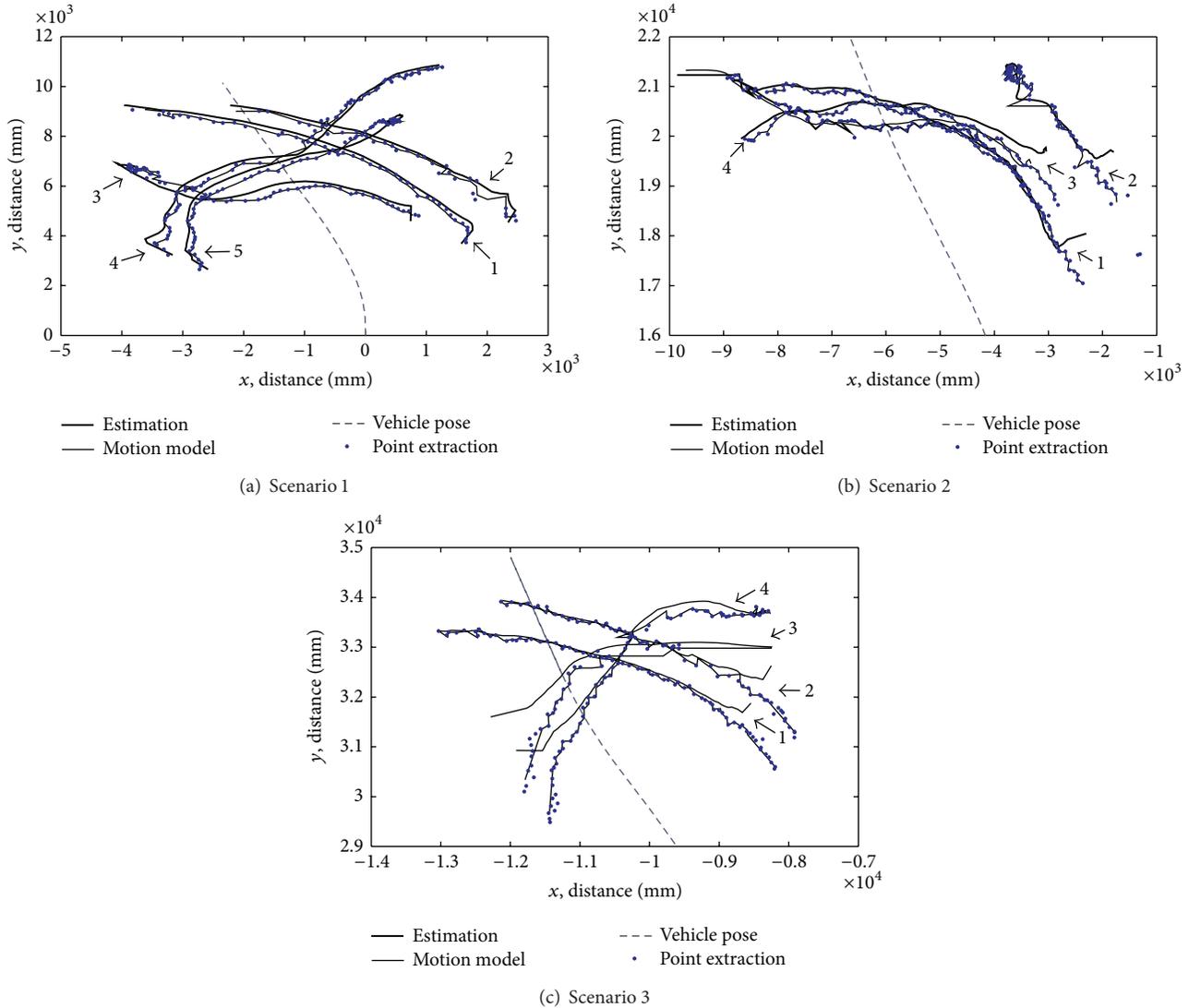


FIGURE 8: Tracking results for all tracks.

the computation time. The average computation time in idle cases where no tracking was performed is 9 ms. At 115th iterations, initialization process started when new tracks are added and started the tracking process. During this process, the computation time was at the peak level at 30 ms. The algorithm processed 5 pedestrians from 125th to 161th iterations and recorded computation time at 14.2 ms. The computational time is fluctuated due to missed detection which requires more computation time. From 162th to 181th iterations, the tracks reduce to four tracks which recorded less computational time at 13.1 ms. The average computation time reduced to 12.9 ms when tracking 2 pedestrians from 182th to 193th iterations. The fact that tracks reduced to one and disappeared after a few iterations means no available pedestrian to track. The computation time and track management second scenarios are illustrated in Figure 9(b). Track creation was performed at the 547th iteration when 2 pedestrians were detected and tracked. Furthermore tracks are added in

the 559th and 581th iterations. Missed detection happened between 584th and 591th iterations but no track deletion was performed as it still did not exceed the value for track deletion. Track deletions were accomplished at iterations 620, 629, and 643. In the 3rd scenario as shown in Figure 9(c), track creation started at the 678th iteration and the tracks were maintained until the 701th iteration before a track deletion was performed. All 4 tracks were deleted at 748th iteration.

From the results in all three scenarios, the average computation time for each number of pedestrians is recorded between 11.8 ms and 17.23 ms as in Table 3 to deal with up to 5 pedestrians. The average incremental time needed to process every new pedestrian is 1.3575 ms. It allows process for larger number of pedestrians for each scene. The track reinitialization adopted in the track management process reduces the hypotheses complexity problem which could be enlarged as a response to the incremental number of pedestrians tracked.

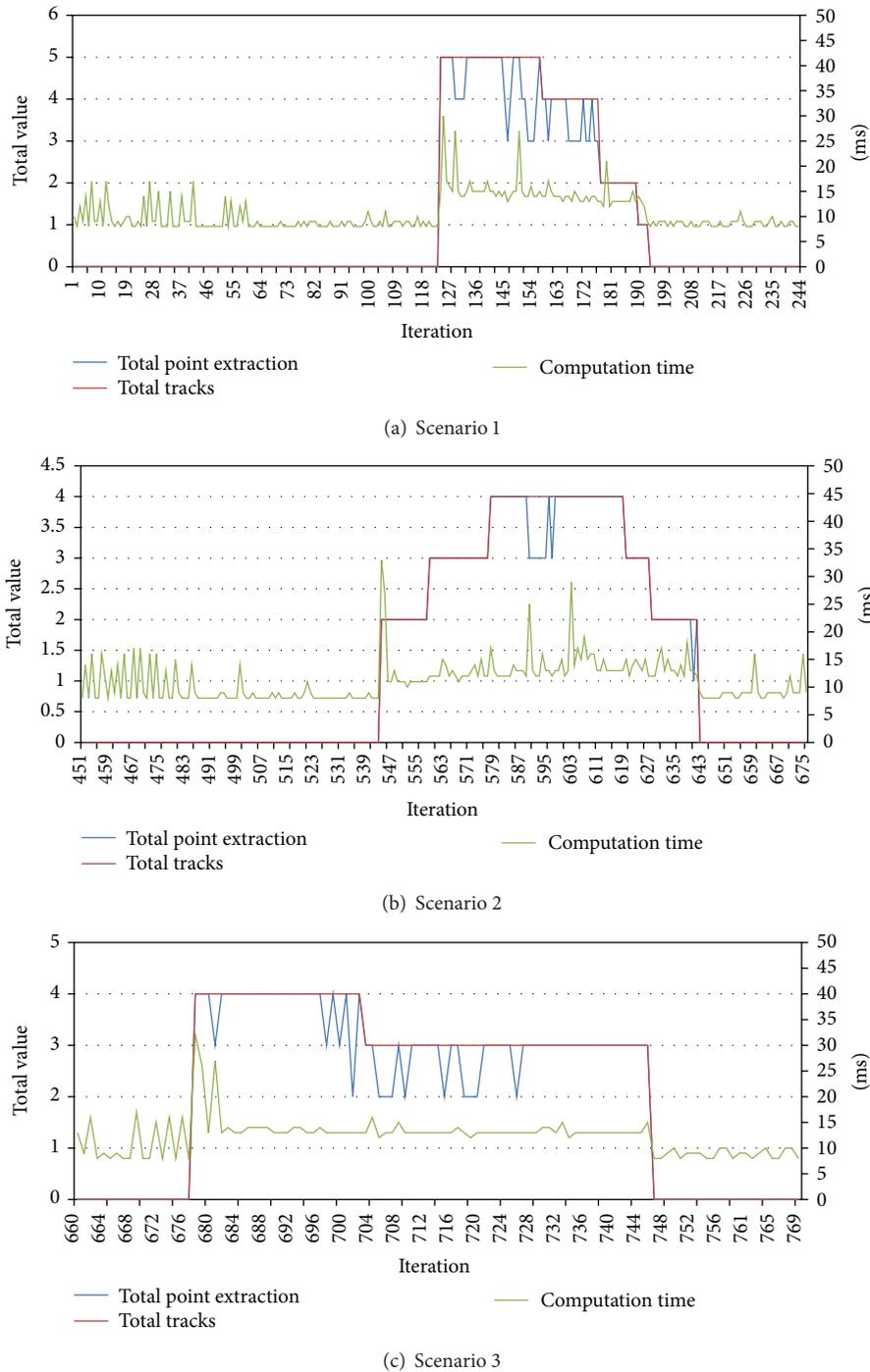


FIGURE 9: Computational time and track management.

6. Conclusion and Future Work

This paper mainly presents the experimental results of the developed DATMO algorithm with DTM-MHT and multiclustering for tracking multiple pedestrians in real time implementation. It was validated via real time experiment in terms of its computational time and accuracy. It was observed that low computational time is achieved in all

scenarios given and capable of associating all the detected object with their tracks and predicted their movements. The multiclustering method successfully filtered unwanted data and produced reliable observations for tracking purpose. Furthermore, this proposed algorithm will be applied in more complex scenarios involving more pedestrians. The detection accuracy could be improved using fusion of various sensors and configurations. The tracking technique could be

TABLE 3: Average computation time for each tracked pedestrian.

Total pedestrian	Average computation time
1	11.8 ms
2	12.63 ms
3	13.41 ms
4	15.34 ms
5	17.23 ms

expanded to other fields such as image processing and other intelligent surveillance applications.

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 the Malaysia-Japan International Institute of Technology (MJIIT) in Universiti Teknologi Malaysia and Ministry of Education Malaysia (MOE) under FRGS (vote: 4F370) and Research University Grant (vote: 00G64) for funding and supporting this research.

References

- [1] T. Zhao, R. Nevatia, and B. Wu, "Segmentation and tracking of multiple humans in crowded environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1198–1211, 2008.
- [2] F.-M. Chang and F.-L. Lian, "Polar grid based robust pedestrian tracking with indoor mobile robot using multiple hypothesis tracking algorithm," in *Proceedings of the 51st Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE '12)*, pp. 1558–1563, Akita, Japan, August 2012.
- [3] K. O. Arras, Ó. M. Mozos, and W. Burgard, "Using boosted features for the detection of people in 2D range data," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pp. 3402–3407, IEEE, April 2007.
- [4] A. Carballo, A. Ohya, and S. Yuta, "Reliable people detection using range and intensity data from multiple layers of laser range finders on a mobile robot," *International Journal of Social Robotics*, vol. 3, no. 2, pp. 167–186, 2011.
- [5] P. Kondaxakis, S. Kasderidis, and P. Trahanias, "A multi-target tracking technique for mobile robots using a laser range scanner," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '08)*, pp. 3370–3377, September 2008.
- [6] K. Klasing, D. Wollherr, and M. Buss, "Realtime segmentation of range data using continuous nearest neighbors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 2431–2436, May 2009.
- [7] K. Wenzl, H. Ruser, and C. Kargel, "Performance evaluation of a decentralized multitarget-tracking algorithm using a LIDAR sensor network with stationary beams," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 5, pp. 1174–1182, 2013.
- [8] Y. Zhang, S. Wang, P. Phillips, and G. Ji, "Binary PSO with mutation operator for feature selection using decision tree applied to spam detection," *Knowledge-Based Systems*, vol. 64, pp. 22–31, 2014.
- [9] Y. Zhang, S. Wang, Y. Huo, L. Wu, and A. Liu, "Feature extraction of brain MRI by stationary wavelet transform and its applications," *Journal of Biological Systems*, vol. 18, no. 1, pp. 115–132, 2010.
- [10] R. Ding and W.-H. Chen, "Moving object tracking in support of unmanned vehicle operation," in *Proceedings of the 19th International Conference on Automation and Computing (ICAC '13)*, pp. 1–6, IEEE, London, UK, September 2013.
- [11] S. Coraluppi, "Multi-stage multiple-hypothesis tracking," *Journal of Advances in Information Fusion*, vol. 6, no. 1, pp. 57–68, 2011.
- [12] J. R. Vasquez and J. L. Williams, "Improved hypothesis selection for multiple hypothesis tracking," in *Signal and Data Processing of Small Targets*, vol. 5428 of *Proceedings of SPIE*, 2005.
- [13] D. Mohammed, K. Mokhtar, O. Abdelaziz, and M. Abdelkrim, "A new IMM algorithm using fixed coefficients filters (fast-IMM)," *AEU—International Journal of Electronics and Communications*, vol. 64, no. 12, pp. 1123–1127, 2010.
- [14] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, no. 1, pp. 5–18, 2004.
- [15] T.-D. Vu, J. Burlet, and O. Aycard, "Grid-based localization and local mapping with moving object detection and tracking," *Information Fusion*, vol. 12, no. 1, pp. 58–69, 2011.
- [16] J. Xavier, M. Pacheco, D. Castro, A. Ruanot, and U. Nunes, "Fast line, arc/circle and leg detection from laser scan data in a player driver," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3930–3935, IEEE, April 2005.
- [17] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.
- [18] M. A. Zakaria, H. Zamzuri, R. Mamat, and S. A. Mazlan, "A path tracking algorithm using future prediction control with spike detection for an autonomous vehicle robot," *International Journal of Advanced Robotic Systems*, vol. 10, article 309, 9 pages, 2013.

Research Article

KD-ACP: A Software Framework for Social Computing in Emergency Management

Bin Chen, Laobing Zhang, Gang Guo, and Xiaogang Qiu

Research Center of Computational Experiments and Parallel System Technology, College of Information System and Management, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Bin Chen; nudtcb9372@gmail.com

Received 4 June 2014; Accepted 23 August 2014

Academic Editor: Praveen Agarwal

Copyright © 2015 Bin Chen 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 addresses the application of a computational theory and related techniques for studying emergency management in social computing. We propose a novel software framework called KD-ACP. The framework provides a systematic and automatic platform for scientists to study the emergency management problems in three aspects: modelling the society in emergency scenario as the artificial society; investigating the emergency management problems by the repeat computational experiments; parallel execution between artificial society and the actual society managed by the decisions from computational experiments. The software framework is composed of a series of tools. These tools are categorized into three parts corresponding to “A,” “C,” and “P,” respectively. Using H1N1 epidemic in Beijing city as the case study, the modelling and data generating of Beijing city, experiments with settings of H1N1, and intervention measures and parallel execution by situation tool are implemented by KD-ACP. The results output by the software framework shows that the emergency response decisions can be tested to find a more optimal one through the computational experiments. In the end, the advantages of the KD-ACP and the future work are summarized in the conclusion.

1. Introduction

Emergency management attracts the attention of scientists from social computing because the whole process of emergent events is deeply coupled with human society and the emergency response decisions need an approach to testify their effect without the reappearance of emergent events in the society. As a new paradigm of computing and technology development, social computing helps scientists to understand and analyze individual and organizational behavior and facilitate emergency management research and application in many aspects [1].

Based on the fruitful development of computational methodology on emergency management research over the last decade, lots of work has been done to solve the problems in society domain. Both the conceptual frameworks in multiple discipline and the technological platforms developed for the domain requirements are more and more popular in the research on emergency management, especially the agent-based modelling and simulation [2]. The bottom-up technique describes the society in microview by modelling

individual behavior, communications in agents, and evolution rules of agent organizations. It is worth notifying that the modelling of agent does not emphasize the intelligence of individual. Large scale, communications and the emergence phenomena are the objects of agent-based modelling and simulation. The agent oriented platforms such as Biowar [3], GASM [4], and EpiSims [5] to study emergency problems have been proposed in many fields. Biowar developed by Carnegie Mellon University is used to study the bioattacks in city with the ability of scalable agent modelling. GASM (Global-Scale Agent Model) by Epstein simulates a global H1N1 epidemic with 6.5 billion people. EpiSims from Los Alamos national lab is used to testify the intervention measures in epidemics of smallpox from United States Department of Health and Human Services.

With the help of agent-based modelling, simulation technique and the concept of artificial society [6], a novel conceptual framework based on artificial systems is introduced in the social computing. The conceptual framework called ACP (Artificial Society, Computational Experiments, Parallel Execution) approach is proposed by Wang in 2004 [7–9]. It is

a novel approach in social computing to solve the problems in society domain. ACP approach is categorized into three aspects: representing and modelling society with artificial systems, analysis and evaluation by computational experiments, and control and management of real society by parallel execution. Under the instruction of conceptual framework of ACP approach, a wide spectrum of complex systems, such as transportation, medicine, finance, and environment, can be studied in the computational manner. Actually, many real-world applications using ACP approach have been developed to solve the real problems in domains. For instance, complex socioeconomic system [10] and the research framework for e-commerce system [11] are the good applications of ACP approach in the economic area. The ACP-based framework for integrative medicine [12] is proposed to solve the problems in medicine. An overall framework of emergency rescue decision support system of petrochemical plant [13] is proposed based on ACP theory to study environment risk accidents of petrochemical plant. Parallel BRT operation management system [14] based on ACP approach has been constructed to detect the quantity of passengers on stations real-time, traffic flow on stations or at intersections, and queuing length of vehicles on the road. A novel parallel system for Urban Rail Transportation (URT) [15] based on ACP approach is proposed to address issues on safety efficiency and reliability of the operation of URT. An artificial power system [16] is set up on the models of power systems and complex power grids to provide a feasible approach for the control and management of the modern power system.

Although a lot of work has been done on the concepts and theory framework to study problems by social computing, the following problems of computational experiments in emergency management are still not solved from the perspective of modelling for social systems theory and software framework of platform implementation.

- (i) The modelling and simulation of emergency management are not given the special consideration. The representation of society focuses on the generic modelling of agent (represented by Repast [17, 18]). The description of environments is too simple to meet the requirements from research on emergency problems, such as the building size, the place related agent contact frequency.
- (ii) The existing tools and platforms cannot support the design of experiment. Computational experiments cannot be done systematically and automatically.
- (iii) The existing applications of ACP-based frameworks are still domain specific. A generic workflow and integrated toolkit are needed to implement ACP approach, especially in the application of emergency management.

Therefore, it is necessary to develop an ACP-based software framework for the research on emergency management. The artificial system is the projection of real world in the emergent scenario. The modelling of the system including the emergent events modelling and intervention measures modelling, the design of computational experiments considering the settings

of emergency parameters, the settings of large samples experiments, and the parallel execution with loose connection of real society should be considered inside the software framework.

As a result, the purpose of this paper is to propose a software framework called KD-ACP applying the ACP-based computational theory and corresponding methods in studying emergency management problems. KD is short for the Chinese phonetic alphabets of China National University of Defense Technology. KD-ACP means the software framework is developed by National University of Defense Technology to implement ACP approach. The remainder of this paper is organized as follows. Section 2 summarizes the existing agent-based modelling and simulation platforms. Section 3 introduces ACP approach first and proposes the KD-ACP platform. Section 4 illustrates the modelling of Beijing city with KD-ACP; both the agent models and initial data are considered. Section 5 shows how to do experiments with KD-ACP using the H1N1 case study in artificial Beijing. In the end, the paper is concluded in Section 6.

2. Related Works

There have been many efforts on social computing, especially on the emergency management. Agent-based modelling and simulation are popular in the implementation of social computation. The related works are mainly categorized into SWARM-like agent-based modelling and simulation platforms and agent-based platforms for emergency management.

2.1. SWARM-Like Agent-Based Modelling and Simulation Platforms. SWARM [19] originally proposed by Santa Fe institute is widely used in many research areas such as biology, ecology, and society. The tool provides a simulation environment for simulating agent with the support of a series of class libraries. It is worth noting that SWARM is the precursor of multiagent simulation tool; it influences lots of multiagent simulation platforms such as NetLogo [20], RePast (REursive Porus Agent Simulation Toolkit), MASON [21], and SOARS (Spot Oriented Agent Role Simulator) [22].

NetLogo is a multiagent programming language and modelling environment for simulating natural and social phenomena. It is particularly well suited for modelling complex systems evolving over time. The language is easy to study and the agent-based complex systems could be built rapidly; RePast is a software framework for agent-based simulation created at the University of Chicago. An extensible simulation package makes RePast become a generic multiagent simulation platform in social science research computing; MASON designed by George Mason University is used to serve as the basis for a wide range of multiagent simulation tasks ranging from swarm robotics to machine learning to social complexity environments. The tool is a fast discrete-event multiagent simulation toolkit in Java; SOARS is designed by Tokyo Institute of Technology to describe agent activities under the roles of social and organizational structure. Decomposition of multiagent interaction is the most important characteristics in this framework.

All the SWARM-like agent-based modelling and simulation platforms provide a portable, lightweight, and easily extensible environment for simulating agents in arbitrary research areas. However, the heterogeneity in specific social computing domain is not considered. Furthermore, most of the platforms cannot well support large scale agent simulation because of the lightweight engine. The engine cannot afford the simulation of super cities like Beijing and New York which have millions of people.

2.2. Agent-Based Platforms for Emergency Management. Biowar proposed by Carnegie Mellon University simulates the impact of background diseases, bioterrorism attacks within a city. 62 diseases are modeled in this platform to simulate the outbreaks on the population's behavior. GASM (Global-Scale Agent Model) is designed to study the spreading of H1N1; 6.5 billion population is modeled with the support of official statistical data. A global H1N1 spread from Tokyo is simulated in GASM. EpiSimS proposed by Los Alamos lab simulates the spread of disease in regions such as cities, allowing for the assessment of disease prevention, intervention, and response strategies. The daily movements and interactions of synthetic individuals are represented explicitly. Burke and Epstein propose a computational model of smallpox epidemic transmission and control [23]. The agents in this model interact locally with one another in social units such as homes, workplaces, schools, and hospitals.

However, these platforms cannot provide a generic software framework to study emergency problems. Biowar only focuses on social networks; individuals are all modeled as the nodes of social networks. Agents in GASM and EpiSimS are isomorphic, without considering the heterogeneity in specific domains.

To sum up, this section briefly reviews the existing multiagent simulation platforms including SWARM-like agent-based modelling and simulation platforms and agent-based platforms for emergency management. However, they cannot satisfy the requirements of simulation performance, adaptability of software framework, and heterogeneity of individuals in research of different emergency scenarios.

3. KD-ACP

KD-ACP is an integrated software framework designed and implemented based on the principle of the ACP approach shown in Figure 1.

3.1. The ACP Approach. ACP approach is a social computing-based research paradigm. It is composed of three components as its name: artificial society for A, computational experiments for C, and parallel execution for P. The basic idea of ACP approach is listed as follows.

- (i) Model the complex societies involving human behavior and social organizations as artificial societies using multiagent modelling techniques in a “bottom-up” fashion. Artificial societies are regarded as a research platform to study emergency management.

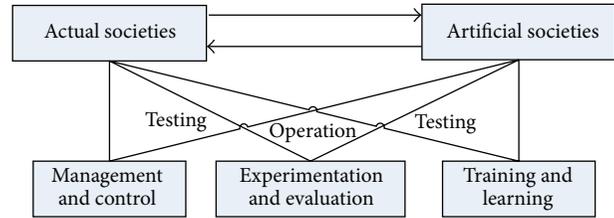


FIGURE 1: The parallel execution of ACP approach [8, 33].

- (ii) Utilize innovative computing technologies to evaluate and analyze various factors in emergency management quantitatively; the computers are regarded as the experimental social laboratories for investigating emergency management problems.
- (iii) Provide an effective mechanism for the control and management of complex actual social society through comparison, evaluation, and interaction with artificial society.

It is worth notifying that “P” here is not the “parallel” in “parallel simulation” but the representation of “parallel execution.” The idea of parallel execution is to build the parallel scenarios by paralleling the actual societies and artificial societies. Consequently, parallel control and management of actual societies are implemented with the help of interactions between parallel scenarios. The goal of parallel execution is to find the best plans to adjust the methods of control and management based on the comparison and analysis of differences between actual and artificial societies. Artificial societies provide possible simulated results of evolutions by repeated computational experiments. The simulated results provide evidences for the adjustment plans. These plans are used in the control and management of actual societies, such as emergency management. After the application of these plans, the observations from actual societies are collected for the comparison with expectation. The differences are used to feedback to artificial societies. The new turn of comparison and analysis to find best adjustments of control and management is repeated.

The mechanism of “parallel execution” has been proved to be effective for use in networked complex traffic systems and is closely related to emerging technologies in cloud computing, social computing, and cyber-physical-social systems [24]. In order to promote the development of parallel control and management in emergency management, the artificial society is proposed in ACP approach which is the expansion of “artificial traffic systems.”

Instructed by the ACP approach, KD-ACP is also composed by three components. The details of the architecture and implementation of KD-ACP are discussed below.

3.2. The Software Architecture of KD-ACP. The architecture of KD-ACP is shown in Figure 2; the software framework is composed of a series of tools. These tools are grouped into three parts to support artificial society modelling, computational experiments, and parallel execution.

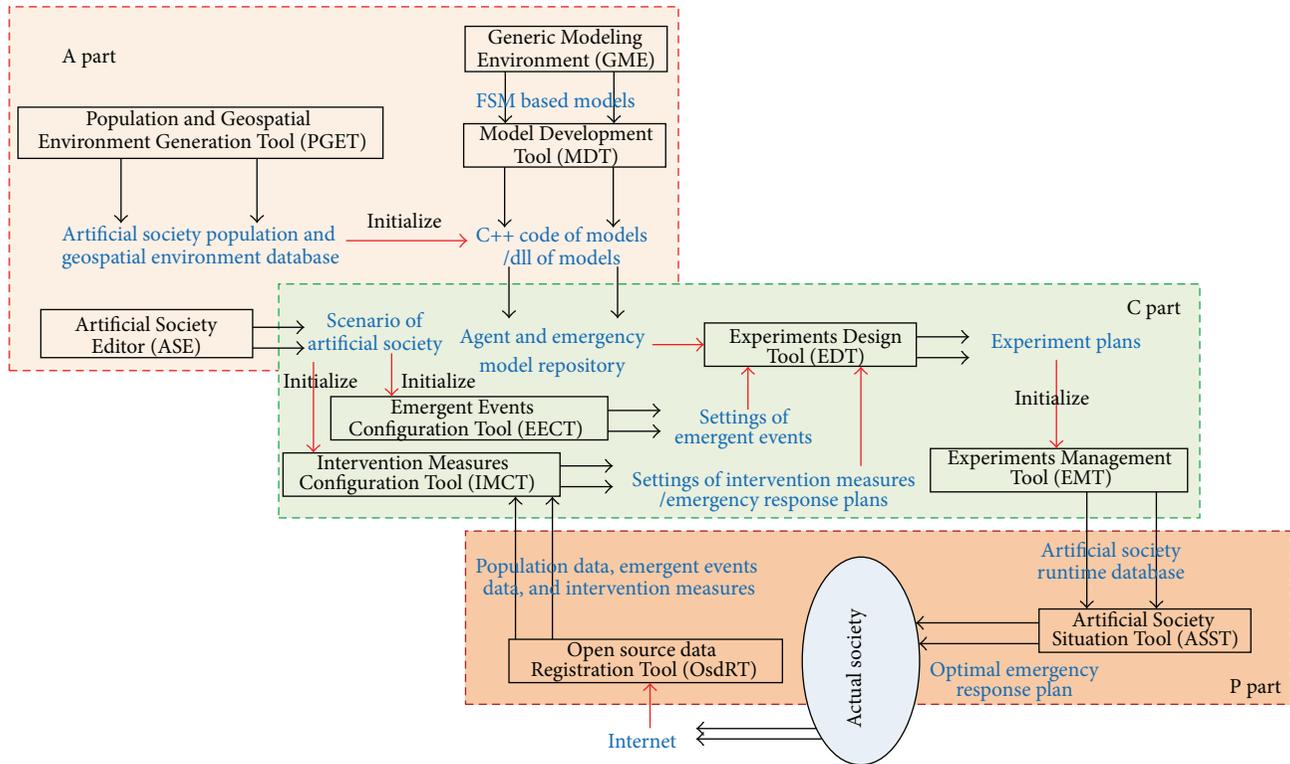


FIGURE 2: The software architecture of KD-ACP.

In the “A” part, Generic Modeling Environment (GME) [25] and Model Development Tool (MDT) are the kernel tools in the modelling of artificial society. GME is an open source modelling tool which supports domain-specific modelling. The domains of artificial society are created by GME in our work. Models such as agent, environment, emergent event, and intervention measure are described in specific domains first in GME. With the help of model transformation, these models are all transformed to the Finite State Machine (FSM) models. Meanwhile, code generations are supported by MDT, and these models are all implemented in C++. Artificial Society Editor (ASE) is used to describe the concrete scenario of actual society, which defines the scope of models set for artificial society; Population and Geospatial Environment generation Tool (PGET) generates the population and geospatial environment data with the support of statistical data from actual society.

In the “C” part, Emergency Events Configuration Tool (EECT) initializes the models of emergent events while Intervention Measures Configuration Tool (IMCT) initializes the models of intervention measures. Experiments plans are generated by Experiments Design Tool (EDT). Based on these plans, Experiments Management Tool (EMT) is used to run and manage the computational experiments to study the emergency problems.

In the “P” part, Artificial Society Situation Tool (ASST) seemed as the monitor of running artificial society. The statistical data and situation are shown by ASST at runtime. In the meantime, the emergency response plans are made by

emergency decision organizations. Parts of the influences of emergency plans are reflected on Internet. Open source data Registration Tool (OsdRT) is used to register the open source data from Internet to artificial society.

KD-ACP is developed using the Browser/Server architecture, the tools are integrated in the home page of KD-ACP as shown in Figure 3. Each tool is activated by the click on the link. For example, Artificial Society Editor is started when the link of ASE is clicked. The working environment and programming languages of tools in KD-ACP are listed in Table 1.

Moreover, the implementation of KD-ACP is mainly composed of modelling phase and computational experiments phase. It will be discussed in the next section.

3.3. The Modelling of Artificial Society in KD-ACP. It is a critical problem to focus on the key parts of society in social computing. Based on the ACP approach, the bottom-up modelling is used to build the artificial society. As a result, modelling of artificial society is composed of three basic elements: agents, environments, and rules for interactions. However, we still meet the problem that specific features should be supported in artificial society. For example, emergent events and intervention measures are required in artificial society for emergency management. The modelling of only basic elements cannot cover the specific features in domains. Therefore, domain-specific modelling [26] is introduced to solve the problems in modelling artificial society.

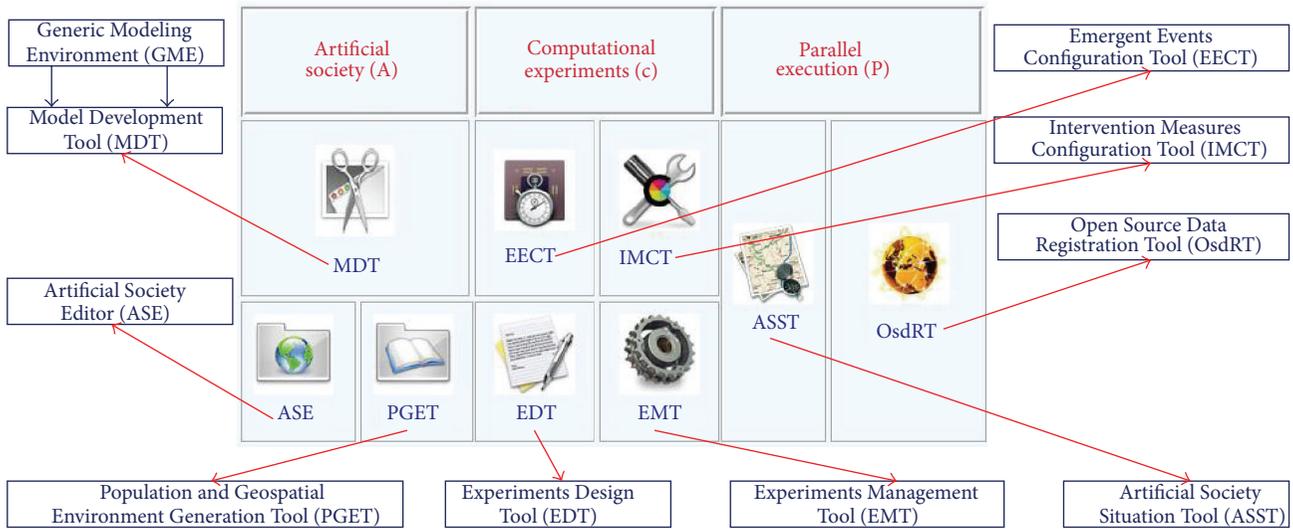


FIGURE 3: The implementation of KD-ACP.

TABLE 1: The working environment and programming languages of tools.

Tools	Working environment	Programming languages	Development platform	User type
Generic Modeling Environment (GME)	General computer (desktop application)	NULL	NULL	Domain experts
Model Development Tool (MDT)	General computer (desktop application)	C++	Visual Studio	Model developers
Artificial Society Editor (ASE)	General computer (desktop application)	C#	Visual Studio	Domain experts
Population and Geospatial Environment generation Tool	General computer (desktop application)	C#	Visual Studio	Domain experts
Emergency Events Configuration Tool (EECT)	General computer (with Internet Explorer, client side)	C#	ASP.NET	Domain experts
Intervention Measures Configuration Tool (IMCT)	General computer (with Internet Explorer, client side)	C#	ASP.NET	Domain experts
Experiments Management Tool (EMT)	General computer (with Internet Explorer, client side)	C#	ASP.NET	
Server of EMT	Nodes in supercomputer (Console Program, server side)	C++	Visual Studio	Users of computation experiment
Runtime Infrastructure of EMT	Nodes in supercomputer (Console Program, server side)	C++	Visual Studio	
Artificial Society Situation Tool (ASST)	General computer (desktop application)	C++	Visual Studio	Domain experts
Open Source Data Registration Tool (OsdRT)	General computer (with Internet Explorer, client side)	Java	JSP	Domain experts

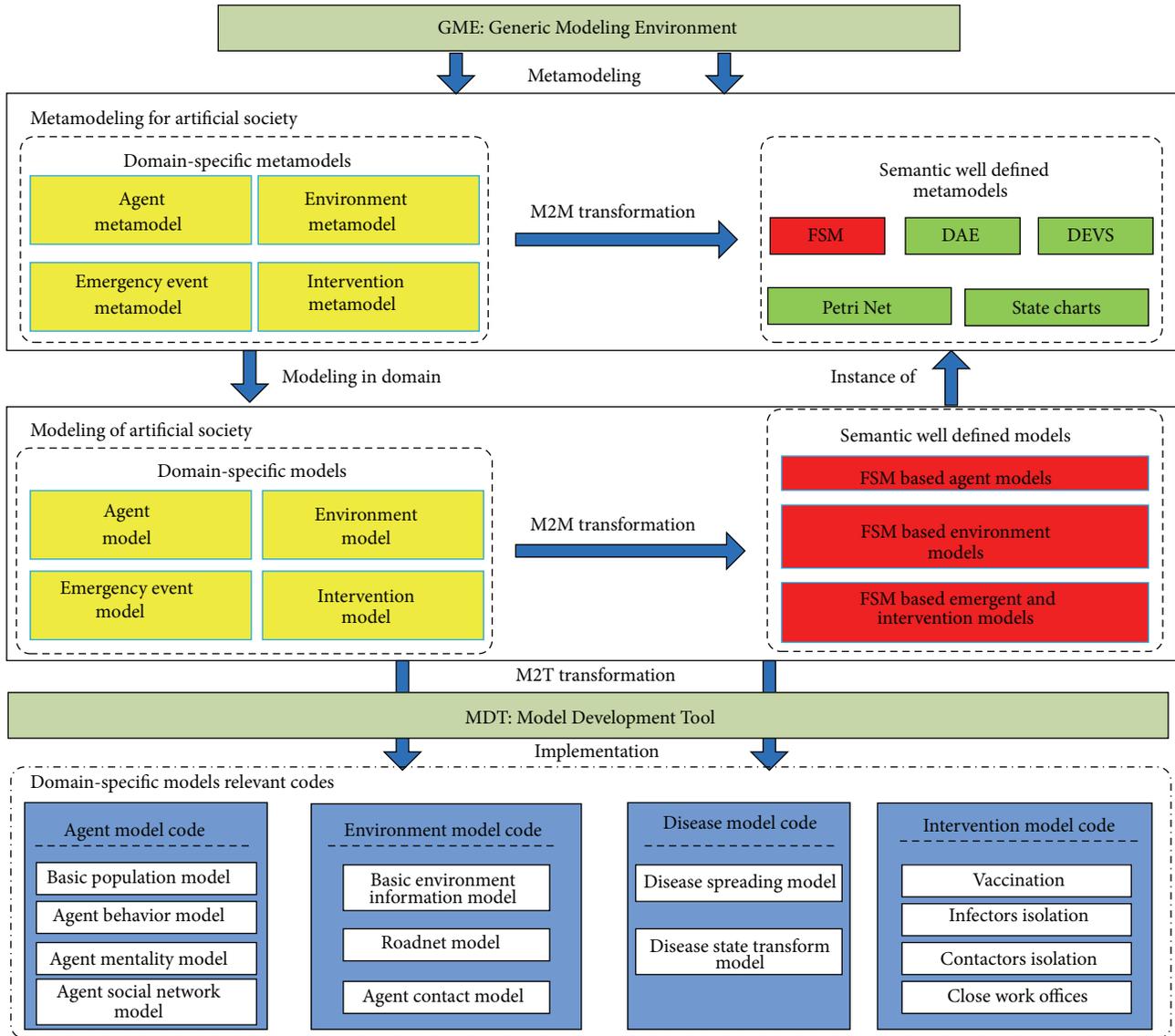


FIGURE 4: From metamodeling and modelling by GME to implementation by MDT of artificial society.

3.3.1. The Principle of the Modelling of Artificial Society.

According to the principle of domain-specific modelling, the modelling of artificial society contains the following steps: first, metamodeling the basic elements of artificial society; second, modelling the specific features in domain of emergency management; third, implementing the models of artificial society in codes. The whole process is illustrated in Figure 4. The first and second steps are implemented in GME while the third step is implemented in MDT.

The first step is metamodeling, which mainly focuses on constructing the metamodels of artificial society. Metamodeling tries to study the common patterns of artificial society. The outputs of metamodeling are metamodels, which represents the abstraction of the whole system. The basic elements of artificial society are described in metamodel. The process of metamodeling is divided into four phases. The first is the construction of the domain-specific metamodels. As shown

in Figure 4, agent metamodel, environment metamodel, emergent event metamodel, and intervention metamodel compose the metamodel of artificial society. The second is the construction of the metamodels described by typical modelling formalisms such as FSM, DAE, DEVS, and Petri Net [27]. These formalisms are all semantically well defined. The third is the definition of the model transformation from domain-specific metamodels to metamodels of typical modelling formalisms. The transformation standardizes the metamodels of artificial society by typical modelling specifications. The fourth is the definition of the transformation templates from metamodels to code framework. The templates list the basic abstract interfaces of metamodels of artificial society. These abstract interfaces are implemented in the specific-domain modelling and code generations.

The second step is modelling; the models of artificial society such as agent model, environment model, emergent

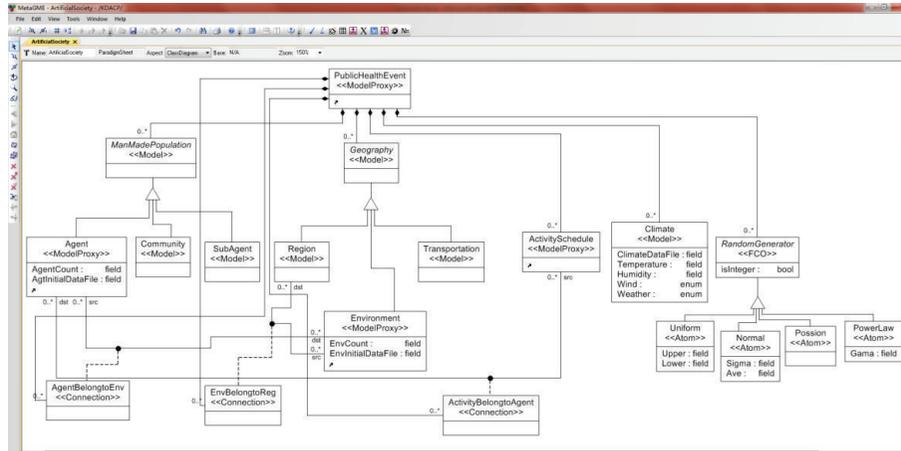


FIGURE 5: The metamodels of artificial society in GME.

event model, and intervention model are built. Actually, the models are the instantiation of metamodels in the last step. Different from the general modelling environment like UML [28], the domain-specific modelling provides a familiar modelling environment for the domain experts in artificial society. For example, emergency response experts only concern emergent event model and intervention model inherited from metamodels. After constructing the domain-specific models based on domain-specific metamodels, domain users execute the model transformation defined in the first step. All the models of artificial society are transformed into FSM models. As a result, the models are implemented in this unified modelling formalism (FSM). The model transformation makes the simulation of the models possible.

The third step is the generation of executable codes of models. The executable code framework is generated by mapping template from metamodels to code framework defined in the first step. Moreover, domain developers also add necessary codes to the framework to integrate the dynamic semantics of the models. The code framework outputs the dynamic link libraries by compiling. The dynamic link libraries are loaded in the large scale artificial society runtime infrastructure in computational experiments.

3.3.2. The Metamodelling and Modelling of Artificial Society by GME. GME is used to build metamodels and models in our work. As mentioned before, the abstraction and common patterns of society are represented in metamodels. According to the bottom-up modelling style, metamodels of agent, environment, and communications are described in GME. Figure 5 shows part of metamodels of artificial society. The features of an agent metamodel are extracted from the census figures and statistical data. Environment metamodel simulates the geospatial places for the behaviors of agents. The metamodel of communications among agents is modeled to simulate the interactions such as infection in epidemics and rumor propagation in public opinion formation events. It is worth notifying that the metamodel of communications

includes both the emergent event metamodel and intervention metamodel.

From the perspective of modelling, the details from specific domains are considered in the models by the instantiation from metamodels of artificial society. For example, social relationships based on complex networks are added in agent model to support the communications. Agent activity is also used to quantify the agent activity under different scenario. Environment models are linked with the help of transportation services; subways and roads are modeled while the path search is encapsulated in the services. Emergent event model and intervention model are also the domain-specific models. The modelling of artificial Beijing in GME will be discussed in detail in next section.

3.3.3. The Implementation of Models of Artificial Society by MDT. As mentioned before, MDT is used to implement models such as agent, environment, emergent event, and intervention. According to the template of code framework, the implementations of models are generated by MDT. The implementations are classified into two categories: FSM models and services. FSM models such as agents and environments are built under the specification of Finite State Machine (FSM) [29] in MDT, while all the services such as transportation are encapsulated under the Public Service Standard. This standard provides a generic interface specification for modelers to encapsulate public common services in artificial society. FSM models like agent are built statistically from the quantitatively analyzed characteristics, such as demographic attributes, social behaviors, emergency behaviors, and social networks. Social behaviors describe the daily behaviors of individuals while emergency behaviors describe the individual behaviors in emergent events. For example, infected individuals are all isolated in hospital in SARS. Isolation is modeled as a typical emergency behavior in our work. Correspondingly, services are used to simulate the macroactual society. Take transportation service for instance; the path search is needed by almost every agent during moving from spot to spot.

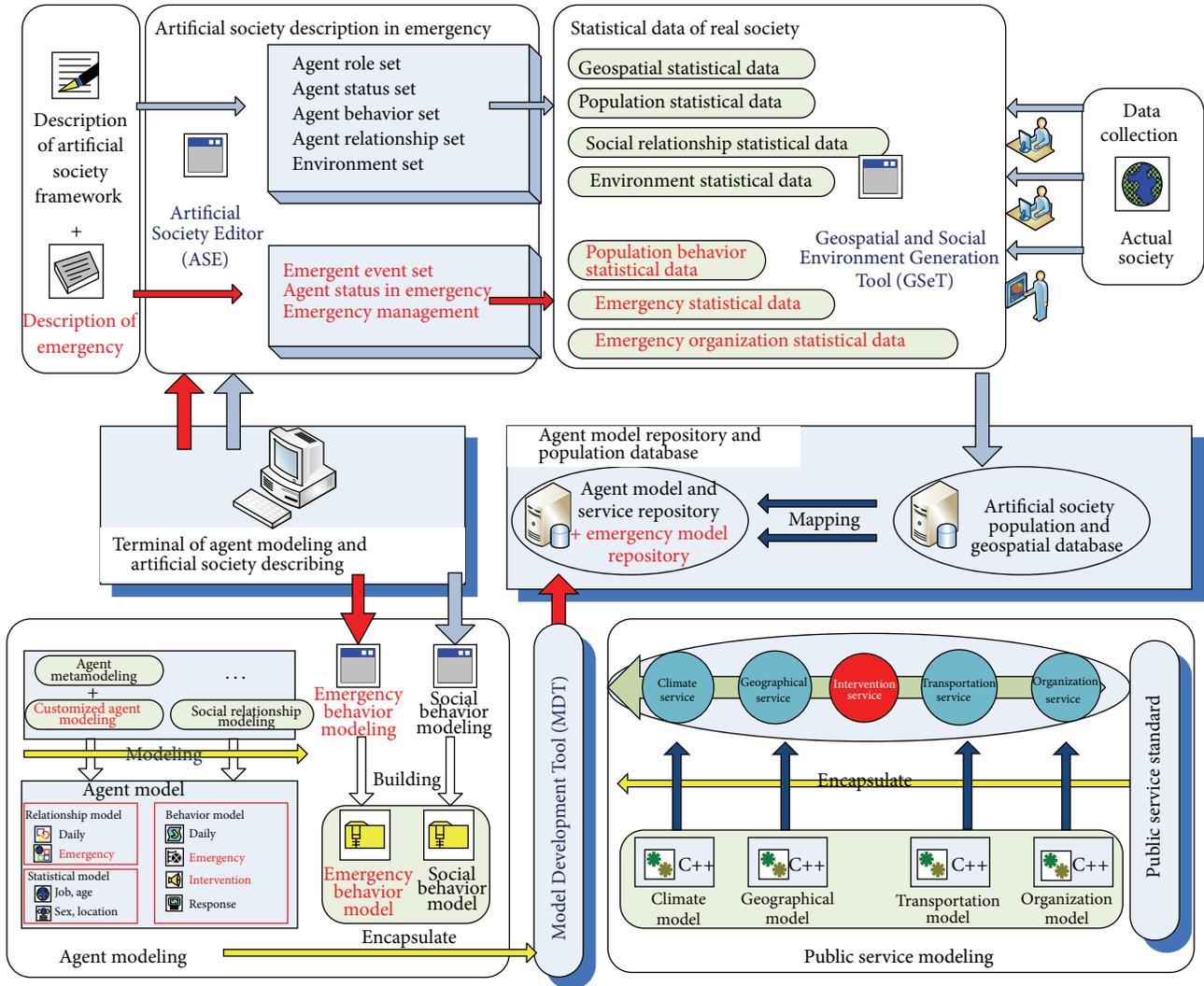


FIGURE 6: The editing and initialization of KD-ACP.

MDT provides domain experts with a tool to obtain the code implementations of models. With the help of compile environment like visual studio, MDT also supports the further programming development of the specific domain details which cannot be described in modelling step.

Both the FSM models and services are developed by the MDT first and then stored in the agent model and service repository. The repository manages the models according to the requirements from emergency problems and provides the models for EDT to make the experiment plans.

3.4. The Editing and Initialization of Artificial Society by ASE and PGET. As shown in Figure 6, ASE is used to edit the scenarios of artificial society within emergent events. The editing is composed of two parts: (1) the statistical information of artificial society in daily life, such as the roles of agents, the relationships of agents, and the types of environments, and (2) the statistical information of artificial society in emergency, including the statistical data of emergent

events, the emergency organization, and emergency related behaviors of agents.

According to the requirements of the editing, these statistical data are collected from actual society manually by the domain experts. Based on these statistical data, PGET generates the artificial society population and geospatial environment database. The database supports the instantiation of artificial society at individual level. For example, the attributes such as age and gender of each agent can be found in the database. With the support of the database, FSM models, and service repository discussed before, it is sufficient for domain experts to study the emergency problems by computational experiments.

3.5. The Computational Experiments and Parallel Execution in KD-ACP. The tools of “C” part and “P” part in KD-ACP are used to support the process of computational experiments and parallel execution. The working process is shown in Figure 7; EECT and IMCT are both the starting and ending

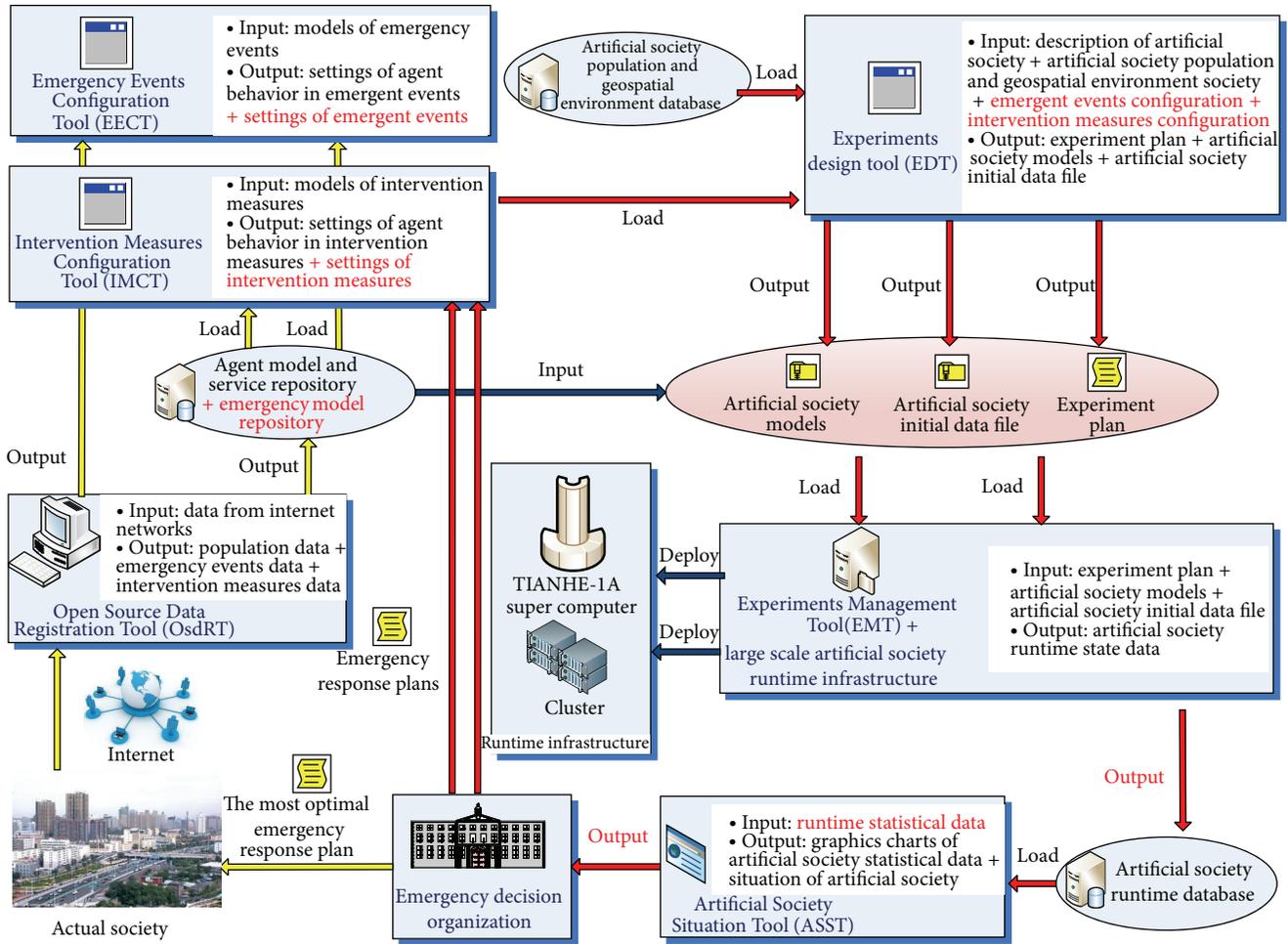


FIGURE 7: The computational experiments and parallel execution of KD-ACP.

point. The emergent events and intervention measures are configured by EECT and IMCT, respectively. The configurations of emergent events are used to simulate both the real emergencies like SARS and H1N1 and the supposed emergencies for experiments. Similarly, the configurations of intervention measures are also used to reproduce the real one and simulate the supposed one. The repeat of the emergency is used to verify the models while the supposed configurations are used to obtain the optimized decision plan to the response of the possible emergencies.

With the input of artificial society model and service repository, artificial society population and geospatial environment database, and the configurations discussed before, EDT generates the experiment plans to meet the requirements of research on emergency management. The output of EDT includes artificial society models, artificial society initial data files, and experiment plan. The models are downloaded from the repository while the data file is the collection of data from the database to initialize the models. When the models and data files are ready, EMT loads the experiment plan and deploys the models and data to the cluster or TIANHE-1A supercomputer [30] which was the world's fastest supercomputer built by National University of

Defense Technology (NUDT) in China in 2010. According to the plan, the experiments are done repeatedly on the large scale artificial society runtime infrastructure [31, 32] by the multisample settings. The work process is the implementation of computational experiments in ACP approach.

Traditionally, emergency response plans are made by emergency management theories and experiences. The only way to test the effective of plans is the feedback results of real world. ACP approach provides a novel method to support emergency response plan making by parallel execution. As shown in Figure 7, the work process of KD-ACP is composed of two loops. The inner loop composed by red arrows describes the process of computational experiments while the outer loop of yellow arrows illustrates the process of parallel execution. During the runtime of computational experiments, the statistical data of artificial society is collected and stored in the artificial society runtime database. Based on the database, ASST outputs the customized situation of running artificial society by graphics charts and situation maps. The information is sent to the organizations of emergency decision to support making the emergency response plans. With the help of computational experiments loop, these plans are simulated repeatedly to find the most optimal one.

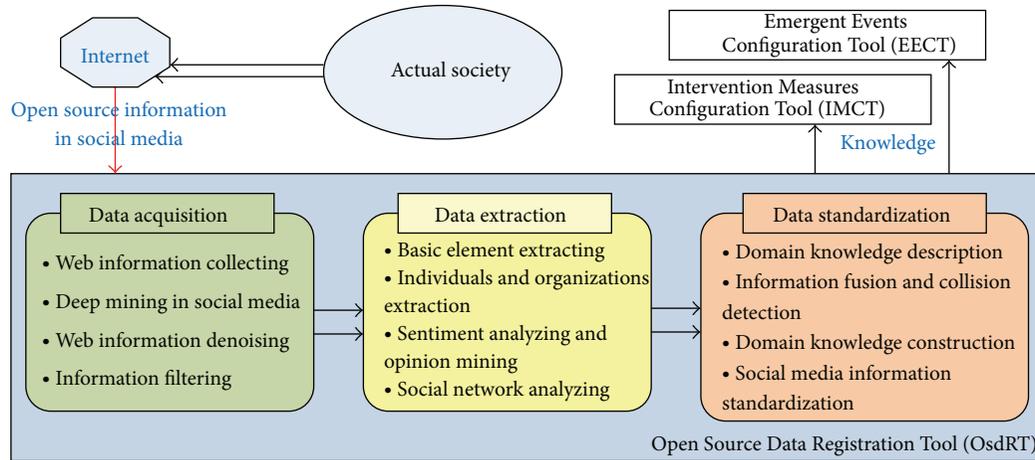


FIGURE 8: The compositions of OsdRT [34].

Moreover, the most optimal plan is used to the response of emergency in actual society. According to the idea of parallel control in [24], the feedback of actual society is partly collected from Internet networks by OsdRT. As shown in Figure 8, OsdRT is composed of three components: data acquisition, data extraction, and data standardization. Data acquisition collects, mines, and filters information from social sensing networks. Data extraction includes basic element extracting, individual and organization extracting, sentiment analyzing, and social networks analyzing. Data standardization specifies the useful knowledge and sent it to configuration tools in KD-ACP.

By processing in OsdRT, the knowledge about emergent events and intervention measures are analyzed first and registered in the EECT and IMCT. The registration updates the settings of emergent events and intervention measures. This loop composed of yellow arrows implements the parallel execution in ACP approach. The implementation of parallel control and management provides a data-driven approach that considers both the engineering and social complexity for modelling, analysis, and decision making in emergency management.

4. Modelling Beijing City with KD-ACP

4.1. How to Build the Artificial Beijing. According to the modelling of artificial society in KD-ACP discussed before, the Beijing city is modeled as follows.

To meet the requirements of emergency managements, the basic elements of artificial society are extended. As shown in Figure 9, six elements are required to simulate the city: agents, environments, transportation, activity schedule, communication, and agent activity.

4.1.1. Modelling Artificial Beijing. Figure 10 shows the main GUI of GME for the modelling of artificial modelling in public health events. Metamodels are listed in the left area of Figure 10; the list provides basic syntax elements for domain experts to model artificial society. Domain experts build

models based on the knowledge of their own. Meanwhile, GME supports hierarchy for building large scale systems. The syntax symbol listed in GUI can be extended in new tab by double clicking. Take agent for example; the model of agent can be detailed by edition in another tab page of agent.

As shown in the center of Figure 10, the models of artificial Beijing consist of five parts: models of agent and environment, domain models of public health events, intervention models, controller models, and services. Agent model describes individuals in society; it is composed of basic population information, action, social relationships, activity schedule, and disease related information. Activity schedule represents individual's physical action model, focusing on the daily action of agents. Environment model includes physical entities such as buildings, playground, transportations, and agents contained in environment. Domain models of public health events are composed of the propagation model of disease, disease state transition model, and so on. Intervention models include the settings of vaccination, isolation, and so on. The models mentioned before are all FSM models. The mechanisms of these models will be detailed in next sections. Controller models and services are the public service modules; they are implemented in the development in MDT.

4.1.2. Modelling Agents and Environments. Under the specification of FSM, agent and environment models are implemented in two parts: the state space and state transitions. The state space is composed of the demographic attributes and behavior related attributes. The transitions are triggered when the conditions of states are satisfied. As shown in Figure 11(a), the action of agent is changed when the "next time" condition is satisfied in agent model while the agents list is changed when the agent arrival condition is satisfied in the environment model.

4.1.3. Modelling Activities. Agent activities come from the agent state transitions of actions such as movements and communications. The actions of agents are instructed by the activity schedule shown in Table 2. Activity schedule lists

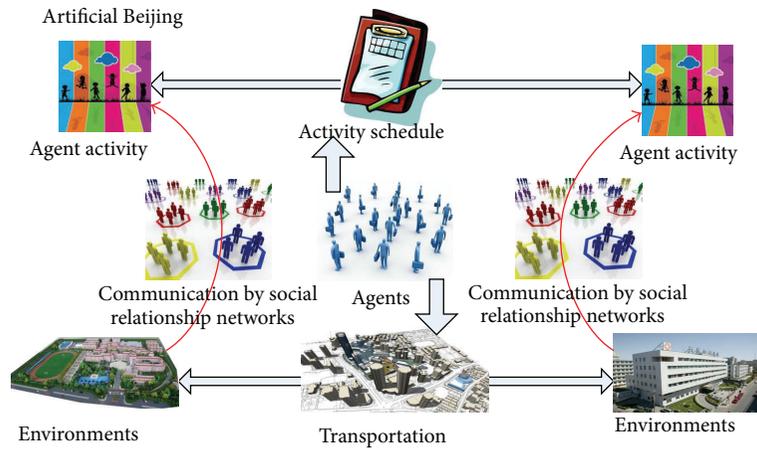


FIGURE 9: The basic elements of artificial society.

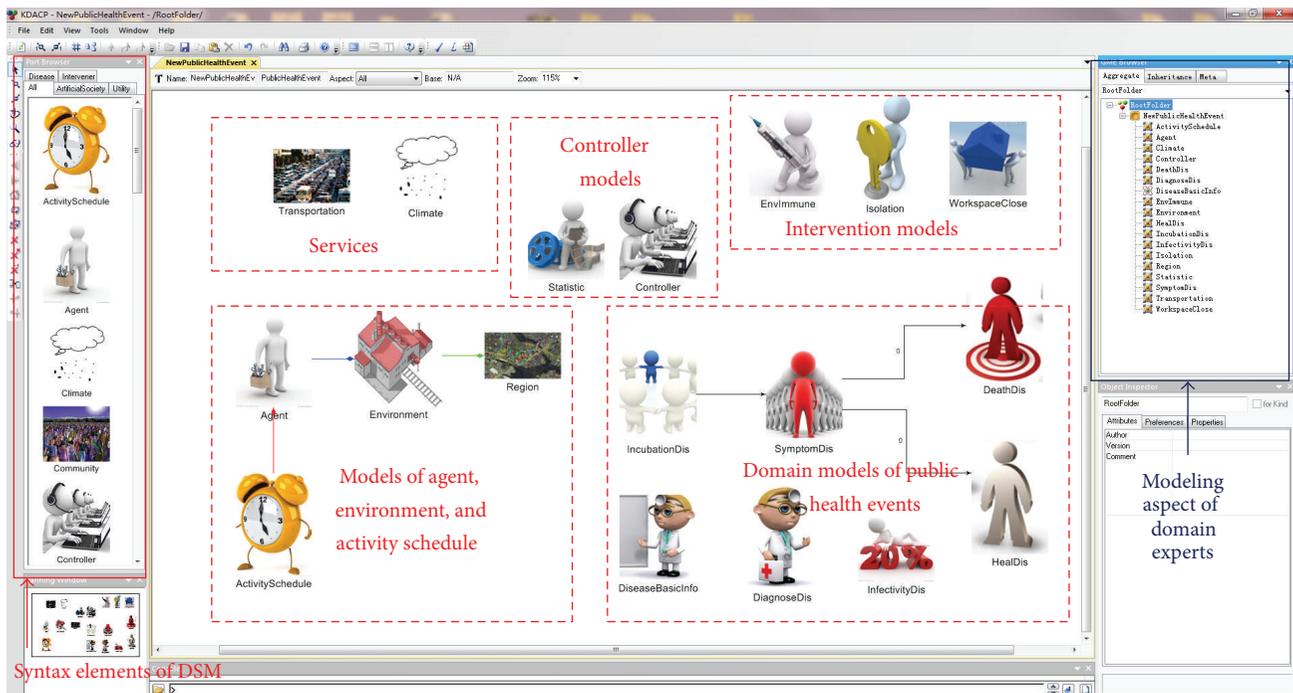


FIGURE 10: The models of artificial Beijing in GME.

all the actions with probability in one day for agents in both normal and emergent situation [38]. There are several types of activity schedule in artificial Beijing: student agent activity schedule, worker agent activity schedule, emergent agent schedule, and so on. For example, Table 2 gives an agent activity schedule. Upon the instruction of activity schedule, student agent changes the actions by p_i after state transitions. The p_i in the table means the action probability in the relevant period. In the duration from 08:00 to 12:00, a student agent either goes to classroom to have class or goes to library to study. The probability of class action is p_2 while the probability of study action is $1 - p_2$.

Agent behaviors are decided by the settings of activity schedule. In addition to the daily activity schedule mentioned before, emergent activity schedules are also considered in our work. Take public health events for instance; an infected agent changes schedule from a normal one to emergent one. The workflow of a susceptible agent is illustrated in Figure 12 to show the change of behaviors. After the infection, the agent is set in incubation phase. Not all the incubation agents will become symptomatic. Some of them turn back to being susceptible and some of them become symptomatic. The symptomatic agents change their activity schedule from normal to emergent. In the emergent case,

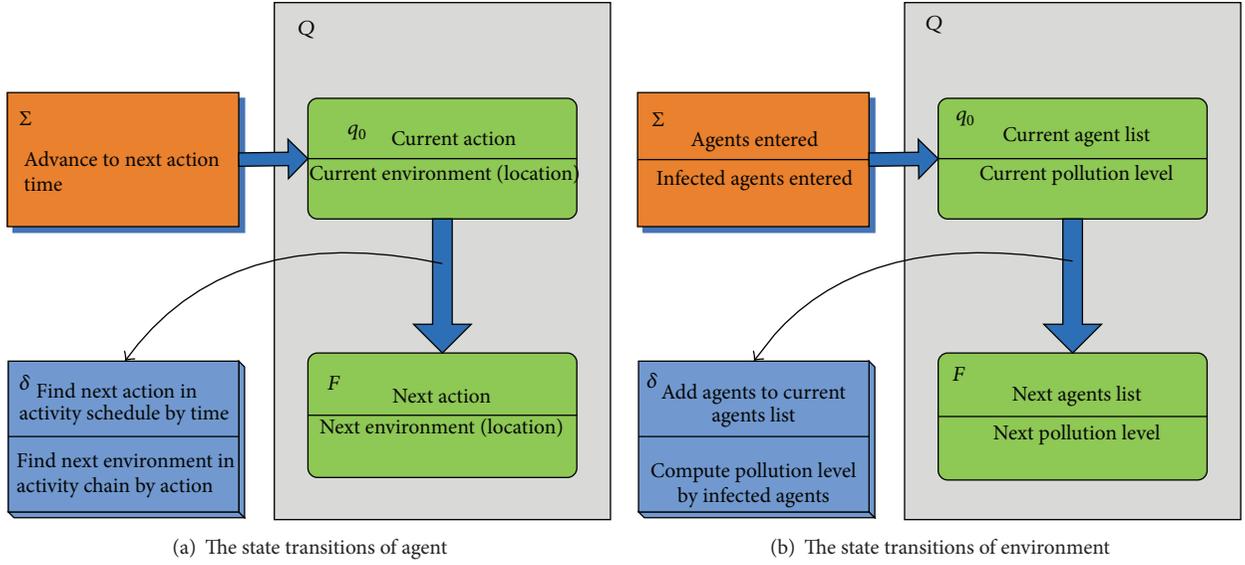


FIGURE 11: The state transitions of models.

TABLE 2: The activity schedule of a student agent.

Duration (Δt)	Activity	Location	Probability	T_{Location} (minute)
00:00–06:00	Sleep	Dormitory	P_0 (1.00)	360
06:00–08:00	Breakfast	Dormitory/restaurant	P_1 (0.68)	120
	Sports-breakfast-travel	Playground-dormitory/restaurant	$1 - P_1$ (0.32)	
08:00–12:00	Class	Classroom	P_2 (0.77)	240
	Study	Library	$1 - P_2$ (0.23)	
12:00–14:00	Lunch	Restaurant	P_3 (0.90)	120
	Lunch-shopping	Restaurant-convenience store	$1 - P_3$ (0.10)	
14:00–18:00	Class	Classroom	P_4 (0.70)	240
	Study	Library	$1 - P_4$ (0.30)	
18:00–19:00	Dinner	Dormitory/restaurant	P_5 (0.63)	60
	Sports-shopping	Playground/convenience store	P_6 (0.25)	
	Study	Classroom/library	$1 - (P_5 + P_6)$ (0.12)	
19:00–22:00	Rest	Dormitory	P_7 (0.45)	180
	Dinner-rest	Dormitory/restaurant-dormitory	P_8 (0.20)	
	Study	Classroom/library	$1 - (P_7 + P_8)$ (0.35)	
22:00–24:00	Rest	Dormitory	P_9 (0.72)	120
	Sleep	Dormitory	$1 - P_9$ (0.18)	

infected agents go to hospital according to the treatment schedule or stay in dormitory according to isolation schedule. After the treatment in hospital or self-immunoprocess, agents become healthy and immune of disease. If the agents are treated in the hospital, they are not allowed to get out until recovered. Moreover, activity schedules are also influenced by the emergency response plans. For example, in the case of isolation in emergency response plans, the activity schedule of agents who had contacted with infected is changed. Only the locations including home and dormitory compose this emergent schedule. Likewise, based on the statistical data from emergency response plans, the additional possibilities

are added into activity schedules. The behaviors of agents also change with activity schedules. In the view of ACP approach, the injection of new emergency response plans implements the parallel execution of microagent behaviors.

In another aspect, contact frequency is also another crucial element to determine the infected rate in public health event. It is important to model the contact frequency and contact time of individuals. Based on the studies on the contact behavior of human being by questionnaire survey, Edmunds found that the contact frequency of individual could be fitted approximately into a normal distribution, and the mean and the standard deviation distributions are

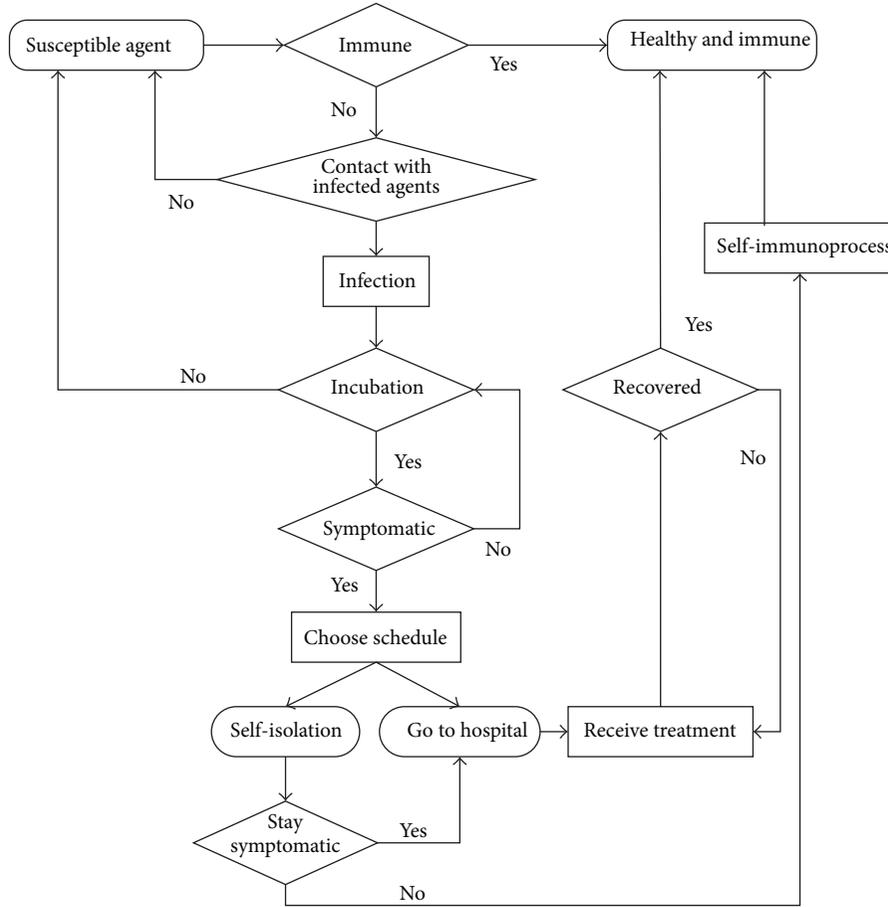


FIGURE 12: The workflow of a susceptible agent.

16.8 and 8.5 [39]. So it is possible to apply normal random variable to model the contact frequency. In our work, Box-Muller method [40] is used to generate the random variable of contact frequency, shown in the following:

$$F_C = \mu_F + \sigma_F(-2 \ln(\gamma_1))^{1/2} \cos(2\pi\gamma_2), \quad (1)$$

in which F_C is the random variable of contact frequency, μ_F is the mean value of normal distribution, σ_F is the standard deviation of normal distribution, and γ_1 and γ_2 are the uniform random variables distributed in the interval $[0, 1]$. Based on (1) and survey data [41], the contact frequencies of agent $F_{\text{contact}}(A_i)$ are discretized as in Table 3 within the consideration of activity differences.

Similarly, the duration per contact between individuals is another key factor of AHC transmission. The duration of contact could also be modeled by the normal random variable in the following [42]:

$$T_C = \mu_T + \sigma_T(-2 \ln(\gamma_1))^{1/2} \cos(2\pi\gamma_2), \quad (2)$$

in which T_C is the random variable of duration per contact, μ_T is the mean value of normal distribution, σ_T is the standard deviation of normal distribution, and γ_1 and γ_2 are the uniform random variables distributed in the interval

$[0, 1]$. In addition, because the time spent in specific location $T_{\text{Location}}(A_i)$ also follows the random distribution like (3) in our work. In order to make the duration of contact $T_{\text{contact}}(A_i)$ shorter than the $T_{\text{Location}}(A_i)$, the mean and standard deviation of (2) is set up in (3) and (4) [42]:

$$\mu_T = \frac{T_{\text{Location}}}{(\mu_F + \sigma_F)}, \quad (3)$$

$$\sigma_T = \frac{T_{\text{Location}}}{(10\mu_F + 10\sigma_F)}, \quad (4)$$

in which T_{Location} is the time of an activity in a specific location set in Table 2. μ_T and σ_T are the mean and standard deviation of contact frequency. As a result, the duration of agent $T_{\text{contact}}(A_i)$ in specific locations (listed in Table 2) is also discretized in Table 3 within the consideration of activity differences.

4.1.4. Modelling Transportation. In artificial Beijing, three kinds of travel models are considered: walking inside the district, travelling by road networks, and travelling by subway networks. According to the traveling models, the agent movements are implemented by the compositions of traveling models before. So it is important to build the basic road

TABLE 3: The contact frequency and duration in different locations and activities by social relationships in case of student and teacher.

Social relationship	Activity	Location	$T_{\text{contact}}(A_i)$ (minute)	$F_{\text{contact}}(A_i)$
Student-student	Sleep	Dormitory	0	0
	Breakfast	Restaurant	$N(5,2)$	$N(3,2)$
	Breakfast	Dormitory	$N(5,2)$	$N(2,1)$
	Sports	Playground	$N(2,1)$	$N(5,3)$
	Class	Classroom	$N(2,1)$	$N(2,1)$
	Study	Library	$N(2,1)$	$N(2,1)$
	Lunch	Restaurant	$N(15,8)$	$N(3,2)$
	Shopping	Convenience Store	$N(20,3)$	$N(2,1)$
	Dinner	Restaurant	$N(10,8)$	$N(2,1)$
	Dinner	Dormitory	$N(15,8)$	$N(3,2)$
Teacher-student	Rest	Dormitory	$N(6,3)$	$N(5,2)$
	Class	Classroom	$N(2,1)$	$N(2,1)$

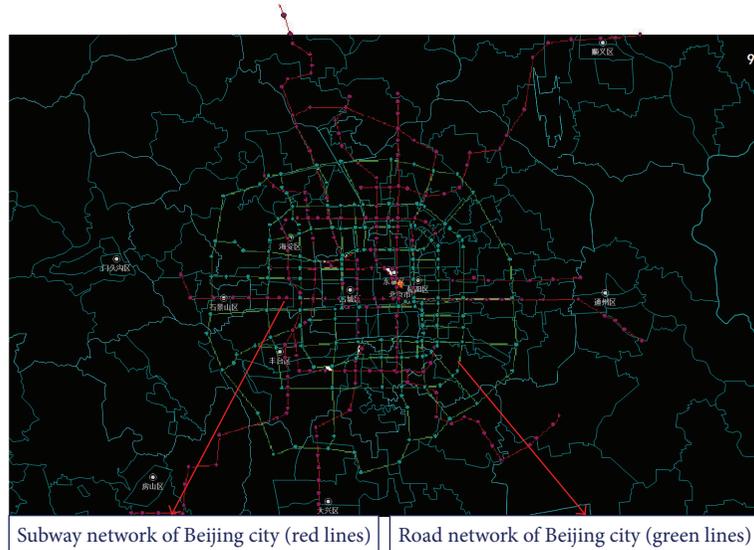


FIGURE 13: The implementation of transportation for artificial Beijing.

and subway networks. With the help of Google map, the key points of road are sampled first and the road networks are generated by the links among point. Similarly, subway networks are generated by the links between subway stations mapped from the real stations in Google map. Figure 13 shows the road and subway networks in artificial Beijing, with the road networks in green and the subway networks in red. Most of the districts in Beijing city are covered by these two networks. The algorithm of hierarchical route planning is proposed as follows.

- (i) Find the next position where agent will be located in next action.
- (ii) Obtain the traveling models from the statistical transportation data.
- (iii) Search the path from the starting position to the nearest road entry or subway stations.

- (iv) Search the optimal path from the starting station or road entry to the nearest station or road exit of target position.
- (v) Search the path from the target subway station or road exit to the target position.
- (vi) Attach the subway train number or the bus number including the transfer information to the path according to the timetable of subway or bus.
- (vii) Connect all the paths obtained before. Generate the path from the starting position to the target position.

Based on the algorithm for each agent in artificial Beijing, the transportation is simulated during the computational experiments. It is worth noting that the activities inside the buses or train cars are considered, because the agent communication during travelling is also necessary to be simulated.

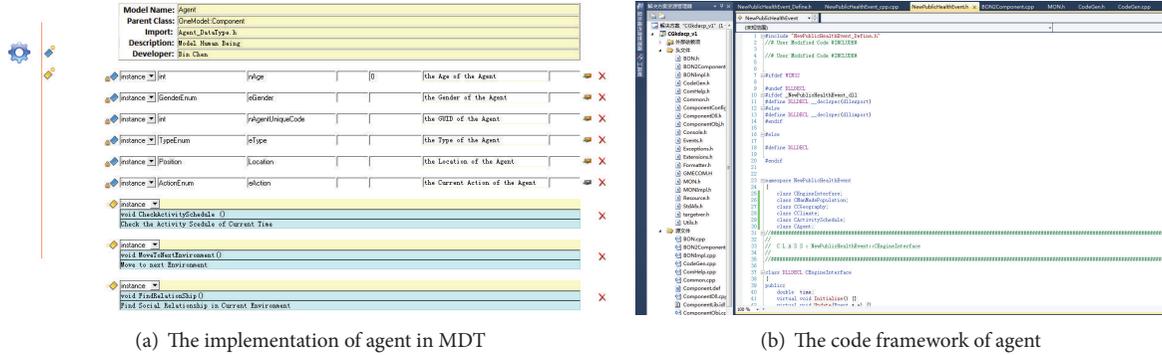


FIGURE 14: The agent model implementation in MDT [35].

4.1.5. *The Implementation of Models by MDT.* All the models are developed in the MDT; Figure 14 shows the GUI of the tool. MDT is used to implement the state space and state transitions of FSM models. The code framework is shown in Figure 14(b). Using the C++ inheritance features and the techniques of dynamic link library (dll), the models are encapsulated in the dll components. The flexible composition mechanism is designed to support the models evolution in computational experiments.

4.1.6. *The Description of Artificial Beijing by ASE.* As mentioned in Section 3.4, ASE describes the artificial Beijing in macroview. The statistical information of Beijing city is collected by ASE. Table 4 lists part of the description of artificial Beijing generated by ASE.

4.2. How to Obtain the Data of Beijing City

4.2.1. *Population and Geospatial Environment Database of Beijing City.* Based on the models developed for Artificial Beijing, it is necessary to generate the initial data to do the computational experiments. Because the individual level data are not available, it is necessary to construct an individual-based population database for both accurate computational experiment and determining optimal decisions.

According to the state spaces of models discussed in Section 4.1.2, the geospatial and population database of Artificial Beijing is designed as shown in Figure 15. The kernel part of database consists of the tables such as agent list table, geospatial environment list table, household list table, and agent distribution table. Agent list table is an individual level table used to store the data to initialize the state space of agent model such as id and gender age. Geospatial environment list table is also an individual level table which stores the data of environment attributes such as id, street id, and type. Household list table and agent distribution table store the statistical data for the data generation for agent and environment models. With the help of the geospatial and population database, the data generating of artificial Beijing is proposed in the next section.

4.2.2. *Generating Geospatial and Social Environment Data for Artificial Beijing by PGET.* The algorithm of generating geospatial and social environment data for artificial Beijing is used to quantify the spatial distribution of population and formalize geospatial behavior of each agent. The algorithm shown in Figure 16 is capable of generating a synthetic artificial society, which allows multiresolution statistical data, social interactive behavior, and multilayer social networks to be integrated together. The synthetic population can represent individual agents in the form of households and household members, and the synthetic population is statistically equivalent to a real population. For each household, characteristics, such as address, household size, family types and relationships, are generated. Each person is described by characteristics such as age, gender, social role, and correlated locations. The algorithm provides an effective methodology to reconstruct the computing environment in high resolution by using statistical data in low resolution, leading to better prediction and management of emergencies. The details of algorithm are illustrated in [36].

As shown in Figure 17, the implementation of the algorithm is embedded inside the GEPT. The collective input data such as the data files of population distribution is collected in the tool first; then the algorithm is activated to generate the database. The generation lasts almost twelve hours one time with the specified parameters settings. When the generation is finished, the data of artificial Beijing is obtained in the database with 19,610,000 agents and 16,000 environments.

5. H1N1 Experiments in Artificial Beijing with KD-ACP

With the support of the artificial Beijing modeled before, the computational experiments and parallel execution are also implemented by KD-ACP. The experiments are designed to find the most optimal emergency decision response plans. Epidemic in city is a typical scenario in emergency management. H1N1 epidemic in Beijing in 2009 is used to be the used case to test KD-ACP. According to Section 3.4, it is necessary to model emergency management first in order to support the computational experiments.

TABLE 4: The description of artificial Beijing generated by ASE.

Statistical features of artificial Beijing	Descriptions of statistical features	Data source/default values
Agent roles	{Baby, child, primary student, middle school student, college student, worker, . . . , retired}	Demographic data file of Beijing
Environment types	{Super market, store, restaurant, park, stadium, hospital, school, clinic, commercial building, dormitory building, . . . , residential building}	Demographic data file of Beijing
Demographic data	{Age distribution of population, sex distribution of population, household distribution of population, retired age distribution, . . . , children distribution}	Demographic data file of Beijing
Environment data	{Distribution of all types of environments}	Environment distribution data file of Beijing
Relationships	{Classmate, relative, college, friendship, . . . , family}	Demographic data file of Beijing
Varying parameters	Age difference of couples $N(\mu, \sigma)$	$N(0,3) \in [0, 12]$
	Minimum marriage age (male, female)	(22, 20)
	Age difference of children $N(\mu, \sigma)$	$N(1,3) \in [1, 10]$
	Enrollment rate of kindergarten α	0.8
	Enrollment rate of primary school β	1
	Enrollment rate of middle school γ	1
	Enrollment rate of college δ	0.8
	Entrance age of kindergarten α'	3
	Entrance age of primary school β'	7
Entrance age of middle school γ'	13	
Entrance age of middle school δ'	19	



FIGURE 15: The geospatial and population database of Beijing city.

5.1. Modelling Emergency Management. The modelling of emergency management consists of two parts: the modelling of emergent events and the modelling of intervention measures. H1N1 and the intervention measures for H1N1 are modeled in our case.

5.1.1. Modelling H1N1 by MDT. H1N1 Model in agent simulates the states transitions of health status and the relevant actions. Referring to SIR (Susceptible\Infective\Recovered) [43, 44] models, agent has three health statuses: susceptible, infected, and healthy with immunity. Only susceptible agent can be infected by the contact with infected agents. After

the three infected phases (incubation, being symptomatic, and recovery) of H1N1 model, agent is healthy again. It is worthy mentioning that the recovered agents can be infected iteratively when the immunity disappears gradually by time. The state transitions are illustrated in Figure 18.

According to [45], the latent period of H1N1 was in a Weibull distribution within people, from one to seven days [46, 47]. The distribution was usually centered in the range between one and three days. A Weibull random variable [40] is used to model the latent period, as described in

$$T_{Lat} = \alpha \times [-\ln(1 - \gamma)]^{1/\beta} + v (T_{Lat} \geq v). \quad (5)$$

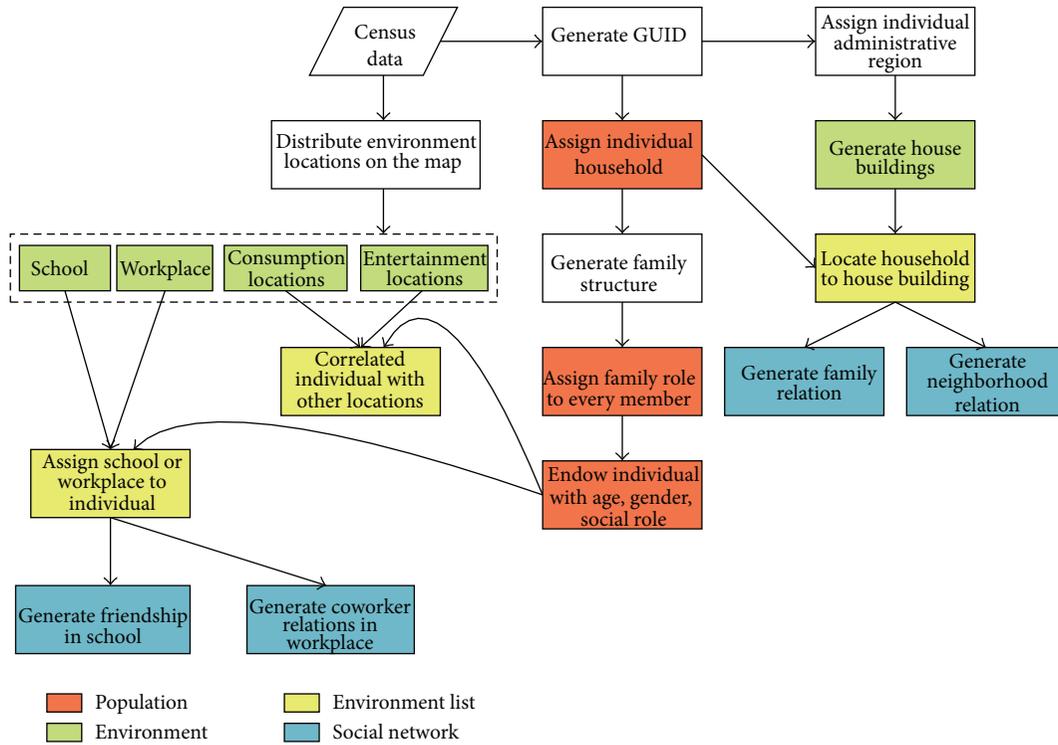


FIGURE 16: The algorithm of generating geospatial and social environment data [36].

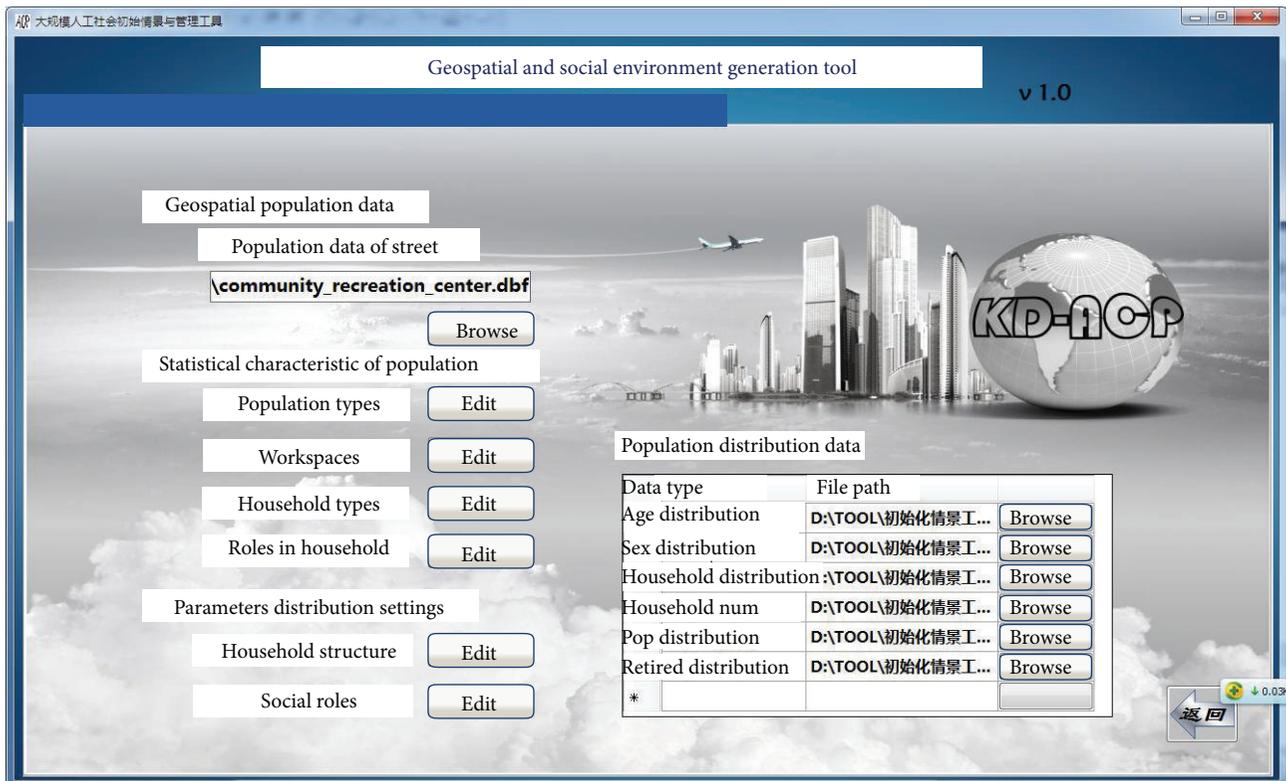


FIGURE 17: The GUI of Population and Geospatial Environment generation Tool (PGET).

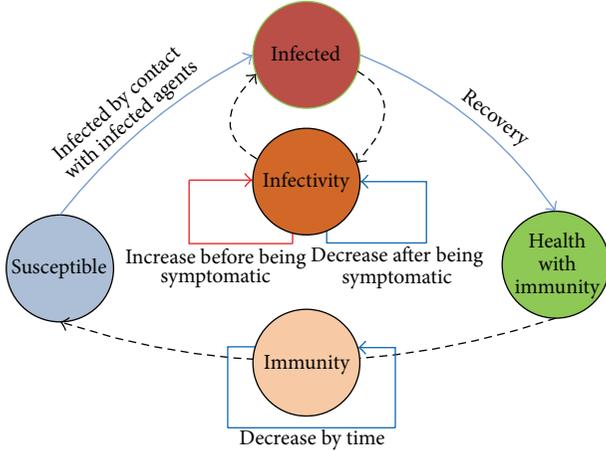


FIGURE 18: The transitions of agent health status.

T_{Lat} denotes the duration of latent period. ν , α , and β are the location parameter, scale parameter, and shape parameter of Weibull distribution, respectively. γ is a uniform random number in the range $[0, 1]$. According to statistics [47], ν , α , and β are set as 0, 1.8, and 1.21, respectively. Then the mean (standard deviation) of latent period is calculated as 1.59 days (0.58 day²). Furthermore, infectious period is set as 7 days.

A susceptible student's probability of getting infected depended on the infectivity of infectious agents, his own immunity level, the duration of the contact action, and so on. The infectivity of an infected agent evolved with time during the infectious period. According to the statistics of infectivity in the chart with red bars in Figure 19, an infected student had the highest level of infectivity in the second day after he had the first symptom. The infectivity levels in other days in comparison with the second day are in Table 5. The day labeled as “-1” means the day before the starting of the symptomatic period.

Based on the algorithm discussed before, the H1N1 model is developed with the help of MDT. The settings of influenza model will be discussed in Section 5.2.1.

5.1.2. Modelling Intervention Measures by MDT. In correspondence, intervention measures model is used to cease the outbreak of H1N1. As discussed in [42, 48, 49], if the appropriate measures are taken when the infectious disease emerges, the transmission of the disease could be slowed down and the damage could be decreased. Therefore, it is necessary and important to design emergency intervention measures in the artificial Beijing. According to the Ministry of Health in China, the intervention policies [50] are designed to control the spread of influenza such as H1N1. The intervention measures of these policies including are interventions activated time, vaccination rate, antibiotic, delay to hospital, isolation duration, close workspace duration, and limitation of activities listed in Table 6. Similarly, the models of these intervention measures are also developed with the help of MDT. Using the reasonable data ranges listed in Table 6, intervention models can be initialized to do the computational experiments. These experiments are used to show how

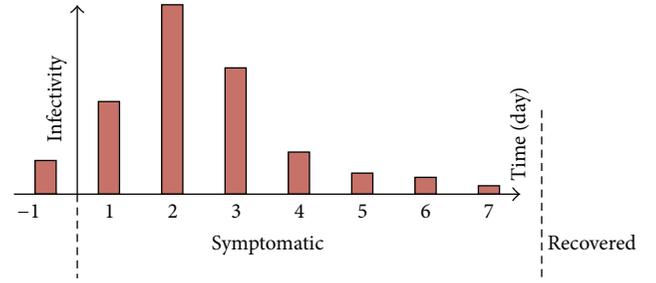


FIGURE 19: The distribution of infectivity rate of an infected agent [37].

TABLE 5: The proportion of infectivity.

Days (i)	-1	1	2	3	4	5	6	6
Proportion of infectivity	0.3	0.5	1	0.8	0.4	0.2	0.1	0.05

to restrain the H1N1 transmission. The settings of these intervention measures in IMCT are introduced in Section 5.2.2.

5.2. Experiment Settings

5.2.1. H1N1 Settings by EECT. EECT is designed to initialize the emergent events models such as H1N1. The settings come from the specified emergent event models. As shown in Figure 20, in the H1N1 used case, the settings are composed of the distribution of infection source, infection rate, infection period, and so on. The infection rate changes with environments and infection periods. The settings are initialized from both the research of medicine on H1N1 and the data of H1N1 outbreak in Beijing in 2009. The process of influenza in artificial Beijing is simulated to repeat the possible scenarios of H1N1 break. During the computational experiments of artificial Beijing, the H1N1 settings such as the period of being symptomatic are updated by the dynamic data registrations from OsdRT. The updating corrects the H1N1 model during the experiments. With the help of OsdRT, artificial Beijing is able to approach actual society. It is also an implementation of parallel execution.

5.2.2. Intervention Measures Settings by IMCT. In the H1N1 case, the settings of intervention measures are shown in Figure 21 in IMCT. These measures are designed under the instruction of Emergency Decision Organizations like China Disease Center (CDC). The measures such as vaccination rate, the rate of closed workspace, and the starting time can be initialized in the tool. Emergency response plans are implemented in these settings. As illustrated in Figure 7, all the possible compositions within value settings in reasonable intervals are tested to find the most optimal emergency response plan. For example, different compositions of measures such as vaccination rates and rates of workspace closed are used to design the plans of experiments. By the analysis of the results, the best composition of intervention measures can be obtained. The analysis will be discussed in case study in detail.

TABLE 6: The intervention measures of influenza.

Intervention measures	Values
Interventions activated Time	$T_{\text{Intervention}} \in [0, 80\text{th day}]$
Vaccination rate	$R_{\text{Vaccination}} \in [0, 1]$
Antibiotic rate	$R_{\text{Antibiotic}} \in [0, 1]$
Delay to hospital	$H_{\text{hospital}}(A_i), T_{\text{infected}} < t < T_{\text{recovered}} + T_{\text{HDelay}}, T_{\text{HDelay}} \in (0, 3\text{th day})$
Isolation duration	$I_{\text{isolation}}(A_i), T_{\text{Intervention}} < t < T_{\text{Intervention}} + T_{\text{IDuration}}, T_{\text{IDuration}} \in (0, 14\text{th day})$
Close workspace duration	$C_{\text{lose}}(E_i), T_{\text{Intervention}} < t < T_{\text{Intervention}} + T_{\text{CWDduration}}, E_i \in \text{Workspace}, T_{\text{CWDduration}} \in (0, 7\text{th day})$
Limitation of agent activity duration	$L_{\text{imitation}}(A_i), T_{\text{Intervention}} < t < T_{\text{Intervention}} + T_{\text{LAADuration}}, T_{\text{LAADuration}} \in (0, 7\text{th day})$

$R_{\text{Vaccination}}$ means the vaccination rate of artificial Beijing while $R_{\text{Antibiotic}}$ means the antibiotic rate used for infected agents in hospital. $H_{\text{hospital}}(A_i)$ means that the infected agent A_i is sent to hospital; he or she will receive the retreatment immediately. $I_{\text{isolation}}(A_i)$ means that the infected agent A_i is isolated at home; he or she can only contact with the other agents until isolation duration is over. $C_{\text{lose}}(E_i)$ means that the environment E_i typed of workspace is closed until the close workspace duration is over. $L_{\text{imitation}}(A_i)$ means that the activities of agent A_i are limited, only the workspace and home are allowed until limitation duration is over.

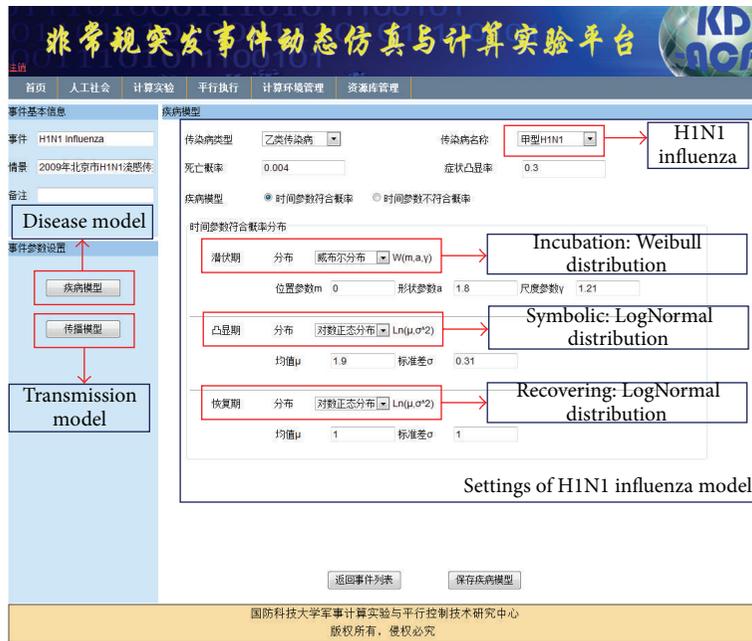


FIGURE 20: The settings of H1N1 in EECT.

It is worth notifying that the settings of intervention measures are also influenced by OsdRT. For example, the H1N1 outbreak areas are notified by OsdRT. The intervention measures such as close workspace are used only in the outbreak areas. Therefore, the unnecessary computation of intervention measures is avoided. So the data injection by OsdRT not only makes artificial society approach actual society, but also increases the performance of computational experiments.

5.3. Experiments by EMT. When the initialization of artificial Beijing is ready, EMT is used to do the computational experiments. EMT is composed of two parts: the central controller in the central node and the residential service deployed in the computational nodes. Central controller loads the experiment plans and controls the whole

experiments process; the GUI of controller is shown in Figure 22; EMT manages the computational experiments in four steps: cluster configuration, load experiment plans, models deployment, and running experiments.

5.3.1. Cluster Configuration. Before the configuration, cluster information is collected from the residential services inside the cluster. The information includes nodes amount, node names, node IPs, and CPU occupations. It is used to quantify the computing power of the cluster. The cluster configuration is based on this information. The nodes are selected for experiments. Then the LP (local process) [51] number for each node is set according to the number of the CPU kernels. After the configuration, the experiment plans are loaded to customize the models and data.



FIGURE 21: The settings of intervention measures in IMCT.

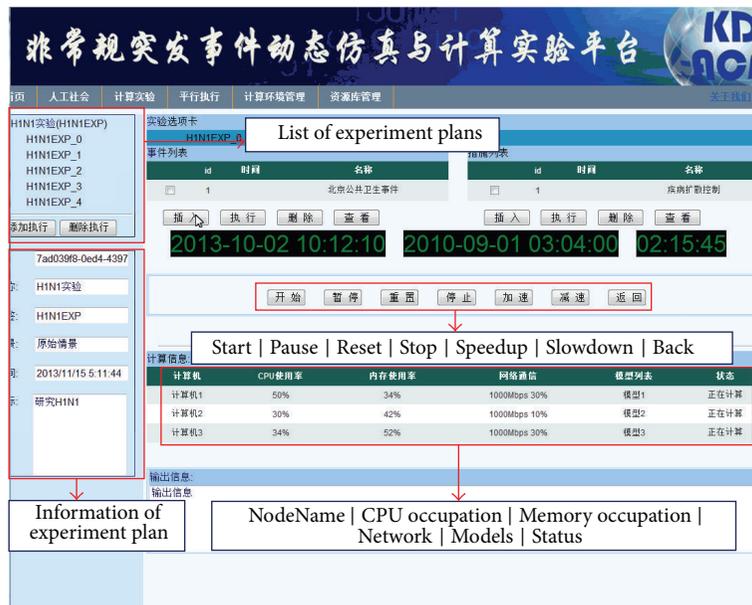


FIGURE 22: The deployment and running of experiments in EMT.

5.3.2. *Load Experiment Plan.* As discussed in Section 3.5, the description of experiment plans is composed of the information in three aspects: models and the initial data for the experiments, the deployment mapping tables from the models to the nodes, and the plan of experiment execution. When the experiment plan is loaded by the EMT, the file names and model descriptions of models and data are listed. With the support of cluster configuration, models can be mapped to the nodes in the GUI according to the computation requirements. EMT also provides the setting of experiments itself, including the running times, start time, and end time.

5.3.3. *Models and Data Deployment.* Models are stored in the agent population and services repository. According to the settings in experiment plan, the customized models are downloaded from the repository first; then the models such as agent model, environment model, emergent events model, and intervention measures model are integrated to be the composition models of artificial Beijing. The models and data locations in the nodes are configured in the “model path” and “data path” in the GUI of central controller. Finally the models are uploaded to the nodes in cluster with the consideration of partition. The partition of artificial society

models is different from the multiagent system in [52]. Models in artificial society are divided into three levels: agents and environments, grids, and city. Agents and environments are encapsulated in grids which compose the whole city. Thus the partition of artificial society becomes the partition of grids. The parallelism degrees are found to support the partition of models: agent independence and grid independence. According to the parallelisms, agents are partitioned to LPs according to the population distribution in grids. Agent movements among grids are simulated by the movement events among grids in different LPs. As a result, a two-tier parallel architecture is proposed to support the simulation of artificial society. The architecture is detailed in [53].

5.3.4. Running of Experiments. When the preparation is ready, the experiments can be done by the central controller of EMT. The running and control commands of the experiments are grouped into three aspects: start, pause, or stop the experiments; get the running information of the nodes such as the snapshot of the desktop of node. With the help of these commands, users can run the computational experiments to do the research on the emergency management. It is worth notifying that the simulation engine is optimized in order to support the ten million agent simulation. Using GPU as coprocessor, a two-tier parallel simulation engine is designed with support of MPI and OpenCL through phased synchronization [54]. One-sided communication is used for reflection of remote simulation objects and message passing between processes. A general kernel function prototype is elaborately designed and conditionally compiled for execution on both CPU and GPU. Moreover, the optimization operations like load balance is developed under the instruction of activity-based simulation [55]. The densely populated grids are given more CPU and memory resource according to the activity predictions.

5.4. Results of H1N1 Experiments with KD-ACP. On the basis of artificial society models and the intervention measures mentioned in Section 5.1.2, a series of computational experiments are performed to study the H1N1 influenza in Beijing. In our case of artificial Beijing, 19,610,000 agents are simulated in the cluster within 48 CPU cores and 128 G memories, the nodes are connected by kilomega networks. It takes 18 hours to simulate a 250-day disease spreading.

In order to illustrate how the optimal plans are obtained by KD-ACP, the experiments are divided into four groups: the experiments of model validation, the experiments of sensitivity analysis of vaccination rate, the experiments of sensitivity analysis of isolation, and the experiments of the combination of intervention measures. These four groups show the general process of the research on emergency management in H1N1 influenza. Firstly, the models of H1N1 are validated with the support of historical data. Secondly, the traditional medical interventions of pandemic such as vaccination are analyzed to find the optimal plan. Thirdly, the traditional nonmedical interventions of pandemic such as isolation are analyzed to find the optimal plan. Finally, the combination of interventions is tested to show the combined effect.

It is worth mentioning that all the experiments in our work are performed 100 times. These experiments are initialized with different settings of interventions in IMCT. The results shown in the figures are the mean values.

5.4.1. The Experiments of Model Validation. In our artificial Beijing, the models of H1N1 influenza are built based on the existing researches. Wang and her colleagues had studied the H1N1 influenza by SEIR (Susceptible\Exposed\Infective\Recovered) models [56]. With the support of their work, we build the models of artificial Beijing in the manner of multiagents.

In Figure 23, the control group shown in red line is drawn based on the realistic statistical data collected in H1N1 influenza in Beijing in 2009. The influenza lasts more than six months, and more than 170,000 people are infected. Comparing with the control group, the simulated results without interventions are drawn in blue. It is obvious that the simulated case fits the control group well before 70 days. But the ascending of control group is slowed after 70 days, and the peak value (17,883) is much less than the simulated results (25,135). It is because the vaccination is activated at day 67 in control group according to the intervention measures used in 2009. The restrain by vaccination becomes effective eight days later. The change of infected number after vaccination is shown by red spotted line. The trend of control group is different from the simulation data after 75 days.

However, the simulated results (blue line) in computational experiments fit the control group (as shown in Figure 23) in the first 70 days when no intervention measures are executed. The key features such as effective basic reproduction number (R_0) are also almost the same in both cases. Therefore, it can be concluded that the models of artificial society built in microview are validated by the realistic statistical data in influenza in Beijing in 2009.

In this group of experiments, no interventions are executed in order to validate the models. Therefore, the simulated results of this extreme case which cannot happen today could be obtained. As shown by blue line, the number of infected agents without interventions reaches peak value at day 91 with 25,135, consisting of 8,052 in symptomatic phase and 28,950 in recovery phase. It is obvious that the total number of infected agents grows slowly at the beginning time. Then the number increases quickly when the influenza breaks, and it reaches its maximum value with 383,870 in the end. This phenomenon can be explained as in the following. At the beginning of influenza, due to the limited number of infected agents, the propagation of the disease remains slow but many agents become susceptible. When the number of infected agents increases rapidly, the spreading becomes significant. However, the number of infected agents decreases after day 91. It is because the disease spreading relies on the social networks and spatial contact networks [49]. It means that the infected agents can only infect agents either in the social relationship networks or in the spatially contacted.

5.4.2. The Experiments of Sensitivity Analysis of Vaccination Rate. According to Section 5.1.2, vaccination is a typical

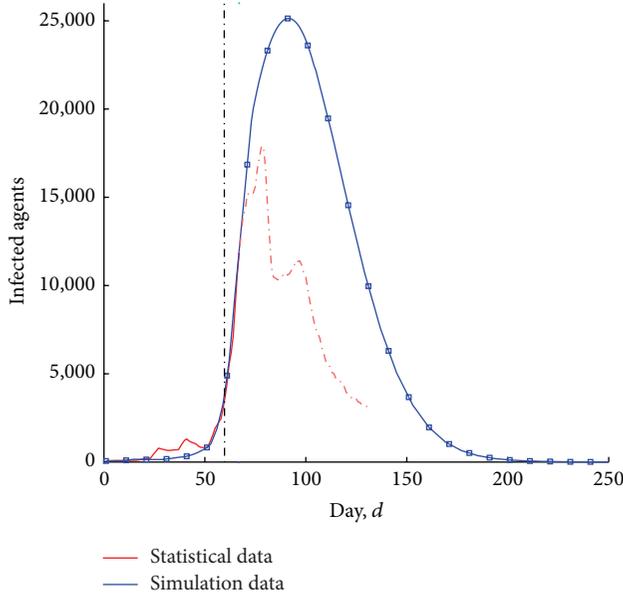


FIGURE 23: The results with no intervention executed, and the comparison with realistic statistical data in Beijing in 2009.

medical intervention in influenza. Traditionally, the effectiveness of vaccination can only be evaluated after the influenza, and the key settings such as vaccination rate are determined by the experience. It is always difficult for emergency management organizations to find the most optimal vaccination rate in advance. Therefore, the sensitivity of vaccination rate is analyzed in this group of experiments. The settings are listed in Table 7; average results are shown in Figure 24.

Vaccination rates are set linearly (10%, 30%, 50%, and 70%) to test the changes of influenza. The peak value of infected agents decreases from 25,135, 22,392, 15,787, 6,907 to 638, respectively. It shows that the more the agents are vaccinated, the smaller the peak value of the infected agents will be. In the case of 70% vaccination, the total infected agent is only 13,799. It means that the influenza in this case does not happen due to the high vaccination rate.

The change of peak values also shows an interesting phenomenon. Vaccination rates increase linearly; the peak values decrease slower than the ascending of vaccination rates at beginning, but the descending becomes more and more faster with the linear increase of vaccination rate. It means that the small scale of vaccination (less than 30%) does not make sense. With the increase of vaccination rate, the number of susceptible agents decreases significantly. In sequence, the possibility of infections is lowered greatly. Additionally, it could be found that the decrease of peak values comes later than the increase of the vaccination rates. This is quite reasonable according to Mei et al.'s work in [57].

As a result, it is concluded that high vaccination rate is effective in the control of H1N1 influenza, though costly.

5.4.3. The Experiments of Sensitivity Analysis of Isolation. According to Section 5.1.2, isolation is a typical nonmedical intervention. It is the most used intervention measure in

TABLE 7: The initial settings of experiments of vaccination.

Parameters	Value
Agent count	19,610,000
Initial infected count	40
Interventions activated time ($T_{\text{Intervention}}$)	0
Vaccination rate ($R_{\text{Vaccination}}$)	{0%, 10%, 30%, 50%, 70%}

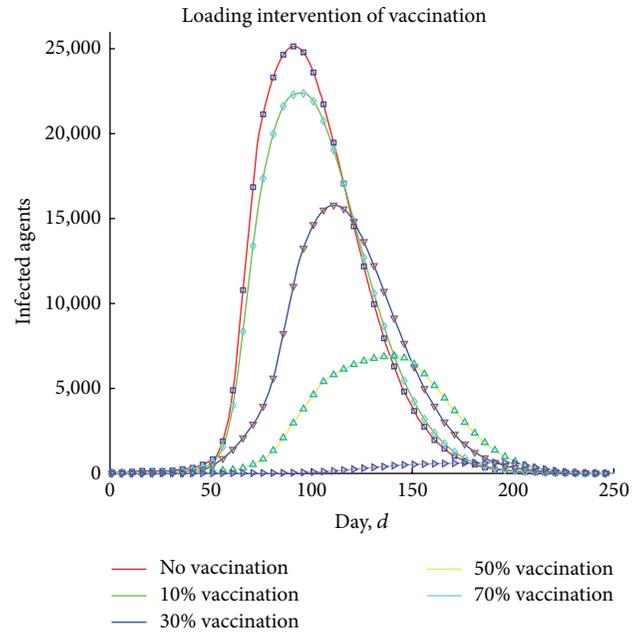


FIGURE 24: The results of loading intervention of vaccination.

the emergency management in influenza. As mentioned before, a problem also exists that the measure cannot be validated before the intervention. Therefore the sensitivity of isolation is analyzed in this group of experiments. It is worth mentioning that the isolation is usually combined with the hospital policy. It means the infected individual is sent to hospital; meanwhile the people who contact with the infected are all isolated. As a result, parameters delay to hospital and isolation duration are designed together in the settings of experiment. Delay to hospital stands for the period from infection time to hospital time while Isolation duration represents the period when an agent cannot contact others if he has contact with an infected agent. The settings of these parameters are listed in Table 8; average results are shown in Figure 25.

Firstly, the parameter delay to hospital is simulated alone to test the sensitivity. The simulated results are shown in solid line in Figure 25. The peak value decreases from 6,708 to 2,218 when the delay to hospital decreases from 1.5 days to 1 day. The half day change brings the 67% reduction. Moreover, the peak value almost approximates to zero in the case of 0.5 days. The influenza does not happen in this case. It can be concluded that the earlier the infected agents

TABLE 8: The initial settings of experiments of isolation.

Parameters	Value
Agent count	19,610,000
Initial infected count	40
Interventions activated time ($T_{Intervention}$)	80
Delay to hospital (T_{HDelay}), isolation duration ($T_{IDuration}$)	{(1.5, 0), (1, 0), (0.5, 0), (1.5, 3), (1, 3), (0.5, 3)}

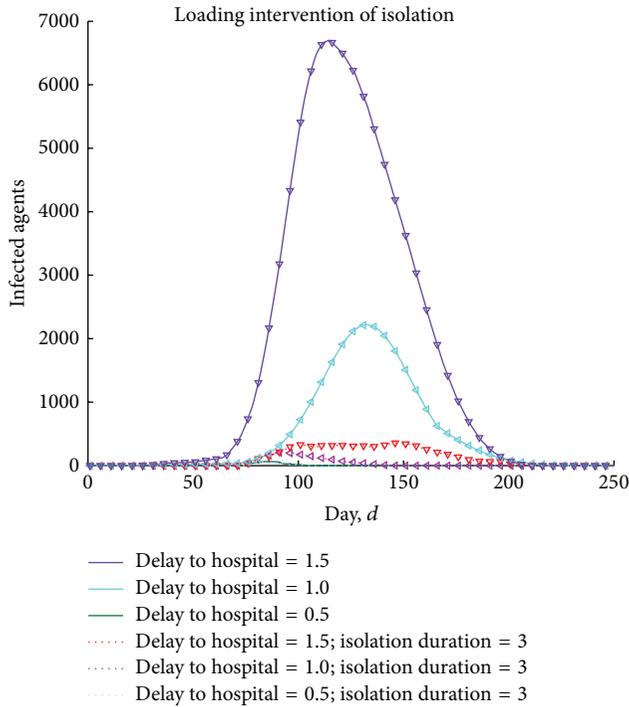


FIGURE 25: The results of loading intervention of isolation.

are sent to hospital, the less the agents will be infected. So the construction of effective mechanism to find infected individuals is important in the emergency management of influenza.

Secondly, the parameter isolation duration is added to simulate the isolation of intervention. Obviously, the isolation measure decreases the infected agents greatly. Compared to the parameter delay to hospital of 1.5-day case, the peak value decreases from 6,708 to 362 when the isolation is added in the intervention. The reduction of 6346 agents comes from the 3-day isolation of agents who had contacted with the infected agents. However, the decrease of parameter delay to hospital cannot bring the similar reduction in the 3-day isolation cases. According to the figure, the peak value decreases from 362 to 51; from 1.5-day case to 0.5-day case, only 311 agents are saved from infection. As a result, the sensitivity of simulated results does not change obviously with delay to hospital in the case of isolation. It can be concluded that isolation is a more effective measure. Though it is not easy for emergency management organizations to decrease the delay to hospital due to the current monitoring mechanism, the influenza can still be controlled by isolation.

5.4.4. *The Experiments of Combination of Intervention Measures.* Isolation is not the only nonmedical intervention in influenza, close workspace is another common measure. When close workspace is activated, the workspaces are all closed, agents can only stay home at the closed time. Luckily, isolation and close workspace are independent from one another. It is possible for emergency management organizations to use these two measures together. So this group of experiments is designed to obtain the effectiveness of the combination of intervention measures. The parameters are initialized with different settings listed in Table 9.

Parameters delay to hospital and isolation duration is set in the middle values (1 day, 3 days) according to the last section. According to the experience, close duration is set as seven days. It is known that close workspace is a really cost intervention measure; the activated time is a key parameter. With the fixed parameters mentioned before, the change of interventions activated time is analyzed in this group of computational experiments. The average results are shown in Figure 26.

Compared with the simulated results of experiments discussed above, Figure 26 strongly illustrates that influenza is ceased when the intervention measures are combined together. The peak value of infected agents is 141 in the “80th days of close-workspace and isolation with 1 day delay to hospital” case while 25,135 in “no intervention” case, 2,218 in “1 day delay to hospital” case, 203 in “1 day delay to hospital and isolation” case. The interventions bring a great reduction of infection; the peak values decrease from ten thousands level to hundred level.

On the other hand, the peak value even decreases to 108 when the execution time of close workspace is 10 days earlier. The date of maximum value of infected agents is advanced from the 135th day to the 108th day. The whole circle of H1N1 influenza is reduced. The results of this group of experiments show that intervention measures should be activated as soon as possible, and the compositions of interventions are more effective in emergency management of H1N1 influenza.

In summary, based on the general process, lots of experiments could be done by KD-ACP to study emergency management in H1N1 influenza. Many conclusions are obtained based on the analysis of the experiments in this section. According to these conclusions, the most optimal response in our case study is the case of “70th days of close-workspace and isolation with 1 day delay to hospital.” It is because this case has the fewest infected agents relatively. But the modelling of H1N1 influenza is not sufficient. Economic and social factors of interventions, such as the money cost by vaccination and isolation, the influence of the close-workspace in society,

TABLE 9: The initial settings of experiments of isolation and close workspace.

Parameters	Value
Agent count	19,610,000
Initial infected count	40
Interventions activated time ($T_{\text{Intervention}}$)	70th day, 80th day
Delay to hospital (T_{HDelay}),	1 day
Isolation duration ($T_{\text{IDuration}}$)	3 days
Close workspace duration ($T_{\text{CWDuration}}$)	7 days

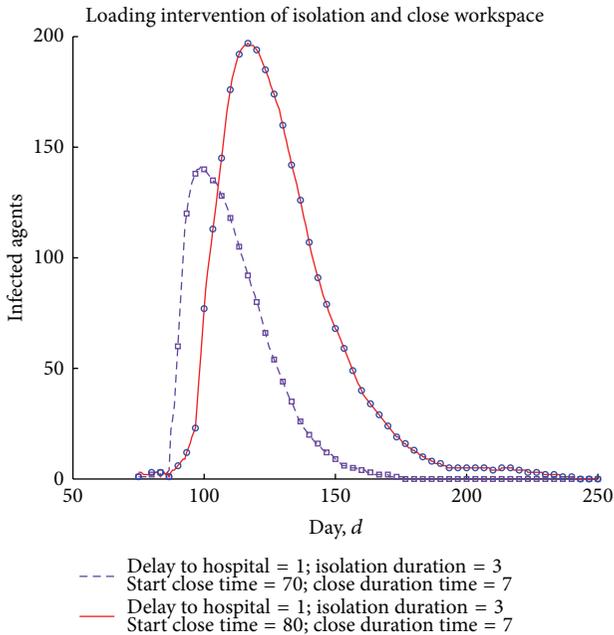


FIGURE 26: The results of loading interventions of isolation and close workspace.

and even opinion transmission of H1N1 in internet, are not considered in our case study. Without the analysis of these factors, it does not make sense to talk about the most optimal intervention measures. However, KD-ACP had provided a software framework and general process of the research on emergency management. Supported by KD-ACP, the experts of emergency management are able to study emergent events such as H1N1 influenza as detailed as possible. Based on the experiments and analysis again and again, the most optimal intervention measures will be found ultimately.

5.5. Parallel Execution by ASST and OsdRT. As discussed in Section 3.5, ASST and OsdRT are used to support the parallel execution of artificial Beijing. The statistical data is collected and stored in artificial society runtime database at runtime. ASST downloads the infection data from the database and displays it in the map. As shown in Figure 27, the infected agents are distributed in the districts in Beijing. The snapshot gives the geospatial distribution of the H1N1 epidemic. The legends of the agents in different status are listed in the left

right corner in Figure 27. Five statuses are shown in the figure: health without vaccination, health with vaccination, incubation, being symptomatic, and convalescence. ASST provides a very direct feeling for the emergency experts. Partly with the situation from ASST, Emergency Decision Organization makes the emergency response plans to the response of the emergency scenario. With the help of plans, the epidemic in actual society can be restrained. In the meantime, the influenza information is reflected in the Internet. According to Figure 8, the data acquisition, data extraction, and data standardization of H1N1 influenza are processed in OsdRT. OsdRT is implemented by Dr. Cao's team from Institute of Automation from Chinese Academy of Sciences [58]. The tool provides not only the reports of H1N1, SARS, hand-foot-mouth disease, and intestinal diseases are all monitored. The web page is shown in Figure 28; H1N1 reports from Internet are collected first. Based on the data extraction of the reports, H1N1 case of distribution is obtained and shown in the incidence map. As a result, H1N1 incidence data is regarded as the knowledge which is sent to EECT and IMCT. The settings of emergent events and intervention measures are modified in order to make artificial society approach actual society in next turn of computational experiments. With the help of OsdRT, the loop of parallel execution process integrates ACP approach.

6. Conclusion

This work provides an integrated software framework for social Computing in emergency management. From a system perspective, KD-ACP provides a reliable, flexible, low cost, and effective platform for the scientists to do the research on emergency response problems. The analysis and prediction of these problems with inherent complexity can be solved by the repetition of possible alternative experiments on the software framework. Therefore, KD-ACP which is an attempt to implement ACP approach seemed as an effective computational framework to support the decision making in emergency response.

Currently, KD-ACP is used to study the H1N1 epidemic in Beijing in 2009. The most optimal compositions of intervention measures are testified to support the response to next influenza. However, it is still a long way ahead before the automation, systematization, and practicalization of KD-ACP. A lot of work will be carried out along several directions as in the following.

- (i) The parallel execution process in KD-ACP is actually not strongly connected with the emergency response organization. Traditional experience and theory still play a dominant role in the decision of emergency management. It is necessary to propose a mechanism of hall for workshop to facilitate the parallel execution loop in actual society.
- (ii) Section 3.3 introduces the principle and process of modelling artificial society. Metamodels and models of artificial Beijing are built by GME. Model transformation and code generation are used to implement codes from FSM based models. But many details

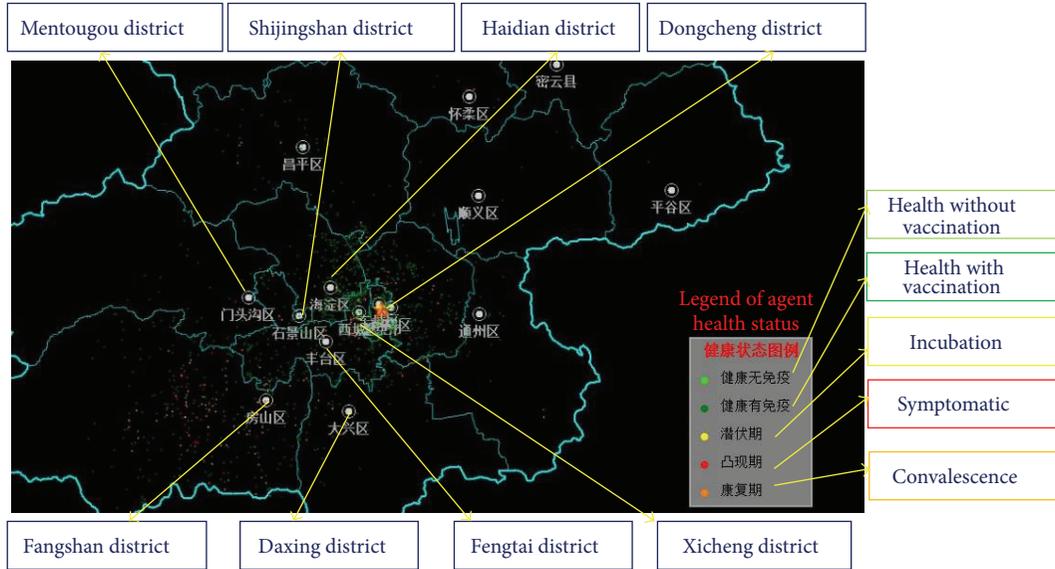


FIGURE 27: The situation of artificial Beijing by ASST.

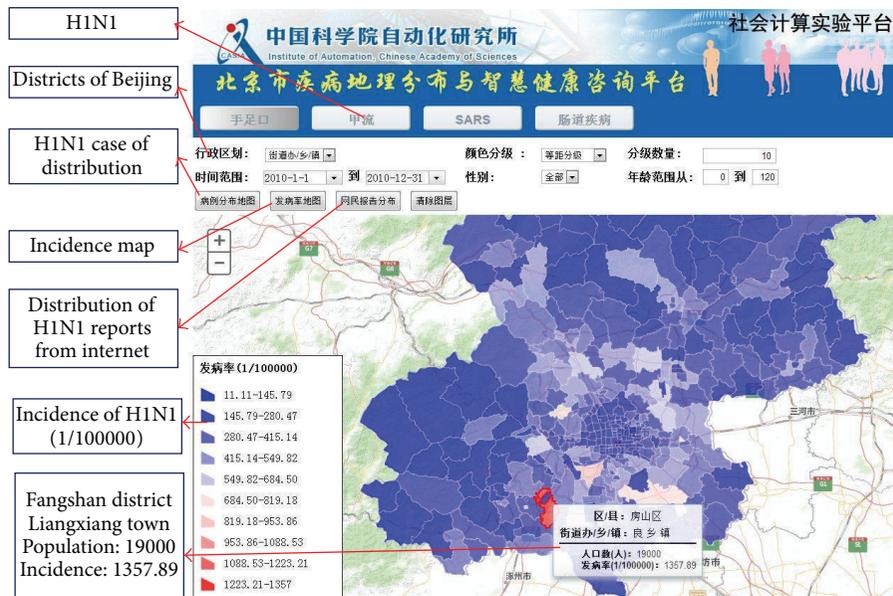


FIGURE 28: The implementation of OsdRT by Institute of Automation from Chinese Academy of Sciences.

are still not illustrated due to the topic of this paper. The key techniques such as template of code framework and model transformations from specific domains to FSM will be focused and introduced in detail in our next paper.

- (iii) The performance of computational experiments in KD-ACP is still slower than expectation. The optimization of simulation engine and models structure is necessary to improve the performance. So it is possible to run artificial society with the real world in parallel.

In summary, KD-ACP paves a new way for scientists in both emergency management and computation simulation to

collaborate with each other in solving the emergency management problems by social computing.

Research Highlights

- (i) KD-ACP is implemented based on ACP approach.
- (ii) Artificial Beijing is built with the help of KD-ACP.
- (iii) H1N1 is modeled to simulate the influenza in Beijing in 2009.
- (iv) Computational experiments testify the effectiveness of intervention measures.

Conflict of Interests

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

Acknowledgment

This work was supported in part by the National Science Foundation of China (nos. 91024030, 71303252, 61403402, and 91324013).

References

- [1] W. Mao and F.-Y. Wang, *Advances in Intelligence and Security Informatics*, Academic Press, Oxford, UK, 2012.
- [2] E. Bonabeau, "Agent-based modeling: methods and techniques for simulating human systems," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 3, pp. 7280–7287, 2002.
- [3] K. M. Carley, D. B. Fridsma, E. Casman et al., "BioWar: scalable agent-based model of bioattacks," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 36, no. 2, pp. 252–265, 2006.
- [4] J. M. Epstein, "Modelling to contain pandemics," *Nature*, vol. 460, no. 7256, p. 687, 2009.
- [5] *Los Alamos National Laboratory Reports about EPISIMS*, Los Alamos National Laboratory, 2003.
- [6] C. H. Builder and S. C. Bankes, "Artificial societies: a concept for basic research on the societal impacts of information technology," RAND Report 149, RAND Corp., 1991.
- [7] F.-Y. Wang, "Computational theory and method on complex system," *China Basic Science*, vol. 6, no. 5, pp. 3–10, 2004.
- [8] F.-Y. Wang, "Toward a paradigm shift in social computing: the ACP approach," *IEEE Intelligent Systems*, vol. 22, no. 5, pp. 65–67, 2007.
- [9] F.-Y. Wang and S. Tang, "Artificial societies for integrated and sustainable development of metropolitan systems," *IEEE Intelligent Systems*, vol. 19, no. 4, pp. 82–87, 2004.
- [10] D. Wen, Y. Yuan, and X.-R. Li, "Artificial societies, computational experiments, and parallel systems: an investigation on a computational theory for complex socioeconomic systems," *IEEE Transactions on Services Computing*, vol. 6, no. 2, pp. 177–185, 2013.
- [11] F. Wang, F. Zeng, and Y. Yuan, "An ACP-based approach for complexity analysis of E-commerce system," *Complex Systems and Complexity Science*, vol. 5, no. 3, pp. 1–8, 2008 (Chinese).
- [12] F.-Y. Wang and P. K. Wong, "Research commentary: intelligent systems and technology for integrative and predictive medicine: an ACP approach," *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 2, pp. 1–6, 2013.
- [13] J. Sifeng, X. Gang, F. Dong, and H. Chunpeng, "Study on the emergency rescue decision support system of petrochemical plant based on ACP theory," in *Proceedings of the 6th Management Annual Meeting (MAM '11)*, 2011, (Chinese).
- [14] X. Dong, D. Fan, G. Xiong, F. Zhu, Z. Zhang, and Y. Yao, "Parallel Bus Rapid Transit (BRT) operation management system based on ACP approach," in *Proceedings of the 6th Management Annual Meeting (MAM '11)*, Beijing, China, 2011.
- [15] B. Ning, F.-Y. Wang, H.-R. Dong, R.-M. Li, D. Wen, and L. Li, "Parallel systems for Urban rail transportation based on ACP approach," *Journal of Transportation Systems Engineering and Information Technology*, vol. 10, no. 6, pp. 23–28, 2010 (Chinese).
- [16] F.-Y. Wang, J. Zhao, and S.-X. Lun, "Artificial power systems for the operation and management of complex power grids," *Southern Power System Technology*, vol. 2, no. 3, pp. 1–11, 2008 (Chinese).
- [17] M. J. North, T. R. Howe, N. T. Collier, and J. R. Vos, "Repast symphony development environment," in *Proceedings of the Agent 2005 Conference on Generative Social Processes, Models and Mechanisms*, 2005.
- [18] M. J. North, T. R. Howe, N. T. Collier, and J. R. Vos, "Repast Symphony runtime system," in *Proceedings of the Agent Conference on Generative Social Processes, Models and Mechanisms*, 2005.
- [19] SWARM Development Group. *Swarm 2.1.1 Reference Guide*, 2000, <http://www.swarm.org>.
- [20] S. Tisue and U. Wilensky, "NetLogo: a simple environment for modeling complexity," in *International Conference on Complex Systems*, pp. 16–21, 2004.
- [21] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "MASON: a multiagent simulation environment," *Simulation*, vol. 81, no. 7, pp. 517–527, 2005.
- [22] M. Ichikawa, H. Tanuma, Y. Koyama, and H. Deguchi, "SOARS: introduction as a social microscope for simulations of social interactions and gaming," in *Organizing and Learning through Gaming and Simulation. Proceedings of the 38th Conference of the International Simulation and Gaming Association (ISAGA '07)*, pp. 149–158, 2007.
- [23] D. S. Burke, J. M. Epstein, D. A. T. Cummings et al., "Individual-based computational modeling of smallpox epidemic control strategies," *Academic Emergency Medicine*, vol. 13, no. 11, pp. 1142–1149, 2006.
- [24] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: concepts, architectures, and applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 630–638, 2010.
- [25] Z. Molnár, D. Balasubramanian, and A. Lédeczi, "An introduction to the generic modeling environment," in *Proceedings of the TOOLS Europe 2007 Workshop on Model-Driven Development Tool Implementers Forum*, Zurich, Switzerland, 2007.
- [26] S. Kelly and J.-P. Tolvanen, *Domain Framework, in Domain-Specific Modeling: Enabling Full Code Generation*, John Wiley & Sons, Hoboken, NJ, USA, 2007.
- [27] B. P. Zeigler, H. Praehofer, and T. G. Kim, *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*, Academic Press, San Diego, Calif, USA, 2000.
- [28] G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modeling Language: User Guide*, Addison-Wesley, Reading, Mass, USA, 1999.
- [29] F. Wagner, R. Schmuki, T. Wagne, and P. Wolstenholme, *Modeling Software with Finite State Machines: A Practical Approach*, CRC & Taylor & Francis, Boca Raton, Fla, USA, 2006.
- [30] T. Espiner, "China builds world's fastest supercomputer," ZDNet UK, October 2010.
- [31] B. Chen and G. Guo, "A two-tier parallel architecture for artificial society simulation," in *Proceedings of the ACM/IEEE/SCS 26th Workshop on Principles of Advanced and Distributed Simulation (PADS '12)*, pp. 184–186, July 2012.
- [32] G. Guo, B. Chen, X. G. Qiu, and Z. Li, "Parallel simulation of large-scale artificial society on CPU/GPU mixed architecture,"

- in *Proceedings of the ACM/IEEE/SCS 26th Workshop on Principles of Advanced and Distributed Simulation (PADS '12)*, pp. 174–177, July 2012.
- [33] S.-X. Lun, “Research on the classification of parallel execution modes of ACP theory,” *Acta Automatica Sinica*, vol. 38, no. 10, pp. 1602–1608, 2012.
- [34] F.-Y. Wang, “A framework for social signal processing and analysis: from social sensing networks to computational dialectical analytics,” *Science China*, vol. 43, no. 12, pp. 1598–1611, 2013.
- [35] G. Guo, *One Model User’s Manual, version 1.2*, National University of Defense University, 2011.
- [36] Y. Ge, R. Meng, Z. Cao, X. Qiu, and K. Huang, “Virtual city—an individual-based digital environment for human mobility and interactive behavior,” *Simulation: Transactions of the Society for the Modeling and Simulation International*, 2014.
- [37] M. Kretzschmar and R. T. Mikolajczyk, “Contact profiles in eight European countries and implications for modelling the spread of airborne infectious diseases,” *PLoS ONE*, vol. 4, no. 6, Article ID e5931, 2009.
- [38] W. Duan and X. Qiu, “Fostering artificial societies using social learning and social control in parallel emergency management systems,” *Frontiers in Computer Science*, vol. 6, no. 5, pp. 604–610, 2012.
- [39] W. J. Edmunds, C. J. O’Callaghan, and D. J. Nokes, “Who mixes with whom? A method to determine the contact patterns of adults that may lead to the spread of airborne infections,” *Proceedings of the Royal Society B: Biological Sciences*, vol. 264, no. 1384, pp. 949–957, 1997.
- [40] J. Banks, J. S. Carson, B. L. Neison, and D. M. Nicol, *Discrete-Event System Simulation*, Prentice Hall, Englewood Cliffs, NJ, USA, 4th edition, 2007.
- [41] H. Yongxia, “Investigation and analysis of college students’ work and rest,” *Value Engineering*, vol. 31, no. 3, 2012 (Chinese).
- [42] W. Duan, Z. Cao, Y. Wang et al., “An ACP approach to public health emergency management: using a campus outbreak of h1n1 influenza as a case study,” *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 43, no. 5, pp. 1028–1041, 2013.
- [43] H. W. Hethcote, “The mathematics of infectious diseases,” *SIAM Review*, vol. 42, no. 4, pp. 599–653, 2000.
- [44] G. Chowell, E. Shim, F. Brauer, and et al., “Modelling the transmission dynamics of acute haemorrhagic conjunctivitis: application to the 2003 outbreak in Mexico,” *Statistics in Medicine*, vol. 25, no. 11, pp. 1840–1857, 2006.
- [45] Y. Ge, L. Liu, B. Chen, X. Qiu, and K. Huang, “Agent-based modeling for Influenza H1N1 in an artificial classroom,” *Systems Engineering Procedia*, vol. 2, pp. 94–104, 2011.
- [46] A. R. Tuite, A. L. Greer, M. Whelan et al., “Estimated epidemiologic parameters and morbidity associated with pandemic H1N1 influenza,” *CMAJ*, vol. 182, no. 2, pp. 131–136, 2010.
- [47] L. Brouwers, B. Cakici, M. Camitz, A. Tegnell, and M. Boman, “Economic consequences to society of pandemic H1N1 influenza 2009—preliminary results for Sweden,” *Euro Surveill*, vol. 14, no. 37, pp. 1–7, 2009.
- [48] Y. Ge, L. Liu, X. Qiu, H. Song, Y. Wang, and K. Huang, “A framework of multilayer social networks for communication behavior with agent-based modeling,” *Simulation: Transactions of the Society for the Modeling and Simulation International*, vol. 89, no. 7, pp. 810–828, 2013.
- [49] W. Duan, X. Qiu, Z. Cao, X. Zheng, and K. Cui, “Heterogeneous and stochastic agent-based models for analyzing infectious diseases’ super spreaders,” *IEEE Intelligent Systems*, vol. 28, no. 4, pp. 18–25, 2013.
- [50] *The Preparation and Emergency Response Plans to Pandemic Published by Ministry of Health*, Ministry of Health, 2005, (Chinese).
- [51] R. M. Fujimoto, *Parallel and Distributed Simulation Systems*, Wiley Interscience, 2000.
- [52] F. Cicirelli, A. Giordano, and L. Nigro, “Distributed simulation of situated multi-agent systems,” in *Proceedings of the 15th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications (DS-RT '11)*, pp. 28–35, September 2011.
- [53] B. Chen and G. Guo, “A two-tier parallel architecture for artificial society simulation,” in *Proceedings of the 26th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation (PADS '12)*, pp. 184–186, July 2012.
- [54] G. Guo, B. Chen, and X. Qiu, “Parallel simulation of large-scale artificial society with GPU as coprocessor,” *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 4, no. 2, Article ID 1350005, 2013.
- [55] B. Chen, L.-B. Zhang, X.-C. Liu, and H. Vangheluwe, “Activity-based simulation using DEVS: increasing performance by an activity model in parallel DEVS simulation,” *Journal of Zhejiang University: Science C*, vol. 15, no. 1, pp. 13–30, 2014.
- [56] X.-L. Wang, P. Yang, Z.-D. Cao et al., “Quantitative evaluation on the effectiveness of prevention and control measures against pandemic influenza A (H1N1) in Beijing, 2009,” *Chinese Journal of Epidemiology*, vol. 31, no. 12, pp. 1374–1378, 2010 (Chinese).
- [57] S. Mei, A. D. van de Vijver, L. Xuan, Y. Zhu, and P. M. A. Sloot, “Quantitatively evaluating interventions in the influenza A (H1N1) epidemic on China campus grounded on individual-based simulations,” *Procedia Computer Science*, vol. 1, no. 1, pp. 1675–1682, 2010.
- [58] X. Qiu, Z. Cao, and Z. Li, *Mid-Term Reports of China National Science Foundation 91024030*, 2014.

Research Article

Weight Optimization in Recurrent Neural Networks with Hybrid Metaheuristic Cuckoo Search Techniques for Data Classification

Nazri Mohd Nawi,¹ Abdullah Khan,¹ M. Z. Rehman,¹
Haruna Chiroma,² and Tutut Herawan³

¹Software and Multimedia Centre, Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400 Batu Pahat, Johor, Malaysia

²Department of Artificial Intelligence, University of Malaya, 50603 Lembah Pantai, Kuala Lumpur, Malaysia

³Department of Information Systems, University of Malaya, 50603 Lembah Pantai, Kuala Lumpur, Malaysia

Correspondence should be addressed to Tutut Herawan; tutut@um.edu.my

Received 16 September 2014; Accepted 15 November 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Nazri Mohd Nawi 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.

Recurrent neural network (RNN) has been widely used as a tool in the data classification. This network can be educated with gradient descent back propagation. However, traditional training algorithms have some drawbacks such as slow speed of convergence being not definite to find the global minimum of the error function since gradient descent may get stuck in local minima. As a solution, nature inspired metaheuristic algorithms provide derivative-free solution to optimize complex problems. This paper proposes a new metaheuristic search algorithm called Cuckoo Search (CS) based on Cuckoo bird's behavior to train Elman recurrent network (ERN) and back propagation Elman recurrent network (BPERN) in achieving fast convergence rate and to avoid local minima problem. The proposed CSERN and CSBPERN algorithms are compared with artificial bee colony using BP algorithm and other hybrid variants algorithms. Specifically, some selected benchmark classification problems are used. The simulation results show that the computational efficiency of ERN and BPERN training process is highly enhanced when coupled with the proposed hybrid method.

1. Introduction

Artificial neural network (ANN) is a well-known procedure that has the ability to classify nonlinear problem and is experimental in nature [1]. However it can give almost accurate solution for clearly or inaccurately formulated problems and for phenomena that are only understood during experiment. An alternate neural network approach is to use recurrent neural networks (RNN), which have inside feedback loops within network allowing them to store previous memory to train past history [2–5]. The RNN model has been implemented in various applications, such as forecasting of financial data [6], electric power demand [7], tracking water quality and minimizing the additives needed for filtering water [7], and data classification [8]. In order to understand the advantage of the dynamical processing of recurrent neural networks, researchers have developed an amount of

schemes by which gradient methods and, in particular, back propagation learning can be extended to recurrent neural networks [8].

Werbos [9] introduced the back propagation through time approach approximating the time evolution of a recurrent neural network as a sequence of static networks using gradient methods. The simple recurrent networks were first trained by Elman with the standard back propagation (BP) learning algorithm, in which errors are calculated and weights are updated at each time step. The BP is not as effective as the back propagation through time (BPTT) learning algorithm, in which error signal is propagated back through time [10]. However, certain properties of the RNN make many of the algorithms less capable, and it often takes huge amount of time to train a network of even a moderate size. In addition, the complex error surface of the RNN network makes many training algorithms more flat to being trapped

in local minima [5]. The back propagation (BP) algorithm is the well-known method for training network. Otherwise, the BP algorithm suffers from two main drawbacks, that is, low convergence rate and instability. They are caused by a possibility of being trapped in a local minimum and view of overshooting the minimum of the error surface [11–17].

To overcome the weaknesses of the above algorithms, there have been many researches on dynamic system modeling with recurrent neural network. This ability of dynamic modeling system formulate a kind of neural network that is more superior to the conventional feed forward neural networks because the system outputs are function of both the current inputs as well as their inner states [18, 19]. Ahmad et al. in [20] investigated a new method using fully connected recurrent neural network (FCRNN) and back propagation through time (BPTT) algorithm to observe the difference of Arabic alphabetic like “*alif*” to “*ya*.” The algorithm is also used to improve the people’s knowledge and understanding of Arabic words using the proposed technique. In 2010, Xiao, Venayagamoorthy, and Corzine trained recurrent neural network integrated with particle swarm optimization (PSO) and BP algorithm (PSOBP) to provide the optimal weights to avoid local minima problem and also to identify the frequency dependent impedance of power electronic system such as rectifiers, inverter, and AC-DC conversion [5]. The experimental results described that the proposed method successfully identified the impedance characteristic of the three-phase inverter system, which not only can systematically help avoiding the training process being trapped in local minima, but also has better performance as compared to both sample BP and PSO algorithms. Similarly, Zhang and Wu [21] used adaptive chaotic particle swarm optimization (ACPSO) algorithm for classification of crops from synthetic aperture radar (SAR) images. During simulations, ACPSO was found to be superior to the back propagation (BP), adaptive BP (ABP), momentum back propagation (MBP), particle swarm optimization (PSO), and resilient back propagation (RPROP) methods [21]. Aziz et al. [22] carried out a study on the performance of particle swarm optimization algorithm with training Elman RNN to discover the classification accuracy and convergence rate compared with Elman recurrent network with BP algorithm. Based on the simulated result it is illustrated that the proposed Elman recurrent network particle swarm optimization (ERNPSO) algorithm is better than the back propagation Elman recurrent network (BPERN) in terms of classification accuracy. However in terms of convergence time the BPERN is much better than the proposed ERNPSO algorithm.

Cheng and Shen [23] proposed an improved Elman RNN to calculate radio propagation loss, with three-dimensional parabola equation method in order to decrease calculation time and to improve approximation performance of the network. Based on results the proposed improved Elman networks show an efficient and feasible performance to predict propagation loss compared with the simple Elman RNN. However, the Elman RNN loses necessary significant data to train the network for predicating propagation. Wang et al. [24] used the Elman RNN to compute the total nitrogen (TN), total phosphorus (TP), and dissolved oxygen (DO)

at three different sites of Taihu during the period of water diversion. The conceptual form of the Elman RNN for different parameters was used by means of the principle component analysis (PCA) and validated on water quality diversion dataset. The values of TS, TP, and DO calculated by the model were intimately related to their respective values. The simulated result shows that the PCA can efficiently accelerate the input parameters for the Elman RNN and can precisely compute and forecast the water quality parameter during the period of water diversion, but not free of local minim problem.

In [25], the proposed LM algorithm based on Elman and Jordan recurrent neural network has been used to forecast annual peak load of Java, Madura, and Bali interconnection for 2009–2011. The study is carried out to check the performance of the proposed LM based recurrent network with respect to their forecasting accuracy over the given period. From the simulation results, it is clear that the proposed LM based recurrent neural network has better performance than the LM based feed forward neural network. After reviewing the above algorithms, it is found that the traditional ANN training has a drawback, that is, slow speed of convergence, which is not definite to find the global minimum of the error function since gradient descent may get stuck in local minima.

To overcome the weaknesses and to improve the convergence rate, this work proposes new hybrid metaheuristic search algorithms that make use of the Cuckoo Search via Levy flight (CS) by Yang and Deb [26] to train Elman recurrent network (ERN) and back propagation Elman recurrent network (BPERN). The main goals of this study are to improve convergence to global minima, decrease the error, and accelerate the learning process using a hybridization method. The proposed algorithms called CSERN and CSBPERN imitate animal behaviour and are valuable for global optimization [27, 28]. The performance of the proposed CSERN and CSBPERN is verified on selected benchmark classification problems and compared with artificial bee colony using BPNN algorithm and other similar hybrid variants.

The remainder of the paper is organized as follows. Section 2 describes the proposed method. Result and discussion are explained in Section 3. Finally, the paper is concluded in Section 4.

2. Proposed Algorithms

In this section, we describe our proposed Cuckoo Search (CS) to train Elman recurrent network (ERN) and back propagation Elman recurrent network (BPERN).

2.1. CSERN Algorithm. In the proposed CSERN algorithm, each best nest represents a possible solution, that is, the weight space and the corresponding biases for ERN optimization. The weight optimization problem and the size of a population represent the quality of the solution. In the first epoch, the weights and biases are initialized with CS and then those weights are passed on to the ERN. The weights

in ERN are calculated. In the next cycle, CS will update the weights with the best possible solution and CS will continue searching the best weights until either the last cycle/epoch of the network is reached or the MSE is achieved. The CS is a population based optimization algorithm; it starts with a random initial population. In the proposed CSERN algorithm, the weight space and the corresponding biases for ERN optimization are calculated by the weight matrices given in (1) and (2) as follows:

$$W_n = U_n = \sum_{n=1}^N a \cdot \left(\text{rand} - \frac{1}{2} \right), \quad (1)$$

$$B_n = \sum_{n=1}^N a \cdot \left(\text{rand} - \frac{1}{2} \right), \quad (2)$$

where $W_n = N$ th weight value in a weight matrix. The rand in (1) is the random number in the range $[0, 1]$, where a is any constant parameter for the proposed method and it is less than 1, and B_n is a bias value. Hence, the list of weights matrix is given as follows:

$$W^c = [W_n^1, W_n^2, W_n^3, \dots, W_n^{N-1}]. \quad (3)$$

Now from neural network process sum of square errors is easily planned for every weight matrix in W^c . For the ERN structure three layers' network with one input layer, one hidden or "state" layer, and one "output" layer are used. Each layer will have its own index variable, that is, k for output nodes, j and l for hidden nodes, and i for input nodes. In a feed forward network, the input vector x is propagated through a weight layer and

$$\text{net}_j(t) = \sum_i^n x_i(t) w_{n(ji)} + B_{n(j)}, \quad (4)$$

where n is the number of inputs and $B_{n(j)}$ is a bias.

In a simple recurrent network, the input vector is similarly propagated through a weight layer but also combined with the previous state activation through an additional recurrent weight layer, U , and

$$y_j(t) = f(\text{net}_j(t)),$$

$$\text{net}_j(t) = \sum_i^n x_i(t) W_{n(ji)} + \sum_l^m y_l(t-1) U_{n(jl)} + B_{n(j)}, \quad (5)$$

$$y_j(t) = f(\text{net}_j(t)),$$

where m is the number of "state" nodes. The output of the network is in both cases determined by the state and a set of output weights W and

$$\text{net}_k(t) = \sum_j^M y_j(t) W_{n(kj)} + B_{n(k)}, \quad (6)$$

$$Y_k(t) = g(\text{net}_k(t)),$$

where g is an output function. Hence, the error can be calculated as follows:

$$E = (T_k - Y_k). \quad (7)$$

The performances index for the network is given as follows:

$$V(x) = \frac{1}{2} \sum_{k=1}^K (T_k - X_k)^T (T_k - Y_k), \quad (8)$$

$$V_F(x) = \frac{1}{2} \sum_{k=1}^K E^T \cdot E.$$

In the proposed method the average sum of squares is the performance index and it is calculated as follows:

$$V_\mu(x) = \frac{\sum_{j=1}^N V_F(x)}{P_i}, \quad (9)$$

where y_r is the output of the network when the k th input net _{i} is presented. The equation $E = (T_k - Y_k)$ is the error for the output layer, $V_\mu(x)$ is the average performance, $V_F(x)$ is the performance index, and P_i is the number of Cuckoo populations in i th iteration. At the end of each epoch the list of average sums of square errors of i th iteration SSE can be calculated as follows:

$$\text{SSE}_i = \{V_\mu^1(x), V_\mu^2(x), V_\mu^3(x), \dots, V_\mu^n(x)\}. \quad (10)$$

The Cuckoo Search is replicating the minimum sum of square error (MSE). The MSE is found when all the inputs are processed for each population of the Cuckoo nest. Hence, the Cuckoo Search nest x_j is calculated as follows:

$$x_j = \text{Min} \{V_\mu^1(x), V_\mu^2(x), V_\mu^3(x), \dots, V_\mu^n(x)\}. \quad (11)$$

The rest of the average sum of squares is considered as other Cuckoo nests. A new solution x_i^{t+1} for Cuckoo i is generated using a Levy flight according to the following equation:

$$x_i^{t+1} = x_i^t + \alpha \oplus \text{levy}(\lambda). \quad (12)$$

Hence, the movement of the other Cuckoo x_i toward x_j can be drawn from (13) as follows:

$$X = \begin{cases} x_i + \text{rand} \cdot (x_j - x_i) & \text{rand}_i > p_\alpha \\ x_i & \text{else.} \end{cases} \quad (13)$$

The Cuckoo Search can move from x_i toward x_j through Levy flight; it can be written as

$$\nabla X_i = \begin{cases} x_i \oplus \alpha \text{levy}(\lambda) \sim 0.01 \cdot \left(\frac{U_j}{|V_j|^{1/\mu}} \right) \cdot (X - X_{\text{best}}) & \text{rand}_i > p_\alpha \\ x_i & \text{else,} \end{cases} \quad (14)$$

where ∇X_i is a small movement of x_i toward x_j . The weights and bias for each layer are then adjusted as follows:

$$\begin{aligned} W_n^{s+1} &= U_n^{s+1} = W_n^s - \nabla X_i, \\ B_n^{s+1} &= B_n^s - \nabla X_i. \end{aligned} \quad (15)$$

The pseudocode for CSERN algorithm is given in Pseudocode 1.

2.2. CSBPERN Algorithm. In the proposed CSBPERN algorithm, each best nest represents a possible solution, that is, the weight space and the corresponding biases for BPERN optimization. The weight optimization problem and the size of the solution represent the quality of the solution. In the first epoch, the best weights and biases are initialized with CS and then those weights are passed on to the BPERN. The weights in BPERN are calculated. In the next cycle CS will update the weights with the best possible solution, and CS will continue searching the best weights until either the last cycle/epoch of the network is reached or the MSE is achieved.

The CS is a population based optimization algorithm, and like other metaheuristic algorithms it starts with a random initial population. In the proposed CSBPERN algorithm, each best nest represents a possible solution, that is, the weight space and the corresponding biases for BPERN optimization. The weight optimization problem and the size of a nest represent the quality of the solution. In the first epoch, the best weights and biases are initialized with CS and then those weights are passed on to the BPERN. The weights in BPERN are calculated. In the next cycle CS will update the weights with the best possible solution and CS will continue searching the best weights until either the last cycle/epoch of the network is reached or the MSE is achieved.

In CSBPERN, the weight value of a matrix is calculated with (1) and (2) as given in Section 2.1. Also, the weight matrix is updated with (3). Now from neural network process sum of square errors (SSE) is easily planned for every weight matrix in W^c . For the BPERN structure three layers' network with one input layer, one hidden or "state" layer, and one "output" layer are used. In CSBPERN network, the input vector x is propagated through a weight layer W using (4). In a simple recurrent network, the input vector is not only similarly propagated through a weight layer, but also combined with the previous state activation through an additional recurrentweight layer U , as given in (5). The output of the network in both cases is determined by the state and a set of output weights W , as given in (6).

According to gradient descent, each weight change in the network should be proportional to the negative gradient of the cost with respect to the specific weights as given in

$$\delta_{pk} = -\eta \frac{\delta E}{\delta y_{pk}}. \quad (16)$$

Thus, the error for output nodes is calculated as follows:

$$\delta_{pk} = (T_{pk} - Y_{pk}) Y_{pk} (1 - Y_{pk}), \quad (17)$$

and for the hidden nodes the error is given as follows:

$$\delta_{pj} = \sum_k^m \delta_{pk} W_{n(kj)} f' (Y_{pj}). \quad (18)$$

Thus the weights and bias are simply changed for the output layer as

$$\begin{aligned} \nabla W^{n+1}_{(kj)} &= \eta \sum_p^n \delta_{pk} y_{pj}, \\ \nabla B^{n+1}_{(kj)} &= \eta \sum_p^n \delta_{pk} y_{pj}, \end{aligned} \quad (19)$$

and for the input layer the weight change is given as

$$\begin{aligned} \nabla W^{n+1}_{(ji)} &= \eta \sum_p^n \delta_{pj} x_{pi}, \\ \nabla B^{n+1}_{(ji)} &= \eta \sum_p^n \delta_{pj} x_{pi}. \end{aligned} \quad (20)$$

Adding a time subscript, the recurrent weights can be modified according to (21) as follows:

$$\nabla U^{n+1}_{(jh)} = \nabla U^n_{(jh)} + \eta \sum_p^n \delta_{pj}(t) y_{ph}(t-1). \quad (21)$$

The network error is calculated for CSBPERN using (7) from Section 2.1. The performance indices for the network are measured with (8) and (9). At the end of each epoch the list of average sums of square errors of i th iteration SSE can be calculated with (10). The Cuckoo Search is imitating the minimum SSE, which is found when all the inputs are processed for each population of the Cuckoo nest. Hence, the Cuckoo Search nest x_j is calculated using (11). A new solution x_i^{t+1} for Cuckoo i is generated using a Levy flight according to (12). The movement of the other Cuckoo x_i toward x_j is controlled through (13). The Cuckoo Search can move from x_i toward x_j through Levy flight as written in (14). The weights and bias for each layer are then adjusted with (15).

The pseudocode for CSBPERN algorithm is given in Pseudocode 2.

3. Result and Discussion

3.1. Datasets. This study focuses on two criteria for the performances analysis: (a) to get less mean square error (MSE) and (b) to achieve high average classification accuracy on testing data from the benchmark problem. The benchmark datasets were used to validate the accuracy of the proposed algorithms taken from UCI Machine Learning Repository. For the experimentation purpose, the data has to be arranged into training and testing datasets; the algorithms are trained on training set, and their performance accuracy is calculated on the corresponding test set. The workstation used for carrying out the experimentation comes equipped with

```

(1) Initializes CS population size dimension and ERN structure
(2) Load the training data
(3) While MSE < stopping criteria
(4) Pass the Cuckoo nests as weights to network
(5) Feed forward network runs using the weights initialized with CS
(6) Calculate the error using (7)
(7) Minimize the error by adjusting network parameter using CS
(8) Generate Cuckoo egg ( $x_j$ ) by taking Levy flight from random nest
 $x_i = x_j$ 
(9) Abandon a fraction  $p_\alpha \in [0, 1]$  of the worst nest. Build new nest at new location via Levy flight to replace the old one
(10) Evaluate the fitness of the nest, Chose a random nest  $i$ 
    If
    (a)  $X_j > X_i$  Then
    (b)  $x_i \leftarrow x_j$ 
    (c)  $X_i \leftarrow X_j$ 
    End if
(11) CS keeps on calculating the best possible weight at each epoch until the network is converged.
End While

```

PSEUDOCODE 1: Pseudocode of CSERN algorithm.

```

(1) Initializes CS population size dimension and BPERN structure
(2) Load the training data
(3) While MSE < stopping criteria
(4) Pass the Cuckoo nests as weights to network
(5) Feed forward network runs using the weights initialized with CS
(6) The sensitivity of one layer is calculated from its previous one and the calculation of the sensitivity start from the last layer of the network and move backward using (17) and (18).
(7) Update weights and bias using (19) to (20)
(8) Calculate the error using (7)
(9) Minimize the error by adjusting network parameter using CS.
(10) Generate Cuckoo egg ( $x_j$ ) by taking Levy flight from random nest.
 $x_i = x_j$ 
(11) Abandon a fraction  $p_\alpha \in [0, 1]$  of the worst nest. Build new nest at new location via Levy flight to replace the old one.
(12) Evaluate the fitness of the nest, Chose a random nest  $i$ 
    If
    (a)  $X_j > X_i$  Then
    (b)  $x_i \leftarrow x_j$ 
    (c)  $X_i \leftarrow X_j$ 
    End if
(13) CS keeps on calculating the best possible weight at each epoch until the network is converged.
End While

```

PSEUDOCODE 2: Pseudocode of CSBPRNN algorithm.

2 GHz processor, 2-GB of RAM, while the operating system used is Microsoft XP (Service Pack 3). Matlab version R2010a software was used to carry out simulation of the proposed algorithms. For performing simulation, seven classification problems, that is, Thyroid Disease [29], Breast Cancer [30], IRIS [31], Glass [32], Australia Credit Card Approval [33], Pima Indian Diabetes [34], and 7-Bit Parity [35, 36] datasets, are selected. The following algorithms are analyzed and simulated on these problems:

(a) Conventional back propagation neural network (BPNN) algorithm.

(b) Artificial bee colony back propagation (ABC-BP) algorithm.

(c) Artificial bee colony neural network (ABCNN) algorithm.

(d) Artificial bee colony Levenberg Marquardt (ABC-LM) algorithm.

(e) Cuckoo Search recurrent Elman network (CSERN) algorithm.

(f) Cuckoo Search back propagation Elman recurrent network (CSBPERN) algorithm.

To compare the performance of proposed algorithms such as CSERN and CSBPERN with conventional BPNN, ABC-BP, and ABC-LM, the network parameters such as number of hidden layers, node in the hidden layer, the value for the weight initialization, and value of learning rate are used similarly. Three layers' NN is used for training and testing of the model. For all problems the NN structure has single hidden layer consisting of five nodes while the input and output layers nodes vary according to the data given. From the input layer to hidden layer and from hidden to output layer log-sigmoid activation function is used as the transform function.

Although the simple Elman neural network (SENN) used the pure line as the activation function for the output layer, learning rate of 0.4 is selected for the entire test. All algorithms were tested using the initial weights and biases are randomly initialized in range [0, 1]; for each problem, one trial is limited to 1000 epochs. A total of 20 trials are run for each dataset to validate these algorithms. For each trial the network results are stored in the result file. Mean square error (MSE), standard deviation of error mean square (SD), the number of epochs, and the average accuracy are recorded in separate file for each trial for selected classification problem.

3.2. Wisconsin Breast Cancer Classification Problem. The Breast Cancer dataset was created by William H. Wolberg. This dataset deals with the breast tumor tissue samples collected from different patients. The cancer analysis are performed to classify the tumor as benign or malignant. This dataset consists of 9 inputs and 2 outputs with 699 instances. The input attributes are, for instance, the clump thickness, the uniformity of cell size, the uniformity of cell shape, the amount of marginal adhesion, the single epithelial cell size, frequency of bare nuclei, bland chromatin, normal nucleoli, and mitoses. The selected network architecture used for the Breast Cancer Classification Problem consists of 9 input nodes, 5 hidden nodes, and 2 output nodes.

Table 1 illustrates that the proposed CSERN and CSBPERN algorithms show better performance than BPNN, ABC-BP, and ABC-LM algorithms. The proposed algorithms achieve small MSE ($3.23E - 05$, 0.00072) and SD ($2.9E - 05$, 0.0004) with 99.95 and 97.37 percent accuracy, respectively. Meanwhile, the other algorithms such as BPNN, ABC-BP, and ABC-LM fall behind the proposed algorithms with large MSE (0.271, 0.014, 0.184, and 0.013) and SD (0.017, 0.0002, 0.459, and 0.001) and lower accuracy. Similarly, Figure 1 shows the performances of MSE convergence for the used algorithms. From the simulation results, it can be easier to understand that the proposed algorithms show better performance than the BPNN, ABC-BP, and ABC-LM algorithms in terms of MSE, SD, and accuracy.

3.3. IRIS Classification Problem. The Iris flower multivariate dataset was introduced by Fisher to demonstrate the discriminant analysis in pattern recognition and machine learning to find a linear feature sets that either merge or separates two or more classes in the classification process. This is maybe the best famous database to be found in the pattern recognition

TABLE 1: Summary of algorithms performance for Wisconsin Breast Cancer Classification Problem.

Algorithms	Accuracy	MSE	SD
ABC-BP	92.02	0.184	0.459
ABC-LM	93.83	0.0139	0.0010
ABCNN	88.96	0.014	0.0002
BPNN	90.71	0.271	0.017
CSERN	99.95	$3.23E - 05$	$2.9E - 05$
CSBPERN	97.37	0.00072	0.0004

TABLE 2: Summary of algorithms performance for Iris Classification Problem.

Algorithms	Accuracy	MSE	SD
ABC-BP	86.87	0.155	0.022
ABC-LM	79.55	0.058	0.0057
ABCNN	80.23	0.048	0.004
BPNN	87.19	0.311	0.022
CSERN	99.97	$5.1E - 06$	$3.2E - 06$
CSBPERN	87.31	0.022	0.006

literature. There were 150 instances, 4 inputs, and 3 outputs in this dataset. The classification of Iris dataset involves the data of petal width, petal length, sepal length, and sepal width into three classes of species, which consist of *Iris setosa*, *Iris versicolor*, and *Iris virginica*. The selected network structure for Iris classification dataset is 4-5-3, which consists of 4 input nodes, 5 hidden nodes, and 3 output nodes. In total 75 instances are used for training dataset and the rest for testing dataset.

Table 2 shows the comparison between performances of the proposed CSERN and CSBPERN algorithms with the BPNN, ABCNN, ABC-BP, and ABC-LM algorithms in terms of MSE, SD, and accuracy. From Table 2 it is clear that the proposed algorithms have better performances by achieving less MSE and SD and higher accuracy than that of the BPNN, ABCNN, ABC-BP, and ABC-LM algorithms. Figure 2 illustrates the MSE convergences performances of the algorithms. From Figure 2, it is clear that the proposed algorithms show higher performances than the other algorithms in terms of MSE, SD, and accuracy.

3.4. Thyroid Classification Problem. This dataset is taken for UCI Learning Repository, created based on the "Thyroid Disease" problem. This dataset consists of 21 inputs, 3 outputs, and 7200 patterns. Each case contains 21 attributes, which can be allocated to any of the three classes, which were hyper-, hypo-, and normal function of thyroid gland, based on the patient query data and patient examination data. The selected network architecture for Thyroid classification dataset is 21-5-3, which consists of 21 input nodes, 5 hidden nodes, and 3 output nodes.

Table 3 summarizes the comparison of performance of the all algorithms in terms of MSE, SD, and accuracy. From the table, it is easy to understand that the proposed CSERN and CSBPERN algorithms have small MSE and SD and

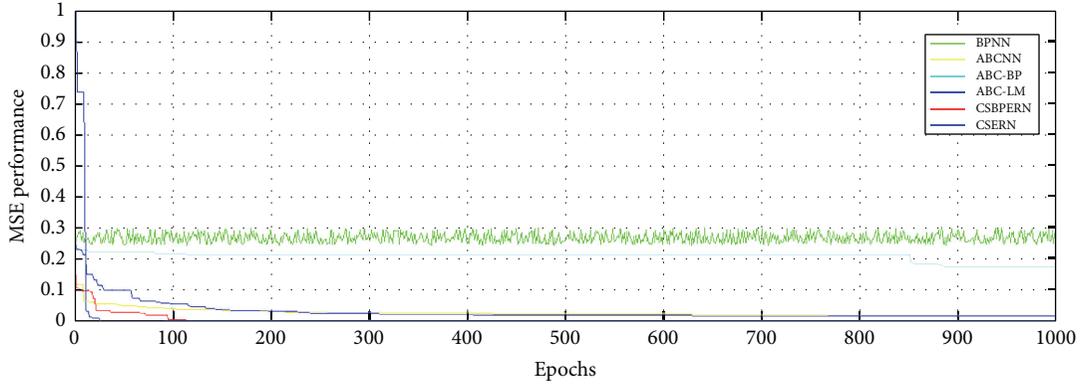


FIGURE 1: MSE via epochs convergence for Wisconsin Breast Cancer Classification Problem.

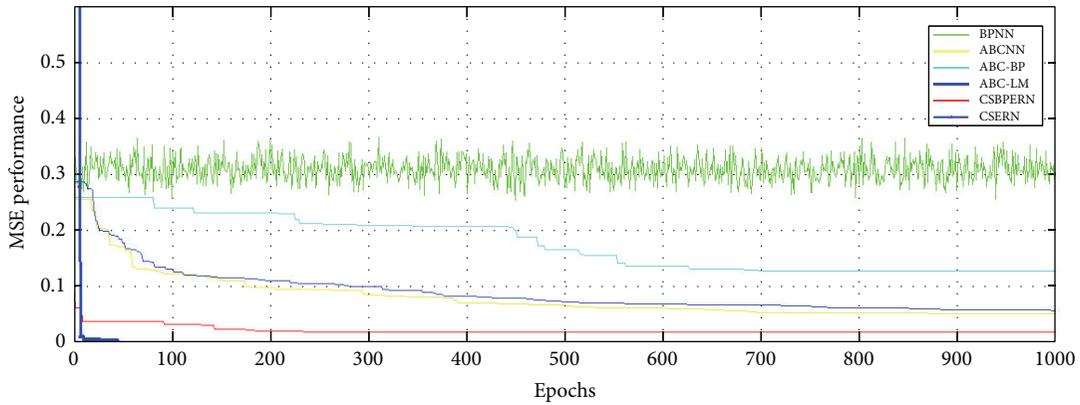


FIGURE 2: MSE via epochs convergence on Iris Benchmark Classification Problem.

TABLE 3: Summary of algorithms performance for Thyroid Classification Problem.

Algorithms	Accuracy	MSE	SD
ABC-BP	93.28	0.046	0.0006
ABC-LM	91.66	0.0409	0.0007
ABCNN	88.18	0.050	0.00064
BPNN	85.88	0.311	0.033
CSERN	99.97	6.6E - 06	2.45E - 06
CSBPERN	95.008	0.0042	0.0086

high accuracy, while the BPNN, BACNN, ABC-BP, and ABC-LM still have large MSE and SD with low accuracy. Figure 3 also shows the MSE convergence performances of the proposed and compared algorithms. From the simulation results, it is realized that the proposed algorithms have better performances in terms of MSE, SD, and accuracy than that of the other compared algorithms.

3.5. Diabetes Classification Problem. This dataset consists of 768 examples, 8 inputs, and 2 outputs and consists of all the information of the chemical change in a female body whose disparity can cause diabetes. The feed forward network topology for this network is set to 8-5-2. The target error

for the Diabetes Classification Problem is set to 0.00001 and the maximum number of epochs is 1000. It is evident from Table 4 that the proposed CSERN and CSBPERN algorithms show better performance than the BPNN, ABCNN, ABC-BP, and ABC-LM algorithms in terms of MSE, SD, and accuracy. From Table 4, it is clear that the proposed algorithms have MSE of $1.7E - 05$, 0.039, and SD of $2.05E - 05$, 0.003, and achieved 99.96, 89.53 percent of accuracy. Meanwhile, the other algorithms such as BPNN, ABCNN, ABC-BP, and ABC-LM have MSE of 0.26, 0.131, 0.2, and 0.14, SD of 0.026, 0.021, 0.002, and 0.033, and accuracy of 86.96, 68.09, 88.16, and 56.09 percent, which is quite lower than the proposed algorithms. Figure 4 describes the MSE convergence performance of the used algorithms for Diabetes Classification Problem.

3.6. Glass Classification Problem. The Glass dataset is used for separating glass splinters in criminal investigation into six classes taken from UCI Repository or Machine Learning database which consists of float processed or non-float processed building windows, vehicle windows, containers, tableware, or head lamp. This dataset is made of 9 inputs and 6 outputs which consist of 214 examples. The selected feed forward network architecture for this network is set to 9-5-6.

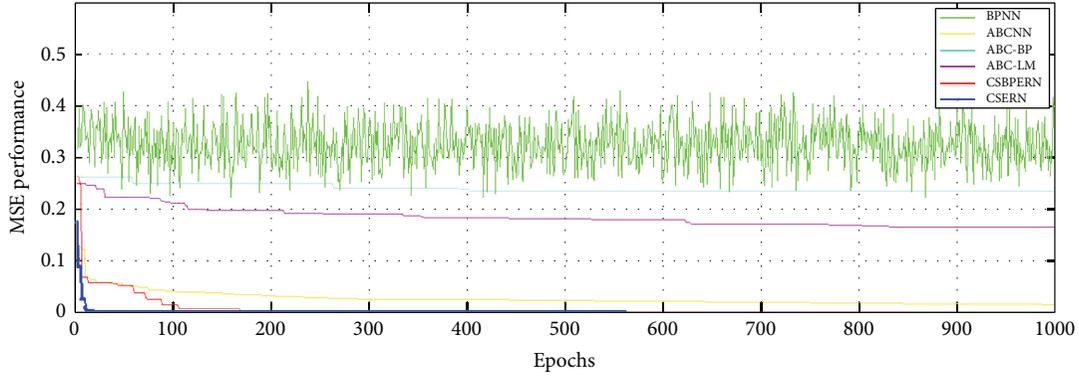


FIGURE 3: MSE via epochs convergence for Thyroid Benchmark Classification Problem.

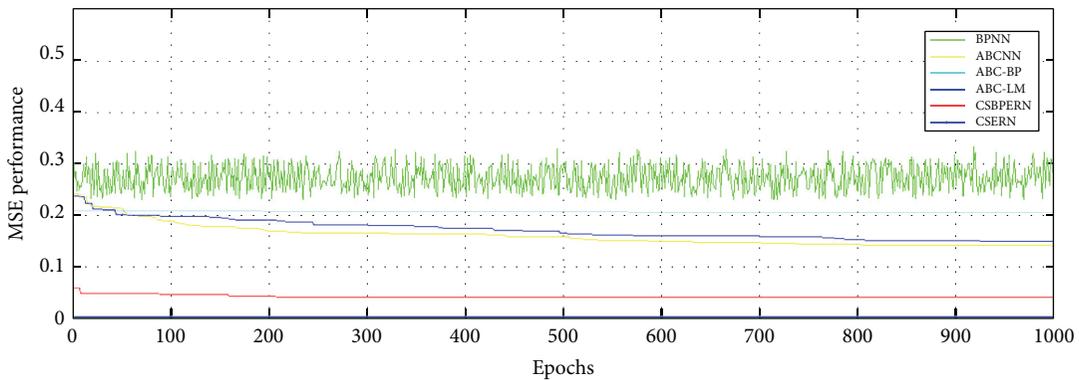


FIGURE 4: MSE via epochs convergence for Diabetes Classification Problem.

TABLE 4: Summary of algorithms performance for Diabetes Classification Problem.

Algorithms	Accuracy	MSE	SD
ABC-BP	88.16	0.20	0.0023
ABC-LM	65.09	0.14	0.033
ABCNN	68.09	0.131	0.021
BPNN	86.96	0.26	0.026
CSERN	99.96	$1.79E - 05$	$2.05E - 05$
CSBPERN	89.53	0.039	0.003

TABLE 5: Summary of algorithms performance for Glass Classification Problem.

Algorithms	Accuracy	MSE	SD
ABC-BP	94.09	0.025	0.009
ABC-LM	93.96	0.005	0.002
ABCNN	91.93	$1.8E - 03$	0.003
BPNN	94.04	0.36	0.048
CSERN	99.96	$2.2E - 05$	$2.5E - 05$
CSBPERN	97.81	0.0005	0.0002

Table 5 summarises the comparison performances of the algorithms. From the table it is clear to understand that the proposed algorithms outperform the other algorithms. The proposed CSERN and CSBPERN algorithms achieve small MSE of $2.20E - 05$, 0.0005 , SD of $2.50E - 05$, 0.0002 , and high accuracy of 99.96 and 97.81 percent. Meanwhile, the BPNN, ABCNN, ABC-BP, and ABC-LM algorithms have large MSE of 0.36, $1.80E - 03$, 0.025, and 0.005, SD of 0.048, 0.003, 0.002, and 0.009, and accuracy of 94.04, 91.93, 94.09, and 93.96 percent, which is quite lower than the proposed algorithms. Figure 5 shows the convergence performance of the algorithms for MSE via epochs. From the overall results, it is clear that the proposed algorithms have better

performances than the other compared algorithms in case of MSE, SD, and accuracy.

3.7. Australian Credit Card Approval Classification Problem.

This dataset is taken from UCI Machine Learning Repository, which contains all the details on the subject of card and application. The Australian Credit Card dataset consists of 690 instances, 51 inputs, and 2 outputs. Each example in this dataset represented a real detail about credit card application, whether the bank or similar institute generated the credit card or not. All attributes names and value have been changed to meaningless symbols to defend the privacy of the data. The selected architecture of NN is 51-5-2.

Table 6 gives the detailed result of the proposed algorithms with the compared algorithms which shows that

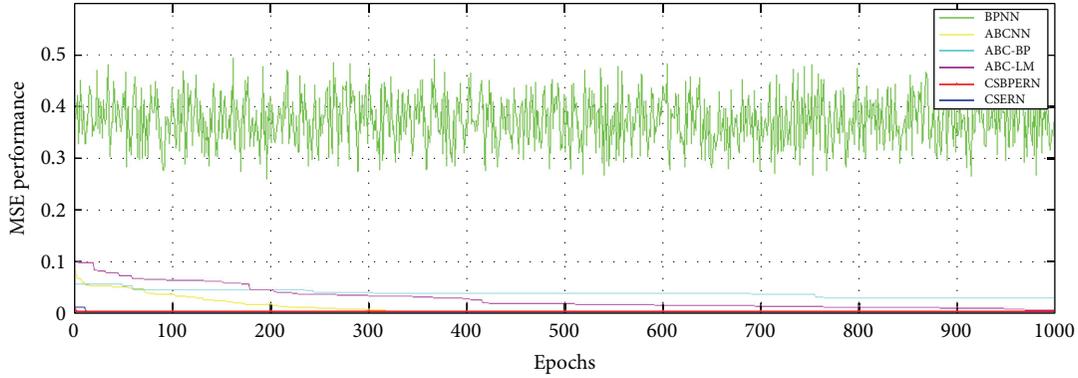


FIGURE 5: MSE via epochs convergence for Glass Benchmark Classification Problem.

TABLE 6: Summary of algorithms performance for Australian Credit Card Approval Classification Problem.

Algorithms	Accuracy	MSE	SD
ABC-BP	89.99	0.17	0.04
ABC-LM	77.78	0.055	0.005
ABCNN	76.79	0.13	0.012
BPNN	88.89	0.271	0.015
CSERN	99.92	$2.15E - 05$	$2.5E - 05$
CSBPERN	85.75	0.021	0.0091

the proposed CSERN and CSBPERN algorithms achieve high accuracy of 99.92, 85.75, with MSE of $2.15E - 05$, 0.021, and SD of $2.5E - 05$, 0.0091. Meanwhile, the other algorithms, that is, BPNN, ABCNN, ABC-BP, and ABC-LM, have accuracy of 88.89, 76.79, 77.78, and 89.99, SD of 0.015, 0.012, 0.005, and 0.04, and MSE of 0.271, 0.13, 0.055, and 0.17, which is quite larger than the proposed algorithms. Similarly, Figure 6 shows the MSE convergence performances of the algorithms for the Australian Credit Card Approval Classification Problem. From these figures it can be easy to understand that the proposed algorithms have better result than that of the other compared algorithms.

3.8. Seven-Bit Parity Classification Problem. The parity problem is one of the most popular initial testing tasks and is a very demanding classification problem for neural network. In parity problem if given input vectors contain an odd number of one, the corresponding target value is 1; otherwise the target value is 0. The N -bit parity training set consists of $2N$ training pairs, with each training pair comprising an N -length input vector and a single binary target value. The $2N$ input vector represents all possible combinations of the N binary numbers. The selected architecture of NN is 7-5-1. Table 7 gives the detailed summary of the algorithms in terms of MSE, SD, and accuracy. From the table, it is clear that the proposed CSERN and CSBPERN algorithms have better performance than BPNN, ABCNN, ABC-BP, and ABC-LM algorithms in terms of MSE, SD, and accuracy. The proposed algorithms have MSE of $2.31E - 06$, 0.052, and SD of $2.6E - 06$, 0.005, and achieve 99.98 and 89.28 percent of accuracy.

TABLE 7: Summary of algorithms performance for 7-Bit Parity Classification Problem.

Algorithms	Accuracy	MSE	SD
ABC-BP	82.12	0.12	0.008
ABC-LM	69.13	0.08	0.012
ABCNN	67.85	0.10	0.015
BPNN	85.12	0.26	0.014
CSERN	99.98	$2.31E - 06$	$2.6E - 06$
CSBPERN	89.28	0.052	0.005

Meanwhile, the other BPNN, ABCNN, ABC-BP, and ABC-LM algorithms converge with MSE of 0.26, 0.10, 0.12, and 0.08, SD of 0.014, 0.015, 0.008, and 0.012, and 85.12, 67.85, 82.12, and 69.13 percent of accuracy, which is quite lower than that of the proposed algorithms. Finally, Figure 7 represents the MSE convergence performance of the algorithms for the 7-Bit Parity Classification Problem.

4. Conclusion

This paper has studied the data classification problem using the dynamic behavior of RNN trained by nature inspired metaheuristic Cuckoo Search algorithm which provides derivative-free solution to optimize complex problems. This paper has also proposed a new metaheuristic Cuckoo Search based on ERN and BPERN algorithms in order to achieve fast convergence rate and to avoid local minima problem in conventional RNN. The proposed algorithms called CSERN and CSBPERN are unlike the existing algorithms; CSERN and CSBPERN imitate animal behaviour and are valuable for global convergence. The convergence behaviour and performance of the proposed CSERN and CSBPERN are simulated on some selected benchmark classification problems. Specifically, 7-Bit Parity and some selected UCI benchmark classification datasets are used for training and testing the network. The performances of the proposed models are compared with artificial bee colony using BPNN algorithm and other hybrid variants. The simulation results show that the proposed CSERN and BPERN algorithms are far better than the baseline algorithms in terms of convergence rate.

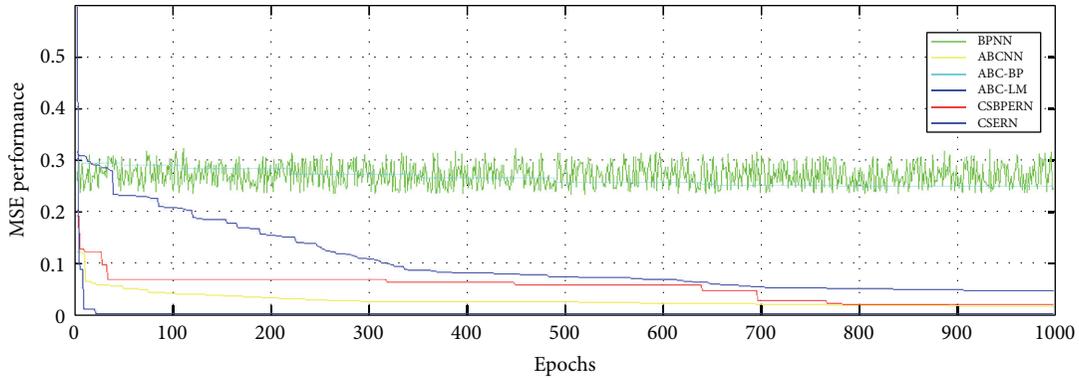


FIGURE 6: MSE via epochs convergence for Credit Card Benchmark Classification Problem.

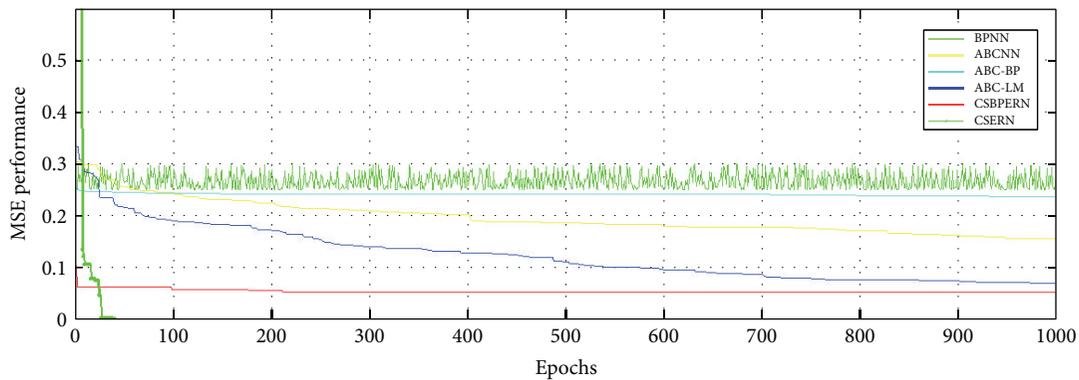


FIGURE 7: MSE via epochs convergence for 7-Bit Parity Classification Problem.

Furthermore, CSERN and BPERN achieved higher accuracy and less MSE on all the designated datasets.

Summary of Acronyms, Mathematical Symbols, and Their Meanings Used

ABP:	Adaptive back propagation
ANN:	Artificial neural network
ACPSO:	Adaptive chaotic particle swarm optimization
BP:	Back propagation
BPERN:	Back propagation Elman recurrent network
BPTT:	Back propagation through time
CS:	Cuckoo Search
CSERN:	Cuckoo Search Elman recurrent network
CSBPERN:	Cuckoo Search back propagation Elman recurrent network
DO:	Dissolved oxygen
ERN:	Elman recurrent network
ERNPSO:	Elman recurrent network particle swarm optimization
FCRNN:	Fully connected recurrent neural network
MBP:	Momentum back propagation
PCA:	Principle component analysis
PSO:	Particle swarm optimization

PSO-BP:	Particle swarm optimization back propagation
RPROP:	Resilient back propagation
RNN:	Recurrent neural network
SAR:	Synthetic aperture radar
SSE:	Sum of square errors
TN:	Total nitrogen
TP:	Total phosphorus
W_n :	Weight value at each layer in the feed forward network
U_n :	Weight value at each addition layer in the recurrent feedback
B_n :	Bias values for the network
W^c :	Total weight matrix for the network
net_j :	Output function for the hidden layer
net_k :	Output function for the output layer
f :	Net activation function for the hidden layer
g :	Net input activation function for the output layer
T_k :	Actual output
Y_k :	Predicted output
$V_\mu(x)$:	Average performance
$V_F(x)$:	Performance index
rand:	Random function for generating random variables
E :	Error at the output layer

x_i : Cuckoo nest at position i
 x_j : Cuckoo nest at position j
 x_i^{i+1} : New solution for Cuckoo
 \oplus : Stepwise multiplication
 ∇X_i : Small movement of Cuckoo x_i towards x_j
 δ_{pk} : Error for output nodes
 δ_{pk} : Error for hidden nodes
 ∇W : Change in weights for the layers
 ∇B : Change in bias weights for the layers
 ∇U : Change in weights for the recurrent layer.

Conflict of Interests

There is no conflict of interests reported regarding the publication of this paper.

Authors' Contribution

All authors equally contributed to this paper.

Acknowledgments

This work is supported by Fundamental Research Grant Scheme (FRGS) Vote no. 1236 from MoHE Malaysia and Program Rakan Penyelidikan University of Malaya (PRPUM) Grant Vote no. CG063-2013.

References

- [1] Y. Zhang and L. Wu, "Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network," *Expert Systems with Applications*, vol. 36, no. 5, pp. 8849–8854, 2009.
- [2] L. Fauseeti, *Fundamental of Neural Network Architecture, Algorithm and Application*, Prentice Hall, Englewood Cliffs, NJ, USA, 1994.
- [3] S. Haykin, *Neural Network: A Comprehensive Foundation*, Macmillan, New York, NY, USA, 1994.
- [4] M. H. Hassoun, *Fundamental of Artificial Neural Network*, Massachusetts Institute of Technology Press, London, UK, 1995.
- [5] P. Xiao, G. K. Venayagamoorthy, and K. A. Corzine, "Combined training of recurrent neural networks with particle swarm optimization and backpropagation algorithms for impedance identification," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS '07)*, pp. 9–15, April 2007.
- [6] C. L. Giles, S. Lawrence, and A. C. Tsoi, "Rule inference for financial prediction using recurrent neural networks," in *Proceedings of the IEEE/IAFE Conference on Computational Intelligence for Financial Engineering*, pp. 253–259, IEEE, March 1997.
- [7] S. Li, D. C. Wunsch, E. O'Hair, and M. G. Giesselmann, "Wind turbine power estimation by neural networks with Kalman filter training on a SIMD parallel machine," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '99)*, pp. 3430–3434, Washington, DC, USA, July 1999.
- [8] N. M. Nawawi, A. Khan, and M. Z. Rehman, "CSBPRNN: a new hybridization technique using cuckoo search to train back propagation recurrent neural network," in *Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013)*, vol. 285 of *Lecture Notes in Electrical Engineering*, pp. 111–118, 2014.
- [9] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.
- [10] P. Coulibaly, F. Anctil, and J. Rousselle, "Real-time short-term natural water inflows forecasting using recurrent neural networks," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '99)*, vol. 6, pp. 3802–3805, IEEE Press, July 1999.
- [11] N. M. Nawawi, R. S. Ransing, and N. A. Hamid, "BPGD-AG: a new improvement of back-propagation neural network learning algorithms with adaptive gain," *Journal of Science and Technology*, vol. 2, pp. 83–102, 2011.
- [12] A. Khan, N. M. Nawawi, and M. Z. Rehman, "A new back-propagation neural network optimized with cuckoo search algorithm," in *Computational Science and Its Applications—ICCSA 2013: Proceedings of the 13th International Conference, Ho Chi Minh City, Vietnam, June 24–27, 2013, Part I*, vol. 7971 of *Lecture Notes in Computer Science*, pp. 413–426, Springer, Berlin, Germany, 2013.
- [13] A. Khan, N. M. Nawawi, and M. Z. Rehman, "A new cuckoo search based Levenberg-Marquardt (CSLM) algorithm," in *Computational Science and Its Applications—ICCSA 2013: Proceedings of the 13th International Conference, Ho Chi Minh City, Vietnam, June 24–27, 2013, Part I*, vol. 7971 of *Lecture Notes in Computer Science*, pp. 438–451, Springer, Berlin, Germany, 2013.
- [14] N. M. Nawawi, A. Khan, and M. Z. Rehman, "A new Levenberg Marquardt based back propagation algorithm trained with cuckoo search," *Procedia Technology*, vol. 11, pp. 18–23, 2013.
- [15] N. M. Nawawi, M. Z. Rehman, and A. Khan, "A new bat based back-propagation (BAT-BP) algorithm," in *Advances in Systems Science: Proceedings of the International Conference on Systems Science 2013 (ICSS 2013)*, vol. 240 of *Advances in Intelligent Systems and Computing*, pp. 395–404, Springer, 2014.
- [16] N. M. Nawawi, R. Ghazali, and M. N. M. Salleh, "The development of improved back-propagation neural networks algorithm for predicting patients with heart disease," in *Information Computing and Applications: Proceedings of the 1st International Conference, ICICA 2010, Tangshan, China, October 15–18, 2010*, vol. 6377 of *Lecture Notes in Computer Science*, pp. 317–324, Springer, Berlin, Germany, 2010.
- [17] Y. Zhang, S. Wang, G. Ji, and P. Phillips, "Fruit classification using computer vision and feedforward neural network," *Journal of Food Engineering*, vol. 143, pp. 167–177, 2014.
- [18] T. G. Barbounis, J. B. Theocharis, M. C. Alexiadis, and P. S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models," *IEEE Transactions on Energy Conversion*, vol. 21, no. 1, pp. 273–284, 2006.
- [19] E. D. Übeyli, "Recurrent neural networks employing Lyapunov exponents for analysis of doppler ultrasound signals," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2538–2544, 2008.
- [20] A. M. Ahmad, S. Ismail, and D. F. Samaon, "Recurrent neural network with backpropagation through time for speech recognition," in *Proceedings of the IEEE International Symposium on Communications and Information Technologies: Smart Info-Media Systems (ISCIT '04)*, pp. 98–102, October 2004.
- [21] Y. Zhang and L. Wu, "Crop classification by forward neural network with adaptive chaotic particle swarm optimization," *Sensors*, vol. 11, no. 5, pp. 4721–4743, 2011.
- [22] M. F. A. Aziz, H. N. A. Hamed, and S. M. H. Shamsuddin, "Augmentation of Elman Recurrent Network learning with

- particle swarm optimization,” in *Proceedings of the 2nd Asia International Conference on Modelling & Simulation (AMS '08)*, pp. 625–630, May 2008.
- [23] F. Cheng and H. Shen, “An improved recurrent neural network for radio propagation loss prediction,” in *Proceedings of the International Conference on Intelligent Computation Technology and Automation (ICICTA '10)*, pp. 579–582, May 2010.
- [24] H. Wang, Y. Gao, Z. Xu, and W. Xu, “An recurrent neural network application to forecasting the quality of water diversion in the water source of Lake Taihu,” in *Proceedings of the International Conference on Remote Sensing, Environment and Transportation Engineering (RSETE '11)*, pp. 984–988, June 2011.
- [25] Y. Tanoto, W. Ongsakul, and C. O. P. Marpaung, “Levenberg-Marquardt recurrent networks for long-term electricity peak load forecasting,” *Telkommika (Telecommunication, Computing, Electronics and Control)*, vol. 9, pp. 257–266, 2013.
- [26] 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.
- [27] X.-S. Yang and S. Deb, “Engineering optimisation by cuckoo search,” *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 330–343, 2010.
- [28] M. Tuba, M. Subotic, and N. Stanarevic, “Modified cuckoo search algorithm for unconstrained optimization problems,” in *Proceedings of the 5th European Computing Conference (ECC '11)*, pp. 263–268, April 2011.
- [29] J. R. Quinlan, P. J. Compton, K. A. Horn, and L. Lazarus, “Inductive knowledge acquisition: a case study,” in *Proceedings of the 2nd Australian Conference on Applications of Expert Systems*, pp. 137–156, 1986.
- [30] W. H. Wolberg and O. L. Mangasarian, “Multisurface method of pattern separation for medical diagnosis applied to breast cytology,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 87, no. 23, pp. 9193–9196, 1990.
- [31] R. A. Fisher, “The use of multiple measurements in taxonomic problems,” *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.
- [32] I. W. Evett and E. J. Spiehler, “Rule induction in forensic science,” in *Knowledge Based Systems in Government*, pp. 107–118, 1987.
- [33] J. R. Quinlan, “Simplifying decision trees,” *International Journal of Man-Machine Studies*, vol. 27, no. 3, pp. 221–234, 1987.
- [34] R. A. Jacobs, “Increased rates of convergence through learning rate adaptation,” *Neural Networks*, vol. 1, no. 4, pp. 295–307, 1988.
- [35] 7-Bit Parity Training data, <http://homepages.cae.wisc.edu/~ece539/data/parity7r>.
- [36] 7-Bit Parity Testing data, <http://homepages.cae.wisc.edu/~ece539/data/parity7t>.

Research Article

Model of Multilayer Knowledge Diffusion for Competence Development in an Organization

Przemysław Różewski and Jarosław Jankowski

Faculty of Computer Science and Information Technology, West Pomeranian University of Technology in Szczecin, Żołnierska 49, 71-210 Szczecin, Poland

Correspondence should be addressed to Przemysław Różewski; prozewski@wi.zut.edu.pl

Received 16 October 2014; Accepted 15 December 2014

Academic Editor: Yudong Zhang

Copyright © 2015 P. Różewski and J. Jankowski. 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.

Growing role of intellectual capital within organizations is affecting new strategies related to knowledge management and competence development. Among different aspects related to this field, knowledge diffusion has become one of the interesting areas from both practitioner and researcher's perspectives. Several models were proposed with main goal of simulating diffusion and explaining the nature of these processes. Existing models are focused on knowledge diffusion and they assume diffusion within a single layer using knowledge representation. From the organizational perspective connecting several types of knowledge and modelling changes of competence can bring additional value. In this paper we extended existing approaches by using multilayer diffusion model and focused on analysis of competence development process. The proposed model describes competence development process in a new way through horizontal and vertical knowledge diffusion in multilayer network. In the network, agents collaborate and interchange various kinds of knowledge through different layers and these mutual activities affect the competencies in a positive or negative way. Taking into consideration worker's cognitive and social abilities and the previous level of competence the new competence level can be estimated. The model is developed to support competence management in different organizations.

1. Introduction

Employees' competence becomes the main part of organization's intellectual capital [1]. According to [2] the management and control of knowledge and skills, and more recently the management of organizations' competencies, have turned out to be essential factors of industrial processes' performance. Modern companies are no longer production systems of products and services but create and sell knowledge-based products. Including competence management into production process required integrating new decision processes regarding the cognitive dimension of business, at every managerial level [2]. Moreover, the companies have to expand knowledge management into competence management. As a result the companies will be able to fulfil the following items [3]: find the right single employee for a specific task or project, retrieve and assemble flexible project teams, develop and update employees' skills, explore the employees' future career paths, and speed up innovation management. The workers

become knowledge workers [4] and continuing needs for upgrading workplace knowledge, skills and competencies are developed. Changes in work and the ways in which it is carried out bring out the need for upgrading workplace knowledge, skills, and competencies. In today's workplaces, and for a number of reasons, workloads are higher than ever and stress is a growing concern [5].

Competence is an observable or measurable ability of an actor to perform a necessary action(s) in a given context(s) to achieve a specific outcome(s) [6]. After analysis of various competence definitions [7, 8] one thing is common: competence is made of different knowledge-based components (e.g., knowledge, skills, and behaviours). Competence development process is an acquisition of a specific set of competence's components that constitutes a particular competence [9].

In our work the modelling of the competence development process is based on the knowledge diffusion model that extends current solutions. The approach is new and required special characteristics of diffusion model. We developed

a multilayer diffusion model based on the multilayer graph reflecting organisation's network. In the graph each layer represents competence's component (some kind of knowledge). There is an interaction between layers defined as a vertical diffusion. The horizontal diffusion occurs on every layer's level and relates to the diffusion of one type of knowledge between knowledge workers. Moreover, every node of organisation's network represents knowledge worker with individual set of knowledge and own cognitive and social potentials for learning (self-learning) and teaching (training). The knowledge worker, in every step of simulation, is looking for best source of knowledge. In addition, depending on node's neighbourhood, the knowledge can be forgotten.

The existing diffusion models from the literature were not suitable for competence modelling due to their limitations. The most important drawback is the lack of simultaneous support of vertical and horizontal diffusion. Moreover, diffusion logic proposed in the literature does not reflect the competence development process. In our approach the diffusion logic is set to search for best teacher (source of knowledge) in node's neighbourhood. The best teacher is a node with the highest value of knowledge and teaching ability. The diffusion result is affected by the learning/teaching abilities of nodes, initial value of knowledge, vertical diffusion form other layers (relation between knowledge), and forgetting process. Similarly constructed diffusion model cannot be found in the literature.

The rest of the paper is organized in the following way. Section 2 covers the issue of competence development in organization. We listed components associated with competence and developed a way to include them in the diffusion model. Section 3 analyses the knowledge diffusion models. We concluded that all of them are lacking some proposition for competence development. Section 4 is dedicated to the description of multilayer knowledge diffusion model for competence development in an organization. The model is the focal point of the paper. Section 5 is dedicated to validation of proposed model. In this section some case studies will be presented.

2. Competence Development in an Organization

2.1. Competence-Based Approach to System Design. In the literature we can find different definitions of competence linked with three fundamental characteristics: resources, context, and objectives [10]. The competence profile is data about a competence that may be aggregated for communication among individuals, organizations, and public administrations. The competence modelling issue has been a subject of research for a long time, starting with Taylor [11]. However, in recent years, studies have greatly accelerated. An interesting review of competence notion can be found in [8, 12, 13]. The history and background of standardization in this area and research project are covered in [7]. Some computational approaches for competence profile processing are described in [14, 15]. The fuzzy nature of competence description is explained in [16]. Moreover, there is a number of high quality scientific journals with special issue dedicated only to

competence including "Competence Management in Industrial Processes" [17], "Skills Management—Managing Competencies in the Knowledge-Based Economy" [18], "Learning Networks for Lifelong Competence Development" [19], "Assessment of Competencies" [20], and "Competencies Management" [21].

Competence-based approaches have proved to be a critical tool in many organizational functions, such as employment planning, recruitment, trainings, raising work efficiency, personal development, and managing key competencies [22]. In addition, competence-based system can be used for different purposes such as staff development and deployment, job analysis, or economic evaluation [23]. The reasons for this are made by [24] as follows.

- (i) Competence-based approach can provide identification of the skills, knowledge, behaviours, and capabilities needed to meet current and future personnel selection needs, in alignment with the differentiations in strategies and organizational priorities.
- (ii) Competence-based approach can focus on the individual and group development plans to eliminate the gap between the competencies requested by a project, job role, or enterprise strategy and those available.

The important way of competence developing is a community of practice because a growing number of people and organizations in various sectors are now focusing on communities of practice as a key to improving their performance [10]. Communities of practice are groups of people who share a concern or a passion for something they do and learn how to do it better as they interact regularly [25].

From a pragmatic point of view competence is a combination of components, usually related to knowledge, experiences, and skills/abilities. It is important to notice that it is not possible to directly develop another person's competence. It is just possible to set the scene, to provide the tools, and to act like a catalyst [26]. As a result, the competence development is regarded as the acquisition of a specific set of competence's components (e.g., knowledge, skills) that constitutes a particular competence [9]. Moreover, overriding principle for development of competence becomes transmitting such attributes (components) to those people who do not possess them by a range of activities, such as general communication, classroom teaching, on-the-job training, and job rotation [9]. The data about the competencies value/state is produced and transformed by identification, assessment, and acquisition processes [27]. Competencies can be processed because there is a certain set of tools used to test competencies and estimate their levels [22, 28].

There are some challenging issues with competence-based approach [29]: development and use of a consistent set of concepts and vocabulary for describing competencies; classification of the different kinds and levels of activities within organizations that collectively contribute to achieving competence; and articulating the interactions of different kinds and levels of organizational activities that are critical in processes of competence building and leveraging.

In order to overcome this challenging issue the following attributes had to be identified and analysed [30]: how roles

TABLE 1: List of main competence components.

Competence components (categories of resources)	Base references
(i) Knowledge (ii) Skills (iii) Abilities (iv) Behaviours	Treasury Board of Canada Secretariat (http://www.tbs-sct.gc.ca/)
(i) Knowledge (ii) Skill (iii) Ability (iv) Other deployment-related characteristics (e.g., attitude, behaviour, and physical ability)	HR-XML (http://www.hr-xml.org/)
(i) Input competencies (a) Knowledge (b) Skills (ii) Personal competencies (a) Core personality characteristics (iii) Output competencies (a) Demonstrable performance	[101]
(i) Know-what (ii) Know-how (iii) Know-who (how closely one is acquainted with someone, knowledge from social network)	[102]
(i) Knowledge (ii) Know-how (iii) Behaviours	[103]
(i) Knowledge (what you learn in education) (ii) Experience (what you gather in your job, at your workplace, and in social life) (iii) Abilities (to use your knowledge and experience)	[26]
(i) Knowledge (theoretical knowledge, knowledge of the existing, and knowledge of procedures) (ii) Know-how (procedural know-how, empirical know-how) (iii) Know-whom (relationship aspects, cognitive capacities, and behaviours)	[103]
(i) Knowledge (includes theoretical knowledge and procedural knowledge) (ii) Skills (includes formalized know-how and empirical know-how) (iii) Behavioural aptitudes (individual's behaviour at work)	[23]

are assigned to employees; guiding principles; defined organizational processes; organizational culture (including values, atmosphere, and practices); organizational knowledge; managerial practices; organizational learning; information and information technology systems; and work environment.

From observation of the organization we can observe that in any organization the competencies are considered at the following levels [2]: individual competence: competence of a person; collective/team competence: competence emerging from a group of persons; and global/organizational competence: describing organizational ability of an enterprise.

The literature proposed some content of particular levels [30, 31]:

- (i) individual competence (e.g., result orientation, role commitment, continuous learning, networking, creativity, intelligence, behavioral traits (including such aspects as honesty and maturity), motivation, and communication capabilities);
- (ii) team competence (e.g., knowledge sharing, cultural integration, resources utilization, innovation, and management/leadership);

- (iii) organizational competence (e.g., knowledge landscape, knowledge assets, information sharing, push/pull power balance, and synergy creation).

The dichotomy between definitions of competence that target individual workers and definitions that target the results of their work is a complex issue [32]. On the one hand the literature has focused on individual competencies and has taken the worker's attributes as a starting point for discussing competence [33]. The worker's competence value is treated as a stock that can be developed through training and validated in "objective" rating schedules [34]. On the other hand, the competence is conceptualised as a characteristic of organisations where human competencies are seen as one of the resources available to organisations [33].

2.2. Competence as a Union of Associated Components. For the propose of model designing it is required to identify the structure of competence. Competencies are considered as a union of different components (see Table 1). Thanks to the literature analysis (e.g., [7, 29, 35]) we can distinguish some base components like knowledge, skills, experience, and so forth. However, the competencies are not themselves

resources in the sense of knowing how to act, knowing how to do, or attitudes [36]. The goal of competencies is to mobilize, integrate, and orchestrate such resources. It is important to notice that all components presented in Table 1 can be somehow measured.

After the presentation of the components of competence it is crucial to understand what the relationship between them is. A good way of understanding the relationship is use of competence ontology structures, which can be found in the literature [14, 24, 37–39]. Most of them are designed to [38]

- (i) define an organization-wide role structure based on the competencies required by job functions and organizational positions;
- (ii) identify the competencies required in order to perform the various activities involved in each business process and assignment of roles to process activities based on these competencies;
- (iii) identify the competencies acquired in the organization and assignment of users to roles through competence matching.

Moreover, the competence ontology is the most important part of an effective competence management system [24]. The competence's ontology is important because competence management system has to collaborate with other similar systems or e-learning and human resources applications. More formal approach to competence ontology building, based on the Description Logics, can be found in [39].

3. Models of Knowledge Diffusion

3.1. The Nature of the Problem. Diffusion of knowledge can be analysed in several dimensions. Knowledge diffusion, treated as a part of an innovative process, is the process by which an innovation is communicated through certain channels over time among the members of a social system [40]. Research shows the importance of social network structure, which should be analysed, developed, and managed for continuous innovation in organizations [41].

In the area of scientific research knowledge diffusion can be defined as the adaptations and applications of knowledge documented in scientific publications and patents [42].

Technology diffusion is a complex social communication process. According to [43] technology diffusion starts with an innovation generated by a particular source. After that, potential adopters are informed about the availability of the new technology and persuaded by contact with prior users to adopt them. It is important to understand that technology diffusion is very sectoral. Ribeiro et al. [44] show that any given innovation related to an area X is intensely used and diffused only within its specific area and will hardly permeate into other areas.

In a learning organisation knowledge diffusion is a process of knowledge communication and learning [45]. Moreover, the close relation among members results in strong willingness of knowledge diffusion.

It is important to say that tracing knowledge diffusion is a challenging issue due to the extreme complexity of diffusion

processes [42]. Morone and Taylor [46] point out that knowledge diffusion is a complex social phenomenon which consists of, among others, knowledge spillover, knowledge transfer, and knowledge integration. The nature of human interactions and information flow is affected mainly by the creation of new knowledge and the process of learning at an individual level [47]. Moreover, the different organizational and teamwork structure conducts different knowledge behaviours and their performance. However, we should have in mind that for group of people learning efficiency is in most cases accelerated [48]. In addition, diffusion makes neighbouring agents tend to display similar knowledge levels [49]. Social influence theories provide an interpretation that different social proximity evokes distinguished contagion effects [48]. The best learning outcome can be determined by the best suitable payoff schemes and network structure changes within a complex social network [47]. In other words the effectiveness of the diffusion is a function of the network structure and seeding strategy used in delivering the initial broadcast [50].

From the economic point of view the knowledge diffusion process is related to the transfer of intellectual capital. Knowledge diffusion takes place through worker mobility [51] and the research task is related to finding the equilibrium between the host and the mobile worker. The [52] model offers a quantitative approach to explore the dynamic relationship between knowledge value and enterprise benefits in a given period.

At cognitive level the research of knowledge diffusion is related to the problem of [53] how do individuals perceive and cognitively represent the social networks that surround them and how do individuals' perceptions of their social networks affect their behaviours and outcomes.

The next important issue, related to knowledge transfer, is homophily, defined as tendency of people to associate relatively more with those who are similar to them than with those who are not [54]. Golub and Jackson [54] show that homophily and the segregation induces in networks have important consequences for processes of interest, particularly the ones of information flow.

Knowledge diffusion must be based on efficient communication channels between all actors. The importance of such efficient channels is empirically supported by MacGarvie [55] who shows that technological knowledge diffusion is faster in countries which share a common official language, whose inventors communicate more frequently by phone and are geographically closer to each other.

The process of diffusion of knowledge is based on several communication mechanisms [56]: formal way of communication through documents, databases, face-to-face meetings, e-mails, videoconferencing, and social communication (excluding commercial transaction) throughout communities—or networks—of practice. Typical artefacts are opinion, practice, and know-how [57]. The knowledge benefits can be externalized from the following three knowledge sources [58]: (1) the use of original knowledge inside the organization; (2) the improvement of original knowledge due to internal investment; and (3) the integration of innovative knowledge.

From one point of view knowledge diffusion is intended by the organization. According to Canals [56] the diffusion process takes place in a formal way through the use of documents and databases or through interaction in face-to-face meetings or by using technological means as e-mail or videoconference. From the other point of view unintended diffusion of knowledge is performed in accordance with knowledge spillovers process. The diffusion process takes advantage of the social relationships between employees of the firms, be it of a professional type through communities—or networks—of practice or more of a personal nature [56]. The most important issue is to combine in knowledge diffusion the intended knowledge diffusion mechanisms and unintended spreading of knowledge. For such reason knowledge diffusion is not equivalent to other diffusion processes modelled in natural sciences as epidemics or in social sciences (e.g., like the spread of rumors) [56]. Nevertheless, there are some attempts to do that, for example, tacit knowledge diffusion modelled as a SIR epidemic transformation [59].

3.2. Knowledge Diffusion Modelling. The problem of knowledge diffusion is an important element of complex network theory application. Based on the literature analysis we can recognise two approaches to problem modelling [60]. The first one focuses on knowledge exchange behavioural patterns between a pair of individuals. The cognitive and social psychology and economics investigated absorptive capability, effectiveness, and stimulation of knowledge share [61]. The second one used computer simulation to discuss the influence of the topology of social networks [60]. The simulation results show in what way the network structural characteristics influence knowledge diffusion.

When discussing the knowledge diffusion modelling we should keep in mind two dimensions of this problem: network topology and design of interaction rules driving knowledge transmission. Many studies show that the effectiveness of the diffusion mainly depends on the network structure and the seeding strategy used [50]. The problem of network topology analysis is solved by the utilisation of an existing network models which support real-world phenomena such as power-law (“scale-free”) degree distributions, high clustering, and short network diameter. In addition some authors make a debate about how accurately present models and corresponding analytic solutions or simulations render real-world network [62].

The main concepts of knowledge transmission mechanism, according to [62], are pay-off based models [63] and opinions vectors, continuous or discrete, one-dimensional [64], or n -dimensional [65]. The other issue is a mechanism of knowledge diffusion; most of models are aiming to maximize the spread of influence in a network and they are based on assumed rather than measured influence effects [66].

The complex nature of knowledge diffusion problem is difficult to conceptualise and formalize. However, there are some propositions in the literature. In [67] formal approach to create integrated ontology, which covers a number of learning activities, is proposed. Due to utilization of OpenCyc framework the way to computational semantics is clear. The real application of complex ontology, which is formalised for

computer-aided control engineering, can be found in [68]. However, before we begin to think of the formalization, we should try to define the based learning process in order to recognise the objectives outside the individual and the transformation of these activities into measurable, efficient behaviour [69]. The presented literature gives some idea how to formalize different parts of knowledge diffusion model.

3.3. Related Work in the Areas of Knowledge Diffusion Models. A number of papers studied a model of a population of agents whose interaction network coevolves with knowledge diffusion and accumulation. General idea of the proposed model is based on the Cowan and Jonard model (CJ) [70]. Similar to CJ model the proposed model is designed to capture effects of incremental innovation and their diffusion over a network of heterogeneous agents. The CJ model assumes [71] the following:

- (i) agents are arranged in one-dimensional space;
- (ii) each agent occupies one vertex and may interact with their k nearest neighbours on either side;
- (iii) the population of individuals is endowed with different levels of initial knowledge;
- (iv) a small number of agents are “experts” and are endowed with a high level of knowledge in at least one value of the vector;
- (v) all individuals interact among themselves, exchanging information;
- (vi) knowledge is a nonrival good and can be transferred without decreasing the level of knowledge possessed by each trader.

In our work we extended the classical CJ model to multidimensional vertical and horizontal diffusion scheme. Moreover, new mechanism of knowledge processing was introduced including self-learning and forgetting processes. Some authors noticed the importance of these factors (e.g., dissemination ability and knowledge forgetting in Geng and Mao work [72]). However, this phenomenon is not regularly analysed because of the complexity.

In [73] two processes on the network are proposed: knowledge diffusion refers to the distribution of existing knowledge in the network, while knowledge upgrade means the discovery of new knowledge. Additionally, authors took into account the agent’s knowledge absorptivity and forgetting factors represent some cognitive ability of agent. However, the proposed model works only for one type of knowledge; absorptivity and forgetting factors are constant and not associated with agent’s network localisation.

The paper [74] focuses on knowledge diffusion as an economic process of different types of knowledge exchanging. Similar to previous work the paper covers the knowledge diffusion process (agent broadcasts his knowledge to the agents to whom he is directly connected) and knowledge creation process (agents receive new knowledge which is combined with their existing knowledge stocks). However, this paper only examines the relationship between network architecture and aggregate knowledge levels.

The key factors that affect the speed and the distribution of knowledge diffusion are identified by Morone et al. [75]. During knowledge transfer, knowledge is mastered through a backward process by which it is confronted and articulated with previous knowledge represented in form of a cognitive map. This paper is focused on knowledge dimension during diffusion process and does not cover the issue of agent's capabilities and diffusion of different kinds of related knowledge.

The diffusion of different kinds of knowledge in an organisation can be interpreted as a multilayers network (share the same set of nodes connected with many links grounded on different layers). In the literature the main emphasis is placed on multilayered diffusion processes through a multilayered material for a wide range of applications, including industrial, biological, electrical, and environmental areas [76]. Nevertheless, there is some activity in the area of multilayered knowledge diffusion. The example is the work [77], where numerical simulation described nonlinear relation between layers representing various network models. Diffusion models are the linear threshold and independent cascade. They are only focused on diffusion process excluding knowledge upgrade/creation process. Moreover, the model missed the relation with cognitive characteristic of actors and related self-learning and forgetting processes.

So far there has been no work combining the competence with knowledge diffusion. Moreover, the proposed model with vertical and horizontal diffusion for multilayer organisation graph with self-learning and forgetting processes is a new approach to the problem of knowledge diffusion.

4. Model of Multilayer Knowledge Diffusion

4.1. Approach to Competence Modelling in the Diffusion Model. Modelling of the competence development based on the knowledge diffusion process requires a new approach. There are a couple of reasons for that. Competence cannot be changed directly; we can only influence its components. The value of the components of competence (see Table 1) can be determined.

In the proposed model, the components of competence are represented by the layers in the multilayer graph. From the point of view of the diffusion process, the content of layers is invalid and the diffusion process is focused on knowledge flow and not their content or meaning. We define only the relationship between their elements, which may be damping (weakening) or forcing (strengthening). This approach is similar to Shannon's information theory where content of the messages has no meaning. However, there is an opinion in the competence literature that the process of competence computing should be understood as enabling the use of competence databases for inference and combination of competencies for different functions and processes, not as a reductionist account of competencies to numeric models [14]. In our case we focused on knowledge flow and linked it to the diffusion process, which is based on mathematical transformations.

Proposition of 3-class layers model for diffusion model for competence development [29] is as follows.

- (i) Class 1: know-how—practical, hands-on forms of knowledge gained through incremental improvements to products and processes.
- (ii) Class 2: know-why—theoretical forms of understanding that enable the creation of new kinds of products and processes.
- (iii) Class 3: know-what—a strategic form of understanding about the value creating purposes to which available know-how and know-why forms of knowledge may be applied.

We assume that one layer in our model is dedicated to one kind of knowledge, which belongs to one class of competence's components. The question arises: can each component of competence be called knowledge? The works [10, 25] prove that it is correct and knowledge can be expressed in various forms.

- (i) Knowledge can be explicitly formalized—texts, documents multimedia.
- (ii) Knowledge can be a practice—it rests on the accumulation of experiments.
- (iii) Knowledge can be tacit—all cannot be formalized. Its transmission requires suitable means: conversation, training, joint work, and so forth.
- (iv) Knowledge can be social—the technical know-how of a company does not rest on an individual but on the interaction of all the members of its technical community. It is while collaborating, by confronting their points of view, that these technicians create and finally hold new knowledge.
- (v) Knowledge is dynamic and evolves/moves in time.

In our approach the term *knowledge* will be used for description of all components of competence. Based on the term *knowledge*, all competence's components, like instances of class: knowledge, skills, and behaviour, can be modelled in a multilayer graph as a single layer.

4.2. Knowledge Network Definition

4.2.1. Knowledge Domain. Every node in the network represents single knowledge worker. According to [4] the knowledge employee's main tasks related to knowledge are capture/extract, analyse/organise, find, create/synthesize, and distribute/share. In the organisation's network different types of knowledge are propagated in order to acquire competence by employee. Moreover, the productivity of knowledge workers is enhanced through competence enhancement and learning which take place directly at workers' workplaces [13].

Definition 1 (organization members). The organization X composed of knowledge worker is determined by index i . $I = \{i : i \in N_+\}$.

Definition 2 (knowledge domain in organisation). The knowledge domain in organisation X is represented by the set $K = \{k_j : k_j \geq 0\}$; the elements of set are indexed by j ,

$J = \{j : j \in N^+\}$. The element k_j is a knowledge element that represents some related part of domain of discourse. In the analysed domain the last position of J is j^* .

The methods of knowledge modelling mainly focus on the formally representation of relationship between different areas/element/types of knowledge. The best way to do is to use the ontological approach. Ontology is a formal, explicit specification of a shared conceptualization [78]. The main components of an ontology are concepts, relations, instances, and axioms [79]. The relations describe the interactions between concepts or a concept's properties. In the problem of knowledge diffusion is essential to show the existence of the relationship between areas/elements of knowledge. Because defining the types of relationship between is not very useful for numerical processing and difficult to determine, we focused on causality and mutual order of knowledge areas/elements. Therefore, we choose the method of presentation of the knowledge domain based on the knowledge space theory [80].

In our approach the competence acquisition is a result of combination of different competence's components [81]. In other words the proper combinations of different knowledge elements, which reflect competence's components, result in an efficient competence acquisition by a knowledge worker. Competence is an ability to find an effective way to theoretical knowledge usage in order to solve a practical problem and the ability to verify the solutions. We use the knowledge space theory to describe the relation between knowledge elements in domain K .

Based on the knowledge space theory let (K, \leq) be a partial ordered set. In this theory the prerequisite relationship is cover by the surmise relation [82] and function $<$ represents prerequisite relationship. Two knowledge items a and b are in surmise relation $a < b$ if, whenever a person has solved/maintained item b correctly, we can surmise that this person is also able to solve item a correctly [83]. According to [84] we say that for $k_a, k_b \in K$, k_b covers k_a denoted by $k_a < k_b$ if $k_a < k_b$ and for $\forall k_c \in K$ and $k_a \leq k_c < k_b$ implies $k_a = k_c$. Moreover, the graph $(K, <)$ is a Hasse diagram for K . The important assumption for future discussion is that due to cognitive nature of problem and mutual relation between them, not all potential knowledge states have to be observed [85].

Let $k_a, k_b \in K$, we say that element k_a is maximal element in K if $\forall k_b \in K$, $k_b \geq k_a$, then $k_b = k_a$. We denoted this by symbol k^{\max} . Similarly, we say that element k_a is minimal element in K if $\forall k_b \in K$, $k_a \geq k_b$, then $k_a = k_b$ and the dedicated symbol is k^{\min} .

For (K, \leq) and $k_a, k_b \in K$, the lower shadow of k_a is a set $\{k_b \in K, k_b < k_a\}$ denoted by Δk_a and consequently the upper shadow of k_a is a set $\{k_b \in K, k_a < k_b\}$ denoted by ∇k_a . Moreover, for $k_a > 0$, all elements of its lower shadow Δk_a have to be greater than zero $\{k_b : k_b \in K, k_b < k_a, k_b > 0\}$.

4.2.2. Knowledge Network. In a knowledge network the node actively processes knowledge and edges represents channels for knowledge relocation [86].

Definition 3 (knowledge network). The knowledge network for organisation X is $G^X = \{G_j\}$ multilayer graph of graphs G_j , where $j \in J$ is an index of knowledge in K . Every knowledge element form set K is represented by a single layer in G^X .

Definition 4 (knowledge layer). Every layer in G^X is a undirected graph without self-loops $G_j = (V, E_j, f_j)$, where $V = \{v_i\}$ is a set of nodes representing knowledge worker i , $E_j \subseteq V \times V$ is a set of edges representing symmetrical relationship between nodes (knowledge employee) on layer j , and $f_j : E_j \rightarrow \mathfrak{R}^+ \cup \{0\}$ is a variable edge labelling function.

For any node from G_j we can recognise knowledge worker's neighbourhood. The $e^j(v_{i_1}, v_{i_2})$ is a binary variable for $v_{i_1}, v_{i_2} \in V$ and $i_1, i_2 \in I$. If connection between v_{i_1} and v_{i_2} exists at layer j , then $e^j(v_{i_1}, v_{i_2}) = 1$; otherwise $e^j(v_{i_1}, v_{i_2}) = 0$. The neighbourhood of node i_1 at layer j is a set $\Gamma_{i_1}^j = \{v_{i_2} : v_{i_2} \in V, e^j(v_{i_1}, v_{i_2}) = 1\}$.

Definition 5 (knowledge vector for worker i). Based on the multilayer graph G^X we can formulate the knowledge vector for worker i : $\bar{K}_i = [k_{1,i}, k_{2,i}, \dots, k_{j,i}, \dots, k_{j^*,i}]^T$, where $k_{j,i} \geq 0$ and $k_{j,i}$ represent the value of knowledge on layer j for worker i . Moreover, any worker i has the knowledge set related in its own way: (K_i, \leq) .

In order to distinguish personal abilities for communication on knowledge level, two parameters are introduced:

- (1) cognitive abilities for node v_i : $o_i \in \langle 0, 1 \rangle$ variable for cognitive abilities for node v_i . The highest o_i is the fastest the actor behind i is able to learn and acquire knowledge from others in order to increase his/her knowledge level;
- (2) social abilities for node v_i : $l_i \in \langle 0, 1 \rangle$ variable for social abilities for node v_i . The highest l_i is the fastest the actor behind i is able to teach others. This means that it has a social skill to adapt (personalize) communication to the recipient [87].

4.3. Structure of Competence Use in the Model. Competence can get gradually stronger, in a situation where surroundings affect and stimulate its components. For example, we acquire new skills in a training session or while working (e.g., software developers programming every day). Competence (its level) can also degrade. The most common reason for this is not using the given competence in everyday work. The other is thanks to technology progress which makes the components of competence outdated. We can distinguish different relations between competencies which affect the interaction between them. Increasing competence in a certain competence group (e.g., communication) can affect the increase of other competencies (e.g., sales of products). Next issues regarding competence processing in an organisation start to show up when we take a look from a company's perspective. From the company's point of view, certain

TABLE 2: Linking competence value with expertise level.

Approximate value of c_i	Expertise level	Description (based on [104])
0–0,2	Novice	Minimal exposure to the domain.
0,2–0,4	Initiate	Began introductory instruction to the domain.
0,4–0,6	Apprentice	Undergoing a program of domain learning beyond the introductory level. Traditionally the apprentice is immersed in the domain by a more experienced employee.
0,6–0,8	Journeyman	Person who can perform a day's labor unsupervised, although working under orders.
0,8–1	Expert	The distinguished or brilliant journeyman, highly regarded by peers, whose judgments are uncommonly accurate and reliable, whose performance shows consummate skill and economy of effort, and who can deal effectively with rare or "tough" cases. Also an expert is one who has special skills or knowledge derived from extensive experience with subdomains.
1	Master	Master is any journeyman or expert who is also qualified to teach those at a lower level. Traditionally, a master is one of an elite group of experts whose judgments set the regulations, standards, or ideals.

competencies are created only by combining the competencies of a greater number of employees. The complexity of these combined competencies is too great for a single person to obtain this kind of competence.

In this approach we do not analyse the content of knowledge included in competence. In our case we are interested in the competence's level, which allows us to analyse the knowledge and competencies growth and dynamics in the organization. In the presented method the competence value will be normalized to range $\langle 0, 1 \rangle$ in order to be compatible with scale in the literature. The level of competence is related to the expertise of an employee (see Table 2) [88]. According to cognitive science employees with more competence (expert) within their domains are skilled, competent and think in qualitatively different ways than novices do [89].

Definition 6 (competence). The competence set for organization X is defined in the following way: $C^X = \{c_a : a = 1 \cdots A, c_a \in \langle 0, 1 \rangle\}$.

There relationships between competence and knowledge in organization X are represent by the matrix $M^X[t] = \|m_{ja}[t]\|$, $m_{ja} : s(C, K) \rightarrow \mathfrak{R}^+ \cup \{0\}$, where $s(C, K)$ is a variable for competence a and knowledge j relation strength representation at time $t \in T$. The relationships in the matrix are continuously changing due to companies' priority and change in the market. If $m_{ja} = 0$ then there is no relation between competence a and knowledge j . In Figure 1 the graphical representation of matrix M^X is presented.

Based on the matrix $M^X[t]$, the competence value, at time $t \in T$, for worker i is the following:

$$c_a^i[t] = \sum_j m_{ja}[t] \cdot k_{ji}[t]. \quad (1)$$

The final value of $c_a^i[t]$ form (1) has to be normalized to fulfil $c_a^i[t] \in \langle 0, 1 \rangle$.

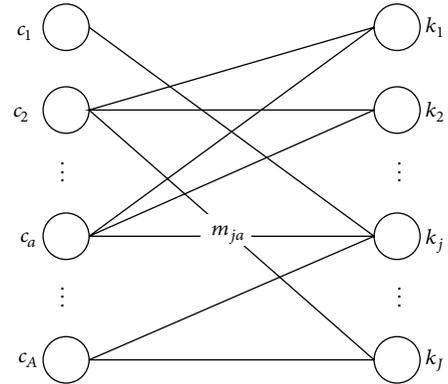


FIGURE 1: Relation mash for competencies and knowledge relation.

4.4. Process Definition. In order to analyse the competence development in an organization in addition to the structure of the network, which represents the relationships that exist between staff, we also need to describe the processes associated with the competence development. Let us introduce the time index: $T = 1, \dots, t, t + 1, \dots, t^*$.

4.4.1. Knowledge Diffusion. In the proposed model the knowledge diffusion process takes place in two directions: vertical ($D_{j,i}^v$) and horizontal ($D_{j,i}^h$).

(1) **Horizontal Knowledge Diffusion.** Horizontal process of knowledge diffusion is related to knowledge diffusion between knowledge network nodes (representing employees) on a selected layer j . In fact, it involves simulating a situation in which the relationship will be created at the level of tacit knowledge. The relationship affects the knowledge of involved employees with regard to their ability to teach and learn. In the classic Nonaka's model this process is called socialization [90]. According to [91] for knowledge sharing to be most effective, it should take place between people

who have a common knowledge and can work together effectively. The mutual relationship should be strong. Thus tacit knowledge sharing is connected to ideas of communities and collaboration.

Horizontal knowledge diffusion occurs only between active nodes. There are two possible methods of knowledge diffusion between nodes:

- (i) broadcast: the node transfers knowledge to all connected nodes;
- (ii) multicast: the node transfers knowledge to a selected set of nodes; the set of receiving nodes may be selected randomly or based on some strategies.

In our approach we focus on the multicast scheme for horizontal knowledge diffusion. Every node on selected layer of multilayer graph G^X is looking for the most effective source of knowledge in its neighbourhood on this layer. In this context, effective means being with best combination of knowledge and social abilities. As a result the horizontal knowledge diffusion occurs only when the node is able to locate node with a greater potential for knowledge transfer in his neighbourhood. In other words weak looks for strong.

Definition 7 (horizontal knowledge diffusion). Horizontal knowledge diffusion is calculated for node i on layer j based on the function α defined in the following way: $D_{j,i}^h = \alpha(k_{j,z}, o_i, l_z, d_i, f_{j,z}) \rightarrow \mathfrak{R}^+$, where $k_{j,z}$ is a knowledge value for node $z \in I$ on layer j , o_i is a cognitive (learning) abilities for node v_i , l_z is social (teaching) abilities for node v_z , d_i is a component responsible for node i degree distribution, and $f_{j,z}$ is a strength of the relationship between nodes j and z . z is a node $v_{j,z} \in \Gamma_i^j$ with maximal combination of cognitive and social abilities. The horizontal knowledge diffusion occurs only if $k_{j,z} \cdot l_z > k_{j,i} \cdot o_i$ for $v_{j,z} \in \Gamma_i^j$.

The interpretation of knowledge diffusion function depends on the purpose and goals of the organization. In addition, the final function form depends on the specific nature and structure of knowledge networks and knowledge resources in an organization. For the purposes in the paper we have proposed functions ((2)–(8)) for implementing the various described processes. It is clear that individual processes can be expressed by other functions depending on the intentions of the designer.

Let us propose some form of function α :

$$D_{j,i}^h [t] = \frac{(k_{j,z} [t] - k_{j,i} [t]) \cdot l_z \cdot o_i}{d_z \cdot A} \cdot d_{\max} \cdot f_{j,z}, \quad (2)$$

where d_{\max} is a maximal node degree on layer j and node z has $\max(k_{j,z} \cdot l_z)$ for $v_{j,z} \in \Gamma_i^j$. A is a fixed value. In function (2) we take into consideration the possible time to give of node $v_{j,z}$. If the node $v_{j,z}$ degree is higher than the selected knowledge diffusion is limited by other relations. Moreover,

higher cognitive abilities and nodes relationship strength also support knowledge diffusion in positive way.

(2) *Vertical Knowledge Diffusion*. Vertical knowledge diffusion takes place in a single node and occurs between its knowledge layers. Generally speaking the knowledge value increasing on layer j may increase the knowledge value on other layers. The relationship between layers can be deducted from (K, \leq) or described by organisation members and saved in a dedicated matrix. Moreover, the vertical knowledge diffusion process is an internal process in contrast to the horizontal knowledge diffusion which is an external process for a knowledge worker.

Let us define the vertical diffusion matrix for worker i : $M_i[t] = \|r_{j_1, j_2}^i [t]\|$, $M_i \subset K_i \times K_i$, where $r_{j_1, j_2}^i \geq 0$, $r_{j_1, j_1}^i = 0$, and $j_1, j_2 \in J$. The matrix may vary with time due to changing relation between knowledge according to technological change and innovation. The $r_{j_1, j_2}^i [t]$ described relations between layers and a single knowledge type at time t . If knowledge value $k_{j_1, i}$ is increasing then the value $k_{j_2, i}$ is increased according to $k_{j_2, i} [t + 1] = k_{j_2, i} [t] + r_{j_1, j_2}^i [t] \cdot k_{j_1, i} [t]$.

Definition 8 (vertical knowledge diffusion). Vertical knowledge diffusion for nodes $v_{\cdot, i}$ causes change of the knowledge value on layer $j \in J$ and affected other layers $g \in J \setminus \{j\}$ according to matrix M_i at time t . Vertical knowledge diffusion is calculated for layers g based on the function μ : $D_{j,i}^v(k_{j,i}) = \mu(r_{j,g}^i, k_{g,i}) \rightarrow \mathfrak{R}$, where $r_{j,g}^i$ is a value of diffusion relationship between layers j and g for node i , and $k_{g,i}$ is a knowledge level for node i on layer g . The function for vertical knowledge diffusion has argument $(k_{j,i})$ containing initial information about the value of knowledge increasing for node i on layer j . This increased value has to be diffused between the other layers. The vertical knowledge diffusion occurs for node $v_{\cdot, i}$ if at any layer the knowledge value has been changed $\exists n \in J : k_{n,i} [t + 1] \neq k_{n,i} [t]$.

Let us propose some form function for vertical knowledge diffusion:

$$k_{j,n} = r_{j,n}^i \cdot k_{j,i}, \quad (3)$$

where $n \in J$ is a layer's index.

The function (3) is one of the possible linear relationships. For the real processes the matrix M_i has to be defined by the expert from organisation with some cognitive competencies. As a result the relationship between different layers can be nonlinear, nested, and with feedback.

4.4.2. *Knowledge Deterioration (Forgetting)*. Over time employee competencies (knowledge) are reduced if they are not stimulated by the workers from surrounding and the work itself. From the formal point of view knowledge forgetting model can be found in [92]. The main concept is to incorporate in knowledge model the fact that agents did not always remember their previous knowledge (i.e., agents have perfect recall). Sometimes we want to model

the fact that certain knowledge is discarded. From the business point of view organizations must forget old habits in order to learn new and more appropriate ways of doing things [93]. Organisation may be forgetting knowledge intentionally (avoiding bad habits, unlearning) and accidentally (failure to capture, memory decay) [94]. From the cognitive science point of view, men develop their skills in an environment that stimulates them [89]. In this case, when my coworkers are less competent, with time my capacity will decrease (equal to their average level). A lot of works in psychology show that the environment impact on our activity. In the case of knowledge processes uninspiring surroundings cause progressive loss of our knowledge.

Let us introduce a formula for average knowledge transfer potential for node neighbourhood calculation:

$$\tilde{k}_{j,i} [t] = \sum_{k_{j,q} \in \Gamma_i^j} \frac{k_{j,q} [t] \cdot l_q}{\text{card}(\Gamma_i^j)}, \quad (4)$$

where $\text{card}(\Gamma_i^j)$ is a number of nodes in i neighbourhood. If the value of (4) is less than worker knowledge acquisition potential (product of worker's knowledge level and his/her cognitive abilities), then the worker's knowledge starts to deteriorate.

Definition 9 (knowledge deterioration (forgetting) process). If for node i on layer j following condition occurs $k_{j,i} [t] \cdot o_i \geq \tilde{k}_{j,i} [t]$, then forgetting process is described by the formula $F_{j,i} = \chi(k_{j,i}, \tilde{k}_{j,i}, o_i)$, where χ is a forgetting function $\chi(k_{j,i}, \tilde{k}_{j,i}, o_i) < k_{j,i}$, $k_{j,i}$ is a knowledge level for node i on layer j , $\tilde{k}_{j,i}$ is an average knowledge transfer potential for node i on layer j neighbourhood, and o_i is a cognitive abilities for node v_i . Moreover, the forgetting process occurs also vertically, $D_{j,i}^v(-F_{j,i})$.

In proposed function $F_{j,i}$ for node i on layer j the forgetting factor is related to node neighbourhood Γ_i^j and knowledge worker's cognitive ability. In general, worker with high cognitive ability forgets slower and the worker is forced to start forgetting by the weakness of his neighbourhood.

Assumption 10 (nonzero knowledge condition). If the knowledge value for node i on layer j is greater than zero in the next steps of the knowledge value has to be always greater than $\Omega > 0$: $k_{j,i} [t] > 0 \rightarrow k_{j,i} [t + \hbar] \geq \Omega$: $\hbar = (1, \dots, t^* - \hbar)$. The minimal value of knowledge value is represented by the variable: $\Omega > 0$.

One of the forgetting formula propositions is the following:

$$\begin{aligned} k_{j,i} [t + 1] &= k_{j,i} [t] - \Xi_j \\ &= k_{j,i} [t] - \frac{(k_{j,i} [t] - \tilde{k}_{j,i} [t]) \cdot (1 - o_i)}{B}, \end{aligned} \quad (5)$$

$$\begin{aligned} D_{j,i}^v(-k_{j,i}) &\rightarrow \forall n \in J : k_{n,i} [t + 1] \\ &= k_{n,i} [t] - \Xi_n \\ &= k_{n,i} [t] - r_{j,n}^i \cdot \left(\frac{(k_{j,i} [t] - \tilde{k}_{j,i} [t]) \cdot (1 - o_i)}{B} \right), \end{aligned} \quad (6)$$

where B is a fixed value. Moreover if for (5) $k_{j,i} [t] < \Xi_j$ then we set the value of $k_{j,i} [t + 1] = \Omega$ and, respectively, for (6) $\forall n \in J : k_{n,i} [t] < \Xi_n \rightarrow k_{n,i} [t + 1] = \Omega$. The variable Ω is the set by the organisation management lowest acceptable value of knowledge in system.

4.4.3. Self-Learning. Due to the rapid obsolescence of knowledge and the requirements of increasingly complex processes there is a need to continuously acquire new knowledge by employees. Lifelong learning philosophy [95] assumes that any worker maintains ongoing, voluntary, and self-motivated pursuit of knowledge. In the proposed model this movement is described as a self-learning process.

Definition 11 (self-learning process). If for node i on layer j the following condition occurs $k_{j,i} [t] \cdot o_i < \tilde{k}_{j,i} [t]$ then self-learning process is described by the formula $S_{j,i} = \delta(k_{j,i}, \tilde{k}_{j,i}, o_i, cc_i)$, where δ is a self-learning function, $k_{j,i}$ is a knowledge level for node i on layer j , $\tilde{k}_{j,i}$ is an average knowledge transfer potential for node neighbourhood, cc_i is a clustering coefficient of node $v_{j,i}$, and o_i is a cognitive abilities for node v_i . Moreover, the self-learning process occurs also vertically, $D_{j,i}^v(S_{j,i})$.

The function δ incorporated node's surrounding and cognitive ability into self-learning process. If the average knowledge level of Γ_i^j is higher than node's knowledge then knowledge worker has to invest some time in order not to stand out from the rest and be valuable for communication. Moreover, the high clustering coefficient reflects larger environment that may have more pressure due to high cliqueness. In this paper we propose the following self-learning formula.

For $cc_i > 0$,

$$\begin{aligned} k_{j,i} [t + 1] &= k_{j,i} [t] + \Psi_j \\ &= k_{j,i} [t] + \frac{(\tilde{k}_{j,i} [t] - k_{j,i} [t]) \cdot o_i \cdot cc_i}{C}, \\ D_{j,i}^v(k_{j,i}) &\rightarrow \forall n \in J : k_{n,i} [t + 1] \\ &= k_{n,i} [t] + \Psi_n \\ &= k_{n,i} [t] + r_{j,n}^i \cdot \frac{(\tilde{k}_{j,i} [t] - k_{j,i} [t]) \cdot o_i \cdot cc_i}{C}. \end{aligned} \quad (7)$$

```

FOR all subsequent element  $l \in \mathfrak{R}^{\text{layers}}$ 
  FOR all subsequent element  $n \in \mathfrak{R}_l^{\text{nodes}}$ 
    Execute horizontal knowledge diffusion process:  $D_{l,n}^h$ 
    Execute vertical knowledge diffusion process:  $D_{l,n}^v(k_{j,i})$ 
    Calculation average knowledge transfer potential for node neighbourhood  $\tilde{k}_{l,n}$ 
    If  $k_{l,n} \cdot o_n \geq \tilde{k}_{l,n}$  then execute forgetting process  $F_{l,n}$ 
      Execute vertical knowledge diffusion process:  $D_{l,n}^v(-k_{j,i})$ 
    If  $k_{l,n} \cdot o_n < \tilde{k}_{l,n}$  then execute self-learning process  $S_{l,n}$ 
      Execute vertical knowledge diffusion process:  $D_{l,n}^v(k_{j,i})$ 

```

PROCEDURE 1: Procedure of knowledge diffusion.

For $cc_i = 0$,

$$\begin{aligned}
k_{j,i}[t+1] &= k_{j,i}[t] + \Psi_j \\
&= k_{j,i}[t] + \frac{(\tilde{k}_{j,i}[t] - k_{j,i}[t]) \cdot o_i}{D}, \\
D_{j,i}^v(k_{j,i}) & \\
&\rightarrow \forall n \in J : k_{n,i}[t+1] \\
&= k_{n,i}[t] + \Psi_n \\
&= k_{n,i}[t] + r_{j,n}^i \cdot \frac{(\tilde{k}_{j,i}[t] - k_{j,i}[t]) \cdot o_i}{D},
\end{aligned} \tag{8}$$

where C and D are fixed values.

4.5. Procedure of Knowledge Diffusion. Competence development based on the knowledge diffusion involves various processes: horizontal knowledge diffusion, vertical knowledge diffusion, knowledge forgetting, and self-learning. In addition, knowledge diffusion is a two-dimensional process. In this section we will develop the main points of procedure for knowledge diffusion calculation in multilayer networks.

Layer selection is based on layer's ranking $\mathfrak{R}^{\text{layers}}$, calculated based on vertical diffusion matrix for variable \hat{k}_w :

$$\hat{k}_w = \sum_i \sum_g r_{w,g}^i, \tag{9}$$

where $w, g \in J$. As a result the ranking consists of indexes of layers with order determined by the value of \hat{k}_w . Taking into consideration all workers we analysed what layer has the biggest influenced on other layers $\max(\hat{k}_w)$. If the layers have the same value of \hat{k}_w the order is selected randomly. The ranking starts with layer with biggest \hat{k}_w .

Node selection is based on node's ranking $\mathfrak{R}_j^{\text{nodes}}$, generated with regard to knowledge level of all nodes on analysed layer $j \in J$. Due to "pull" nature of described knowledge diffusion process the ranking starts with the node with lowest value of knowledge Procedure 1 of knowledge diffusion is as shown below.

The logic of horizontal diffusion refers to the analysis of components' values on single level. Because we operate

on single layer the problem of granularity scales of competence's component is not a crucial one. The problem could arise in terms of relations between layers (which represent competence components). Mutual vertical relations between layers can be nonlinear and based on own logic. In presented approach the matrix M_i is dedicated to competence based transformation. If the competence components are reflected at different levels of granularity then we need to maintain the normalization process.

The presented approach, from the methodical point of view, is an agent based simulation. This kind of simulation is adequate for such systems with complex interaction. In our simulation we based on the NetLogo framework [96].

5. Areas of Applications

5.1. Competence Management Paradigm. The model of knowledge diffusion is used to analyze the development of competencies in an organization. However, its final application is competence management. Keeping in mind main management axioms [97] we are going to discuss whether competence management is possible based on the proposed model.

- (i) Axioms of management 1: the object is suitable for observations and measurements.

The presented model gives ability to observe and measure every component of competence. Moreover, all knowledge flow between actors can be tracked and analysed.

- (ii) Axioms of management 2: at the interval of observation object can change its state.

The dynamics of knowledge flows on different levels of network is noticeable. The process of knowledge diffusion is based on the continuously changing value of actors (workers) knowledge.

- (iii) Axioms of management 3: the predetermined target defined expected object's state.

In the organization the target is set on the strategic level and concerns for expected values of competencies.

- (iv) Axioms of management 4: there are alternative ways to influence the behaviour of an object.

Any types of knowledge, components of competence, can be changed by training (increase of knowledge level), expert's mentoring (direct diffusion of knowledge), or team building (network reconfiguration).

- (v) Axioms of management 5: there is a predefined criterion of management efficiency.

The criterion determines the degree of matching acquired competencies to market or company requirements.

- (vi) Axioms of management 6: there are resources for the execution of the decision.

The network consists of nodes which represents knowledge workers (actors).

Moreover, in the discussed context, competence management is a process of tracking changes in the content of knowledge related to the competencies.

5.2. Simulation Model. All the concepts of knowledge diffusion models require validation. In real conditions only few models can be checked due to the limitation of data. As a result, a number of simulation network models are used. The description of models can be found in [98]. The literature review shows that the diffusion models are validated based on the Watts-Strogatz model [99]. The Watts-Strogatz model reflected the "small-world" characteristic of complex network. According to Cowan and Jonard [70, 74] "small-world" networks generate the fastest knowledge growth. Moreover, Cowan and Jonard found that the steady-state level of average knowledge is maximal when the network structure is a small world. It means that most connections are local, but roughly 10% of them are long distance. The relatively big clustering coefficient is beneficial for knowledge diffusion in the agents' network [45]. The Watts-Strogatz model combines a strong degree of local cohesiveness with a small fraction of long distance links permitting knowledge to be circulated rapidly among distant parts of the network.

At the beginning the network generated by the Watts-Strogatz model is a regular network, and it can rewire from the regular network to the random network by adjusting the parameter p . In the literature in order to generate a small world network for diffusion process validation the parameter p is set as $p = 0, 1$ [70, 74, 75, 100]. For $p = 0$ the network is regular and $p = 1$ generated random network.

The analyzed simulation model contains 500 nodes (agents). The network is generated based on the Watts-Strogatz model for $p = 0, 1$. We are going to simulate 4-layer model, where the relations between the layers are the following: $n_2 < n_1, n_3 < n_2$, and $n_4 < n_2$ (see Figure 2). For a different simulation proposal the value of the competence matrix M^X , the vertical diffusion matrix M_i (the same for all workers), node's cognitive and social abilities, and knowledge vector for worker will be set randomly. The workers are divided into two groups: "normal" workers and experts with knowledge level significantly greater than other workers.

5.3. Applications. Due to high stochastic nature of competence development process and multidimensionality of the

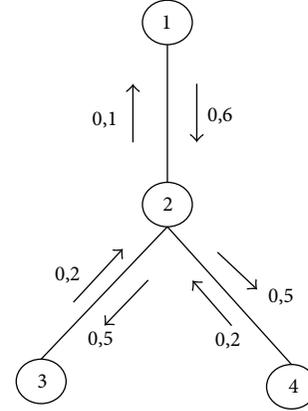


FIGURE 2: Hierarchical structure for knowledge.

TABLE 3: Symmetric settings for vertical diffusion.

	1	2	3	4
1	0	0,4	0,4	0,4
2	0,4	0	0,4	0,4
3	0,4	0,4	0	0,4
4	0,4	0,4	0,4	0

proposed model the deep simulation analysis is very difficult to maintain. In order to illustrate the different aspects of the proposed model, in the context of competence management, we will discuss a number of case studies.

5.3.1. Modelling the Competence Development Based on Multilayer Diffusion. The proposed model was verified during simulations in terms of multilayer diffusion and development of competence. Simulations were performed on Watts-Strogatz network with 0.1 rewiring probability. Initial knowledge to each worker was assigned randomly from the range (0,5) and maximal expert knowledge was assigned to the level of 30. The number of experts was assigned to 3% of all network nodes. Simulations were performed at model parameters with assigned values $A = 2.0, B = 0.1, C = 2.0$, and $D = 2.0$ (for formulas (2)–(8)). Structure of competence c_1 was based on the vector (0.5, 0.5, 0.0, 0.0) assigned to knowledge k_1, k_2, k_3 , and k_4 , respectively, for c_2 (0.10, 0.20, 0.30, 0.40), for c_3 (0.25, 0.25, 0.25, 0.25), and uniform distribution for c_4 with values (0.40, 0.40, 0.10, 0.10). At the first stage simulations were performed without vertical diffusion and show process for knowledge diffusion independent for each component of competence. In the second stage settings for vertical diffusion were based on symmetric relation between layers with the same symmetric intralayer diffusion at the level 0.4 with relations showed in Table 3. Third stage of simulation was configured using structure of dependencies and matrix setting shown, respectively, in Figure 2 and Table 4.

Results for model with disabled vertical diffusion similar to earlier models are presented in Figure 3. For each knowledge workers (nodes) were activated learning, self-learning, and deterioration parameters. Results based on assigned parameters show ability to simulate processes and

TABLE 4: Matrix based setting for vertical diffusion.

	1	2	3	4
1	0	0,6	0	0
2	0,1	0	0,5	0,5
3	0	0,2	0	0
4	0	0,2	0	0

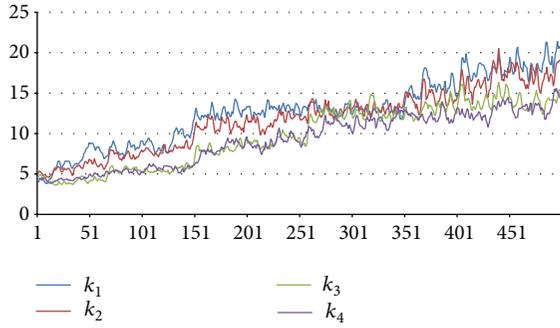


FIGURE 3: Results of simulation with disabled vertical simulation.

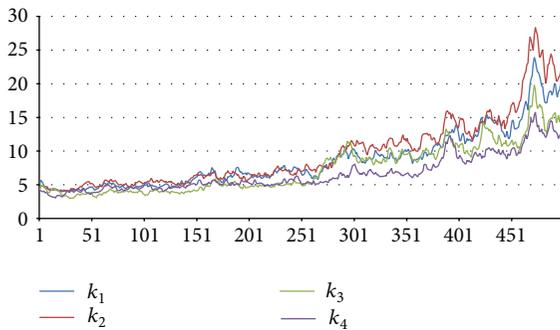


FIGURE 4: Results of simulation with symmetric vertical simulation.

changes within results in terms of deterioration and acquiring knowledge. In the next stage, the simulation was performed with active vertical diffusion based on symmetrical settings and the result is presented in Figure 4.

Simulation model allows tracking incoming and outgoing knowledge for each component. Results for incoming and for outgoing knowledge at each layer are visible in Figures 5 and 6. Monitoring knowledge diffusion in terms of incoming and outgoing knowledge makes possible tracking effectiveness and implementing strategies improving the process. Using presented approach it is possible to detect problems at this level and make diffusion more effective. Activated vertical diffusion resulted in higher total average knowledge in each layer obtained in the end of process.

Combining between model knowledge components and building metric for competencies makes possible tracking different competence development over time. In Figure 7 competence development for each competence that can be modelled using different relations between knowledge components and each competence is shown.

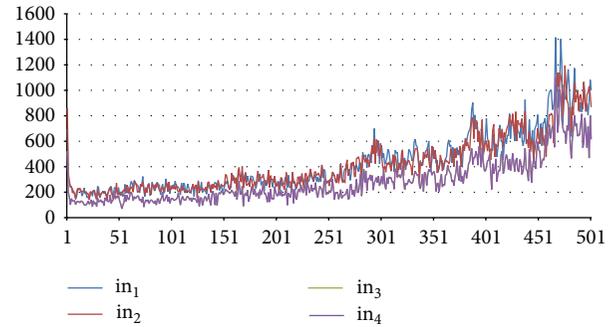


FIGURE 5: Knowledge incoming from vertical diffusion based on symmetric relations.

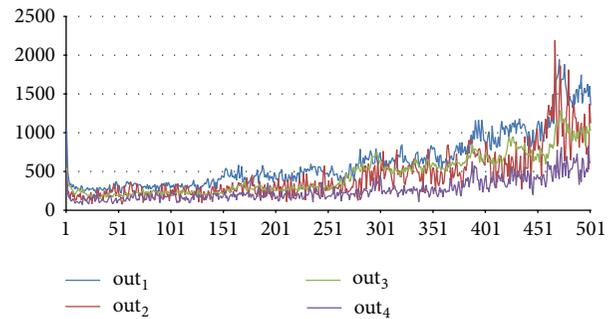


FIGURE 6: Knowledge outgoing from vertical diffusion based on symmetric relations.

In the next step simulation is based on the asymmetric settings for vertical diffusion and results for knowledge in each layer are shown in Figure 8. Results for incoming knowledge in each layer are visible in Figure 9.

Parameters used for different relations between layers can be changed over time and they are related to the situation with emerging technologies and innovations. Obtained results showed diffusion processes between different layers. Dynamics of processes was simulated using both vertical and horizontal diffusion. Effect of deterioration was simulated as well as self-learning which results in changes over the time.

5.3.2. *Modelling the Role of Experts.* In the next step the role of experts within the network was modelled using asymmetric relations between knowledge components. Using the proposed model it is possible to simulate changes after adding experts with assigned knowledge higher than all network members. In the first stage of the simulation shown in Figure 10 within 100 steps a stable result and equilibrium are visible. The proposed model can be used to estimate the effect of adding experts during the process. To improve the dynamics of diffusion expert can be added at a selected layer. After 100th step of the simulation ten experts were added at first layer with maximal knowledge at the level of 50. It resulted in temporal growth within layer one and was followed by vertical diffusion to all layers. High increase of knowledge in a small segment of network resulted in deterioration process and reduction from average 18.45 down to 12.25 for layer number one.

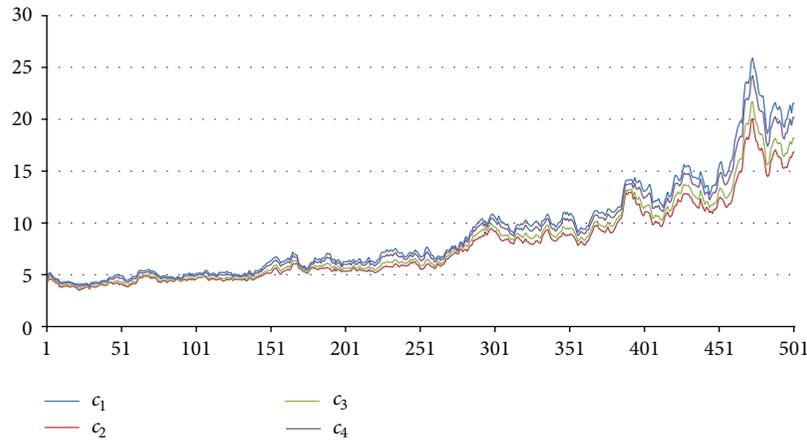


FIGURE 7: Development of competencies based on four layers.

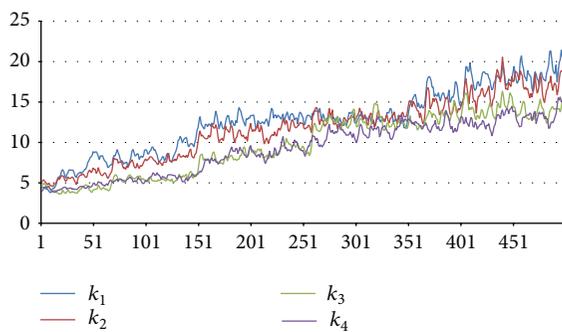


FIGURE 8: Knowledge diffusion based on asymmetric settings.

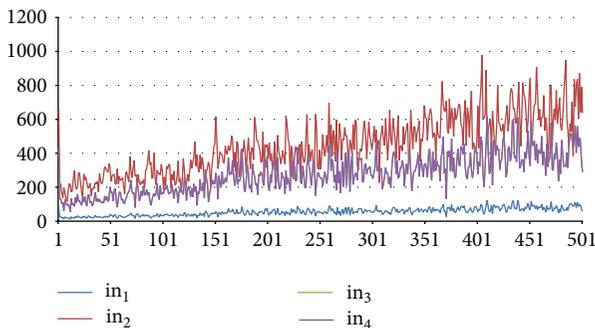


FIGURE 9: Knowledge outgoing from vertical diffusion based on asymmetric relations.

Knowledge transferred with vertical diffusion to other layers resulted in a stable increase. In the next stage, 5 experts were added with maximal knowledge at layer one at the level of 25 and it was repeated after step 300. Adding continuously experts with smaller knowledge delivered better results than one time action based on experts with knowledge much higher than average knowledge within network. Activity for incoming and outgoing knowledge is shown in Figures 11 and 12, respectively.

Using this approach it is possible to evaluate a better strategy to add a smaller number of experts with high

knowledge or add higher number of experts with smaller knowledge. In the simulated conditions adding experts with high knowledge delivered worse results because of observed deterioration process.

5.3.3. Modelling the Changes in Employment. Proposed model can be used for simulating situations of reduction of employment or job quitting. It was simulated in the next step and results are shown in Figure 13. After the 200th step of the simulation 50 random employees were removed and 40% knowledge drop was observed. Improving this situation was possible after the 300th step where 10 experts with knowledge value at 50 were added to layer one and it helped to recover average knowledge.

Even though experts were added to single layer vertical diffusion helped to recover average knowledge at layer number two. Changes in employment are resulting in different activity within incoming and outgoing knowledge at each layer which is illustrated in Figures 14 and 15.

6. Discussion and Conclusions

- (i) Competence has a dynamic nature and can be represented by the set of its states variables values. The state variables represent different pieces of a person's competence; thus the competence can be seen as a function of several time-based arguments, such as [22]: time of training (competence's acquiring), time of working actively using competence (competence's strengthening), time of inactivity (competence's decline), and time of team work/problem solving (competence's transfer). In the future, such time-relevant arguments can create another dimension for diffusion in the model.
- (ii) The model has several properties that allow managing the competence on an operational level. Current trend in the literature is that competence management can be organized according to four kinds of mutual related processes [22]: competence identification, competence assessment, competence acquisition, and

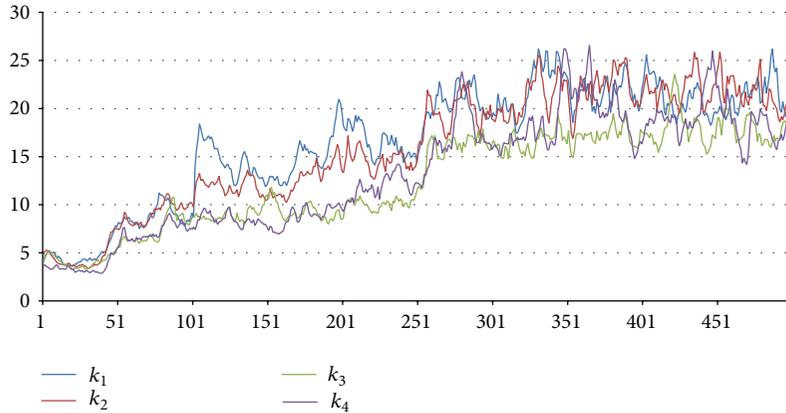


FIGURE 10: Multilayer knowledge diffusion improved by adding experts.

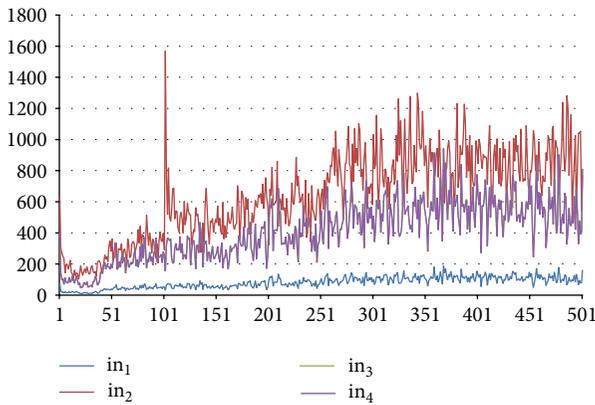


FIGURE 11: Knowledge incoming from vertical diffusion based on asymmetric relations.

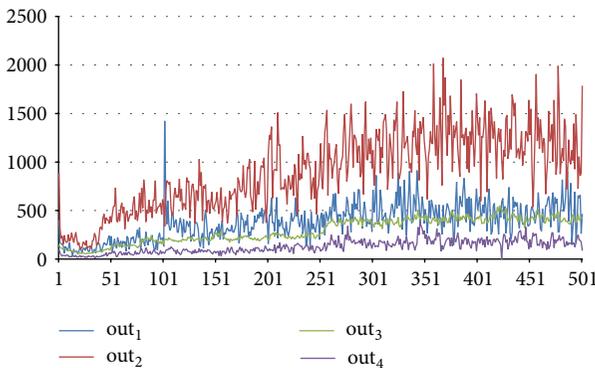


FIGURE 12: Knowledge outgoing from vertical diffusion based on asymmetric relations.

competence knowledge usage. The proposed model supported all these processes on knowledge network's level:

(a) competence identification: in order to create competence matrix all competence components

have to be identified; as a result all competence's component are recognized and measured; moreover, based on each layer analysis, the trends related to the competence are observable and can be predicted;

(b) competence assessment: due to all layers identification and description the assessment tools can be selected effectively;

(c) competence acquisition: the diffusion mechanism supports the competence acquisition and help with the selection process of employee for training;

(d) competence knowledge usage: the analysis of structure and content of network, behind the multilayer graph, give possibility to recognising the communities of practice.

(iii) The model of knowledge diffusion process, which is based on vertical and horizontal diffusion and forgetting/self-learning processes, gives a better picture of knowledge processing in an organization than the models from the literature. The proposed diffusion model allows checking what happens with competence value in an organization if, for example, the following scenario happens: base knowledge of all employees or only a few (experts) is increasing, some employees are removed from the organisation, and some employees are transferred to other parts of the organisation structure (new location in network). Moreover, we can use dedicated algorithms (e.g., [105]) to community detection in networks.

(iv) Application of the theory of the knowledge spaces allows estimating the level of knowledge in the context of existing and required competencies and relations between knowledge layers. We can precisely determine what part of worker's knowledge has to be increasing in order to achieve the required level of personal competence. The same problem for organisation required some optimisation approach. The optimisation problem is the following: how

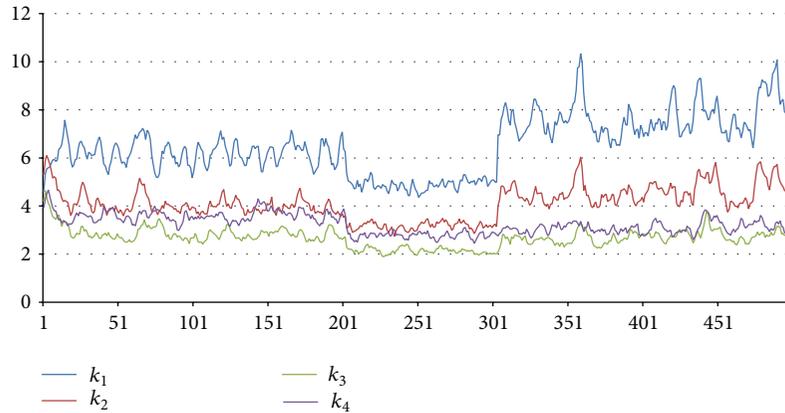


FIGURE 13: Multilayer knowledge diffusion with changes of employment.

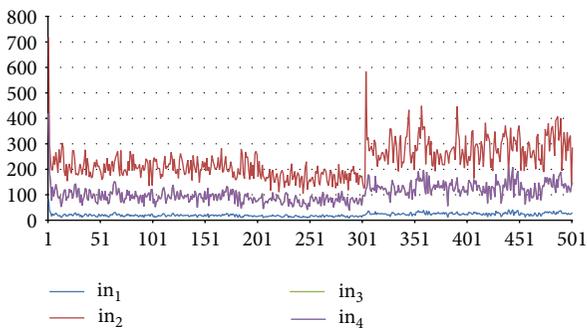


FIGURE 14: Knowledge diffusion based on asymmetric settings.

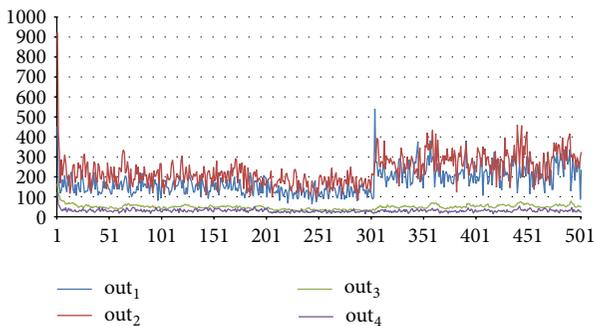


FIGURE 15: Knowledge outgoing from vertical diffusion based on asymmetric relations.

to maximise the competence level on worker or organisation level regarding to social and cognitive personal worker's abilities, knowledge distribution, and domain's structure whereas the constraints are time and cost of training?

- (v) In the presented model, only two workers' roles are distinguished regarding knowledge level: expert and normal worker. However, it is possible to recognise more roles in order to model complex organisation structure. In the work [106] four main actor roles in knowledge workers community are recognised:

the knowledge engineers (modelled the technical system part, had wide expertise in the modelling, and analysed formal semantics), the core domain experts (excellent overview of the relevant topics and players in the domain), the domain experts (professionals representing specific domain), and the application committer (works on the application level).

- (vi) In future approach to the modelling of the discussed issue it will be possible to change the relationship between the layers based on time-dependent function or semantic relations reflecting business rules. In the presented approach, there are linear relationships between the layers described by vertical diffusion matrix.
- (vii) When the network of knowledge, competence, and links are large the complexity of proposed approach is growing. The computational complexity depends on formulas for horizontal and vertical diffusion and self-learning/forgetting processes ((2)–(8)). If these formulas are nonlinear and mutually nested then the resources needed for calculations are significantly higher. The number of objects in the knowledge domain is not crucial due to formal nature of the knowledge space theory (KST). In contrast to the semantic-rooted language (like OWL) in KST all relationships can be explicit interpreted and handled based on mathematical mechanisms. Moreover, the number of connection between workers (nodes) actually does not affect the whole approach because the worker collaborated only with one other worker all the time.
- (viii) The notation of upper and lower shadow for worker's knowledge set gives opportunity to develop a cost estimation method for commencing development. The cost estimation algorithm in the form of a group competencies expansion algorithm is proposed in [107]. In this approach we have to recognise the acquired and required competence set and then based on the competence set theory the cost of competence expansion is calculated.

Conflict of Interests

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

Acknowledgments

This work was partially supported by fellowship cofinanced by the European Union within European Social Fund, by European Union's Seventh Framework Programme for research, technological development, and demonstration under Grant Agreement no. 316097 (ENGINE) and by the National Science Centre, the Decision no. DEC-2013/09/B/ST6/02317.

References

- [1] D. Ulrich, "Intellectual capital = competence x commitment," *MIT Sloan Management Review*, vol. 39, no. 2, pp. 15–26, 1998.
- [2] X. Boucher, E. Bonjour, and B. Grabot, "Formalisation and use of competencies for industrial performance optimisation: a survey," *Computers in Industry*, vol. 58, no. 2, pp. 98–117, 2007.
- [3] W. Hiermann and M. Höfferer, "A practical knowledge-based approach to skill management and personal development," *Journal of Universal Computer Science*, vol. 9, no. 12, pp. 1398–1409, 2003.
- [4] R. Mack, Y. Ravin, and R. J. Byrd, "Knowledge portals and the emerging digital knowledge workplace," *IBM Systems Journal*, vol. 40, no. 4, pp. 925–955, 2001.
- [5] K. Paulsson, T. Ivergård, and B. Hunt, "Learning at work: competence development or competence-stress," *Applied Ergonomics*, vol. 36, no. 2, pp. 135–144, 2005.
- [6] ISO, *ISO/IEC TR 24763 Information Technology—Learning, Education and Training—Conceptual Reference Model for Competency Information and Related Objects*, ISO Standard, 2011.
- [7] S. Grant and R. Young, "Concepts and standardization in areas relating to Competence," *International Journal of IT Standards and Standardization Research*, vol. 8, no. 2, pp. 29–44, 2010.
- [8] F. Draganidis and G. Mentzas, "Competency based management: a review of systems and approaches," *Information Management & Computer Security*, vol. 14, no. 1, pp. 51–64, 2006.
- [9] J. Sandberg, "Understanding of work: the basis for competence development," in *International Perspectives on Competence in the Workplace*, pp. 3–20, Springer, 2009.
- [10] M.-H. Abel, "Competencies management and learning organizational memory," *Journal of Knowledge Management*, vol. 12, no. 6, pp. 15–30, 2008.
- [11] F. W. Taylor, *The Principles of Scientific Management*, Harper-Colins and Norton, New York, NY, USA, 1911.
- [12] P. Różewski and B. Małachowski, "Competence management in knowledge-based organisation: case study based on higher education organisation," in *Proceedings of the 3rd International Conference on Knowledge Science, Engineering and Management (KSEM '09)*, pp. 358–369, December 2009.
- [13] T. Ley, A. Ulbrich, P. Scheir, S. N. Lindstaedt, B. Kump, and D. Albert, "Modeling competencies for supporting work-integrated learning in knowledge work," *Journal of Knowledge Management*, vol. 12, no. 6, pp. 31–47, 2008.
- [14] E. García-Barriocanal, M.-A. Sicilia, and S. Sánchez-Alonso, "Computing with competencies: modelling organizational capacities," *Expert Systems with Applications*, vol. 39, no. 16, pp. 12310–12318, 2012.
- [15] V. Tarasov, "Ontology-based approach to competence profile management," *Journal of Universal Computer Science*, vol. 18, no. 20, pp. 2893–2919, 2012.
- [16] R. Guillaume, R. Houé, and B. Grabot, "Robust competence assessment for job assignment," *European Journal of Operational Research*, vol. 238, no. 2, pp. 630–644, 2014.
- [17] X. Boucher, E. Bonjour, and N. Matta, "Competence management in industrial processes," *Computers in Industry*, vol. 58, no. 2, pp. 95–97, 2007.
- [18] T. Ley and D. Albert, "Skills management—managing competencies in the knowledge-based economy," *Journal of Universal Computer Science*, vol. 9, no. 12, 2003.
- [19] K. Stefanov and R. Koper, "Learning networks for lifelong competence development," *Interactive Learning Environments*, vol. 15, no. 2, pp. 101–105, 2007.
- [20] A. Frey and J. Hartig, "Assessment of competencies," *Studies in Educational Evaluation*, vol. 35, no. 2-3, pp. 55–56, 2009.
- [21] M. Lytras, M.-A. Sicilia, and A. Naeve, "Competencies management," *Journal of Knowledge Management*, vol. 12, no. 6, 2008.
- [22] P. Różewski, B. Malachowski, J. Jankowski, M. Prys, and P. Dańczura, "Preliminaries for dynamic competence management system building," in *Proceedings of the Federated Conference Computer Science and Information Systems (FedCSIS '13)*, pp. 1291–1297, Kraków, Poland, September 2013.
- [23] V. Radevski, Z. Dika, and F. Trichet, "CommOn: a framework for developing knowledge-based systems dedicated to competency-based management," in *Proceedings of the 28th International Conference on Information Technology Interfaces (ITI '06)*, pp. 419–424, June 2006.
- [24] F. Draganidis, P. Chamopoulou, and G. Mentzas, "A semantic web architecture for integrating competence management and learning paths," *Journal of Knowledge Management*, vol. 12, no. 6, pp. 121–136, 2008.
- [25] E. Wenger, R. McDermott, and W.-M. Snyder, *Cultivating Communities of Practice*, Harvard Business School Press, Boston, Mass, USA, 2002.
- [26] L. Sundberg, "A holistic approach to competence development," *Systems Research and Behavioral Science*, vol. 18, no. 2, pp. 103–114, 2001.
- [27] V. Tarasov, T. Albertsen, A. Kashevnik, K. Sandkuhl, N. Shilov, and A. Smirnov, "Ontology-based competence management for team configuration," in *Proceedings of the 3rd International Conference on Industrial Applications of Holonic and Multi-Agent Systems (HoloMAS '07)*, pp. 401–410, September 2007.
- [28] K. Koeppen, J. Hartig, E. Klieme, and D. Leutner, "Current issues in competence modeling and assessment," *The Journal of Psychology*, vol. 216, no. 2, pp. 61–73, 2008.
- [29] R. Sanchez, "Understanding competence-based management—identifying and managing five modes of competence," *Journal of Business Research*, vol. 57, no. 5, pp. 518–532, 2004.
- [30] T. Vesa, *The concept of organizational competence—a foundational analysis [Ph.D dissertations]*, University of Jyväskylä, 2004.
- [31] G. Kayakutlu and G. Büyüközkan, "Effective supply value chain based on competence success," *Supply Chain Management*, vol. 15, no. 2, pp. 129–138, 2010.
- [32] L. Bass, P. Clements, R. Kazman, and M. Klein, "Models for evaluating and improving architecture competence," Tech. Rep. CMU/SEI-2008-TR-006, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pa, USA, 2008.

- [33] J. E. H. McHenry and F. H. Strønen, "The trickiness of IT enhanced competence management," *Journal of Workplace Learning*, vol. 20, no. 2, pp. 114–132, 2008.
- [34] J. E. H. McHenry, *Management of knowledge in practice, learning to visualise competence, series of dissertations 1/2003 [Ph.D. thesis]*, Norwegian School of Management BI, Norli, 2003.
- [35] D. Sampson and D. Fytros, "Competence models in technology-enhanced competence-based learning," in *Handbook on Information Technologies for Education and Training*, International Handbooks on Information Systems, pp. 155–177, Springer, Berlin, Germany, 2nd edition, 2008.
- [36] G. LeBoterf, "Evaluer les compétences, quels jugements? quels critères? quelles instances?" *La Compétence au Travail*, vol. 135, no. 2, pp. 143–151, 1998.
- [37] Y. Jussupova-Mariethoz and A.-R. Probst, "Business concepts ontology for an enterprise performance and competences monitoring," *Computers in Industry*, vol. 58, no. 2, pp. 118–129, 2007.
- [38] A. Macris, E. Papadimitriou, and G. Vassilacopoulos, "An ontology-based competency model for workflow activity assignment policies," *Journal of Knowledge Management*, vol. 12, no. 6, pp. 72–88, 2008.
- [39] S. Colucci, T. Di Noia, E. Di Sciascio, F. M. Donini, M. Mongiello, and M. Mottola, "A formal approach to ontology-based semantic match of skills descriptions," *Journal of Universal Computer Science*, vol. 9, no. 12, pp. 1437–1454, 2003.
- [40] E. M. Rogers, *Diffusion of Innovations*, The Free Press, 4th edition, 1995.
- [41] E. García-Barricocal, M. Á. Sicilia, and S. Sánchez-Alonso, "Social network-aware interfaces as facilitators of innovation," *Journal of Computer Science and Technology*, vol. 27, no. 6, pp. 1211–1221, 2012.
- [42] C. Chen and D. Hicks, "Tracing knowledge diffusion," *Scientometrics*, vol. 59, no. 2, pp. 199–211, 2004.
- [43] R. M. N. Galang, "Divergent diffusion: understanding the interaction between institutions, firms, networks and knowledge in the international adoption of technology," *Journal of World Business*, vol. 49, no. 4, pp. 512–521, 2014.
- [44] L. C. Ribeiro, R. M. Ruiz, E. da Motta E Albuquerque, and A. T. Bernardes, "The diffusion of technological knowledge through interlaced networks," *Computer Physics Communications*, vol. 182, no. 9, pp. 1875–1878, 2011.
- [45] Z. Li, Z. Tao, and W. Lai, "A study on the knowledge diffusion of communities of practice based on the weighted small-world network," *Journal of Computers*, vol. 5, no. 7, pp. 1046–1053, 2010.
- [46] P. Morone and R. Taylor, "Knowledge diffusion dynamics and network properties of face-to-face interactions," *Journal of Evolutionary Economics*, vol. 14, no. 3, pp. 327–351, 2004.
- [47] S. Koohborfardhaghghi and J. Altmann, "How structural changes in complex networks impact organizational learning performance," in *Proceedings of the 6th International Workshop on Emergent Intelligence on Networked Agents (WEIN '14)*, pp. 1017–1022, March 2014.
- [48] H.-C. Huang, H.-Y. Shih, and S.-C. Hsu, "Team structure to accelerate knowledge diffusion: a case study in computer software developer," in *Proceedings of the 5th IEEE International Conference on Management of Innovation and Technology (ICMIT '10)*, pp. 928–933, June 2010.
- [49] G. Ehrhardt, M. Marsili, and F. Vega-Redondo, "Diffusion and growth in an evolving network," *International Journal of Game Theory*, vol. 34, no. 3, pp. 383–397, 2006.
- [50] C. Hui, M. Goldberg, M. Magdon-Ismail, and W. A. Wallace, "Simulating the diffusion of information: an agent-based modeling approach," *International Journal of Agent Technologies and Systems*, vol. 2, no. 3, pp. 31–46, 2010.
- [51] K. Dasgupta, "Learning and knowledge diffusion in a global economy," *Journal of International Economics*, vol. 87, no. 2, pp. 323–336, 2012.
- [52] V. B. Kreng and C. M. Tsai, "The construct and application of knowledge diffusion model," *Expert Systems with Applications*, vol. 25, no. 2, pp. 177–186, 2003.
- [53] R. A. Brands, "Cognitive social structures in social network research: a review," *Journal of Organizational Behavior*, vol. 34, no. 1, pp. S82–S103, 2013.
- [54] B. Golub and M. O. Jackson, "How homophily affects the speed of learning and best-response dynamics," *The Quarterly Journal of Economics*, vol. 127, no. 3, pp. 1287–1338, 2012.
- [55] M. MacGarvie, "The determinants of international knowledge diffusion as measured by patent citations," *Economics Letters*, vol. 87, no. 1, pp. 121–126, 2005.
- [56] A. Canals, "Knowledge diffusion and complex networks: a model of high-tech geographical industrial clusters," in *Proceedings of the 6th European Conference on Organizational Knowledge, Learning, and Capabilities*, pp. 1–25, March 2005.
- [57] M. Kehal, "Knowledge diffusion in a specialist organization: observational and data-driven case study," in *Smart Innovation, Systems and Technologies*, vol. 9, pp. 189–200, Springer, 2011.
- [58] C. M. Tsai, "The knowledge diffusion model associated with innovative knowledge," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11323–11331, 2009.
- [59] Y. Jing, "Tacit knowledge diffusion model in project and its critic factors based on ecology," in *Proceedings of the IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS '09)*, pp. 198–202, Shanghai, China, November 2009.
- [60] P. Yan and X. Yang, "To trade or to teach: modeling tacit knowledge diffusion in complex social networks," in *Proceedings of the 2nd International Conference on Future Information Technology and Management Engineering (FITME '09)*, pp. 151–154, IEEE, Sanya, China, December 2009.
- [61] B. J. Loasby, "Time, knowledge and evolutionary dynamics: why connections matter," *Journal of Evolutionary Economics*, vol. 11, no. 4, pp. 393–412, 2001.
- [62] J.-P. Cointet and C. Roth, "How realistic should knowledge diffusion models be?" *Journal of Artificial Societies and Social Simulation*, vol. 10, no. 3, 2007.
- [63] S. Morris, "Contagion," *Review of Economic Studies*, vol. 67, no. 1, pp. 57–78, 2000.
- [64] F. Deroian, "Formation of social networks and diffusion of innovations," *Research Policy*, vol. 31, no. 5, pp. 835–846, 2002.
- [65] K. Klemm, V. M. Eguíluz, R. Toral, and M. san Miguel, "Globalization, polarization and cultural drift," *Journal of Economic Dynamics and Control*, vol. 29, no. 1-2, pp. 321–334, 2005.
- [66] J. Leskovec, L. A. Adamic, and B. A. Huberman, "The dynamics of viral marketing," *ACM Transactions on the Web*, vol. 1, no. 1, article 5, 2007.
- [67] M.-Á. Sicilia, M. Lytras, E. Rodríguez, and E. García-Barricocal, "Integrating descriptions of knowledge management learning activities into large ontological structures: a case study," *Data & Knowledge Engineering*, vol. 57, no. 2, pp. 111–121, 2006.

- [68] I. García, C. Benavides, H. Alaiz, and A. Alonso, "A study of the use of ontologies for building computer-aided control engineering self-learning educational software," *Journal of Science Education and Technology*, vol. 22, no. 4, pp. 589–601, 2013.
- [69] A. Naeve, M.-A. Sicilia, and M. D. Lytras, "Learning processes and processing learning: from organizational needs to learning designs," *Journal of Knowledge Management*, vol. 12, no. 6, pp. 5–14, 2008.
- [70] R. Cowan and N. Jonard, "Network structure and the diffusion of knowledge," *Journal of Economic Dynamics & Control*, vol. 28, no. 8, pp. 1557–1575, 2004.
- [71] P. Morone and R. Taylor, *Knowledge Diffusion and Innovation: Modelling Complex Entrepreneurial Behaviours*, Cheltenham Edward Elgar Publishing, Cheltenham, UK, 2010.
- [72] X. Geng and R. Mao, "Study on knowledge diffusion in organization: a mathematical model," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '08)*, pp. 1–4, October 2008.
- [73] E. Zhuang, G. Chen, and G. Feng, "A network model of knowledge accumulation through diffusion and upgrade," *Physica A: Statistical Mechanics and Its Applications*, vol. 390, no. 13, pp. 2582–2592, 2011.
- [74] R. Cowan and N. Jonard, "Knowledge creation, knowledge diffusion and network structure," in *Economics with Heterogeneous Interacting Agents*, vol. 503 of *Lecture Notes in Economics and Mathematical Systems*, pp. 327–343, Springer, 2001.
- [75] A. Morone, P. Morone, and R. Taylor, "A laboratory experiment of knowledge diffusion dynamics," *Innovation, Industrial Dynamics and Structural Transformation: Schumpeterian Legacies*, pp. 283–302, 2007.
- [76] R. I. Hickson, S. I. Barry, G. N. Mercer, and H. S. Sidhu, "Finite difference schemes for multilayer diffusion," *Mathematical and Computer Modelling*, vol. 54, no. 1-2, pp. 210–220, 2011.
- [77] R. Michalski, P. Kazienko, and J. Jankowski, "Convince a dozen more and succeed—the influence in multi-layered social networks," in *Proceedings of the 9th International Conference on Signal-Image Technology and Internet-Based Systems (SITIS '13)*, pp. 499–505, IEEE, Kyoto, Japan, December 2013.
- [78] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.
- [79] A. Gomez-Perez, O. Corcho, and M. Fernandez-Lopez, *Ontological Engineering: With Examples from the Areas of Knowledge Management, e-Commerce and the Semantic Web*, Springer, London, UK, 2004.
- [80] J.-P. Doignon and J.-C. Falmagne, *Learning Spaces: Interdisciplinary Applied Mathematics*, Springer, Berlin, Germany, 2011.
- [81] P. Różewski, "Model of intangible production network for competence development," in *Knowledge Science, Engineering and Management: 5th International Conference, KSEM 2011, Irvine, CA, USA, December 12–14, 2011. Proceedings*, vol. 7091 of *Lecture Notes in Computer Science*, pp. 246–256, Springer, Berlin, Germany, 2011.
- [82] S. Brandt, D. Albert, and C. Hockemeyer, "Surmise relations between tests—mathematical considerations," *Discrete Applied Mathematics*, vol. 127, no. 2, pp. 221–239, 2003.
- [83] C. Hockemeyer, O. Conlan, V. Wade, and D. Albert, "Applying competence prerequisite structures for eLearning and skill management," *Journal of Universal Computer Science*, vol. 9, no. 12, pp. 1428–1436, 2003.
- [84] L. Hu, X. Qin, and Y. Hu, "A capability assignment concept model for resource management system of product development," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE '07)*, pp. 1927–1932, June 2007.
- [85] J. Heller, B. Mayer, C. Hockemeyer, and D. Albert, "Competence-based knowledge structures for personalised learning," *International Journal on E-Learning*, vol. 5, no. 1, pp. 75–88, 2006.
- [86] H. Cho, G. Gay, B. Davidson, and A. Ingraffea, "Social networks, communication styles, and learning performance in a CSCL community," *Computers & Education*, vol. 49, no. 2, pp. 309–329, 2007.
- [87] D. Xu, H. Wang, and M. Wang, "A conceptual model of personalized virtual learning environments," *Expert Systems with Applications*, vol. 29, no. 3, pp. 525–534, 2005.
- [88] T. Farrington-Darby and J. R. Wilson, "The nature of expertise: a review," *Applied Ergonomics*, vol. 37, no. 1, pp. 17–32, 2006.
- [89] J. R. Anderson, *Cognitive Psychology and Its Implications*, Worth Publishing, New York, NY, USA, 5th edition, 2000.
- [90] H. Nonaka Takeuchi, *The Knowledge Creating Company*, Oxford University Press, Oxford, UK, 1995.
- [91] A. D. Marwick, "Knowledge management technology," *IBM Systems Journal*, vol. 40, no. 4, pp. 814–830, 2001.
- [92] Y. Zhang and Y. Zhou, "Knowledge forgetting: properties and applications," *Artificial Intelligence*, vol. 173, no. 16-17, pp. 1525–1537, 2009.
- [93] P. M. de Holan and N. Phillips, "Remembrance of things past? The dynamics of organizational forgetting," *Management Science*, vol. 50, no. 11, pp. 1603–1613, 2004.
- [94] P. M. de Holan, N. Phillips, and L. Thomas, "Managing organizational forgetting," *MIT Sloan Management Review*, vol. 45, no. 2, pp. 45–51, 2004.
- [95] P. Różewski, E. Kusztna, R. Tadeusiewicz, and O. Zaikin, *Intelligent Open Learning Systems: Concepts, Models and Algorithms*, vol. 22 of *Intelligent Systems Reference Library*, Springer, Berlin, Germany, 2011.
- [96] U. Wilensky, *NetLogo*, Center for Connected Learning and Computer-Based Modeling, Northwestern University, 1999, <http://ccl.northwestern.edu/netlogo/>.
- [97] P. Różewski, "Discussion of the competence management models for education context," in *Proceedings of the 6th International KES Conference on Agents and Multi-Agent Systems-Technologies and Applications (KES-AMSTA '12)*, pp. 604–613, June 2012.
- [98] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, pp. 47–97, 2002.
- [99] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.
- [100] S. Qu, S. Li, and W. Tian, "Research on knowledge diffusion model based on organizational learning network," in *Proceedings of the Chinese Control and Decision Conference (CCDC '10)*, pp. 2556–2560, May 2010.
- [101] L. Crawford, "Senior management perceptions of project management competence," *International Journal of Project Management*, vol. 23, no. 1, pp. 7–16, 2005.
- [102] H. Wi, S. Oh, J. Mun, and M. Jung, "A team formation model based on knowledge and collaboration," *Expert Systems with Applications*, vol. 36, no. 5, pp. 9121–9134, 2009.
- [103] M. Harzallah and F. Vernadat, "IT-based competency modeling and management: from theory to practice in enterprise engineering and operations," *Computers in Industry*, vol. 48, no. 2, pp. 157–179, 2002.

- [104] R. R. Hoffman, N. R. Shadbolt, A. M. Burton, and G. Klein, "Eliciting Knowledge from Experts: a Methodological Analysis," *Organizational Behavior and Human Decision Processes*, vol. 62, no. 2, pp. 129–158, 1995.
- [105] M. Zhu, F. Meng, and Y. Zhou, "Semisupervised clustering for networks based on fast affinity propagation," *Mathematical Problems in Engineering*, vol. 2013, Article ID 385265, 13 pages, 2013.
- [106] P. De Leenheer, S. Christiaens, and R. Meersman, "Business semantics management: a case study for competency-centric HRM," *Computers in Industry*, vol. 61, no. 8, pp. 760–775, 2010.
- [107] E. Kushtina, O. Zaikin, P. Rózewski, and B. Małachowski, "Cost estimation algorithm and decision-making model for curriculum modification in educational organization," *European Journal of Operational Research*, vol. 197, no. 2, pp. 752–763, 2009.

Research Article

Optimization of Train Trip Package Operation Scheme

Lu Tong, Lei Nie, Zhenhuan He, and Huiling Fu

School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Lu Tong; tonglu_bjtt@163.com

Received 25 September 2014; Accepted 12 January 2015

Academic Editor: Orwa J. Houshia

Copyright © 2015 Lu Tong 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.

Train trip package transportation is an advanced form of railway freight transportation, realized by a specialized train which has fixed stations, fixed time, and fixed path. Train trip package transportation has lots of advantages, such as large volume, long distance, high speed, simple forms of organization, and high margin, so it has become the main way of railway freight transportation. This paper firstly analyzes the related factors of train trip package transportation from its organizational forms and characteristics. Then an optimization model for train trip package transportation is established to provide optimum operation schemes. The proposed model is solved by the genetic algorithm. At last, the paper tests the model on the basis of the data of 8 regions. The results show that the proposed method is feasible for solving operation scheme issues of train trip package.

1. Introduction

Railway transportation is one of the main types of modern transportation. Railway transportation has the features of fast speed, low costs, environmental friendship, high reliability, accuracy, and continuity. Before the completion of high-speed railway network, railway transportation in China heavily focused on passenger transport. For this reason, railway freight transportation was always restricted by the problem of insufficient capacity. With the operation of motor trains and high-speed railway in recent years, a large number of railway capacities have been released and applied to freight transportation. As the main form of railway freight transportation, train trip package transportation is becoming increasingly important. Meanwhile, it is also becoming more complex due to increased railway transport resources.

Train trip package transportation provides luggage and parcel transportation, fast freight, and joint logistics services. Train trip package transportation is based on the existing railway network and uses passenger baggage cars, train trip packages, and mail train lines as the carrier. With the rapid development of social economy and the growing requirement of perfecting transportation service, railway package transportation has been improving its level of service and has become the main body of rapid railway freight transportation.

There are two main types of railway freight transportation: one is the package marshaled in the passenger train; the other is transport by a specialized train which consists of a certain number of baggage compartments and has fixed stations, fixed time, and fixed path. The second type is normally called the train trip package transportation. In the near future, train trip package transportation will become the main form of package transportation. Therefore, a reasonable train trip package operation scheme can help to improve the level of train trip package organization and the quality of service.

Most of the current researches have focused on the passenger train operation scheme, but few focused on train trip package operation scheme [1–10]. Crainic [11] summarized the transportation services network issues. Barnhart et al. [12] regarded railway marshaling issues as network design problems, where the network nodes represented marshaling yard, and the arc segments represented marshaling train. Chang et al. [13] used fuzzy mathematical programming to study Taiwan high-speed railway train operation plan. Crainic and Rousseau [14] used the heuristic algorithm based on decomposition method and the column generation method to solve railway freight service network design problem with no capacity limits. Bussieck et al. [15] discussed the operation plan optimization problem with periodic timetable. Yu et al. [16] proposed a transit operation for passengers

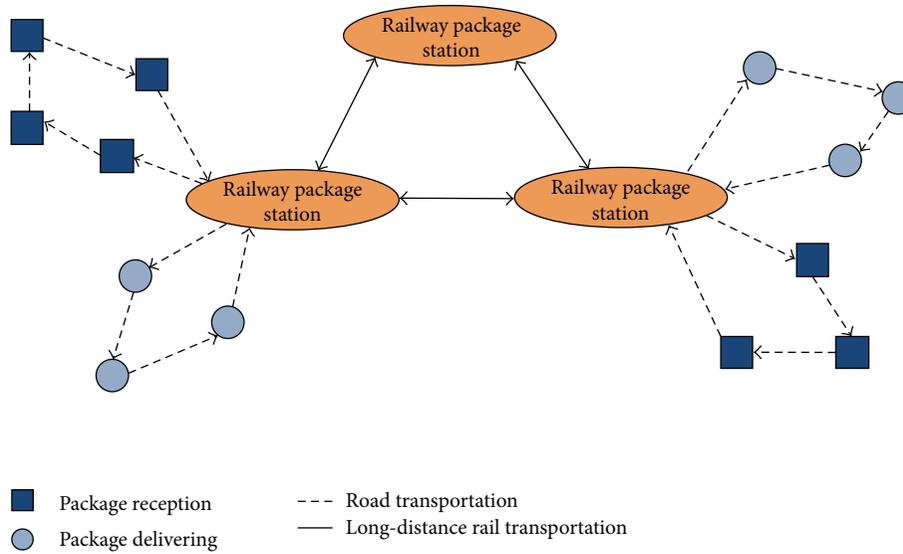


FIGURE 1: The whole process of package transportation.

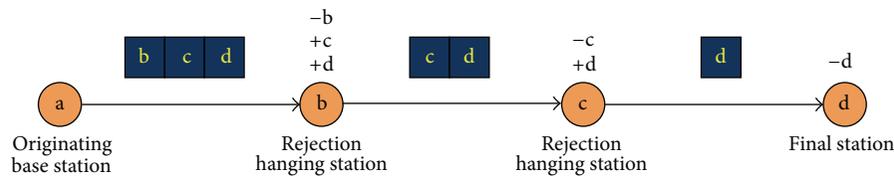


FIGURE 2: The operation process of grouped trains.

transportation, which considered passenger attraction from candidate nodes to destination nodes. It is a novelty method to the best of our knowledge. Newton et al. [17] established a path-based model without considering the associated fixed costs of trains but adding capacity constraints to restrict the number of trains which could be marshaled and the freight amount which could be handled. Due to some similar features of “passenger train” and train trip package, the train trip package operation scheme optimization can refer to existing research of passenger train operation scheme.

In this paper, the existing research results and experiences of passenger train operation scheme are studied. Then the related factors of train trip package operation scheme are analyzed from its organizational characteristics. A mathematical model is developed to optimize the train trip package operation scheme for practical situation. At the end, the proposed model is verified through an example, and some conclusions are obtained from the experiment.

2. Problem Description

The whole process of package transportation can be divided into three operational phases: reception and delivery in the stations of two ends and long-distance rail transportation between the stations. The whole process of package transportation and the relationship among the operational phases

are shown in Figure 1. In this paper, we only consider long-distance railway transport between stations and develop a reasonable operation scheme.

In train trip package transportation, the train marshaling is composed of package trains with the same final station (single set of trains) or package trains whose final stations are in the same running path (grouped trains). The operation process of grouped trains is shown in Figure 2. Grouped trains start from the originating base station and will gradually turn into a single set of trains after rejection and hanging for once or several times along the way. When grouped trains reach the rejection and hanging stations before the final station, they are required for rejection and hanging operations.

3. Model

3.1. Model Assumption. Through the analysis of organizational characteristics of train trip package transportation, we first make the following assumptions before establishing the optimization model of train trip package operation scheme.

- (1) The path of train trip package has been already predetermined.
- (2) The transport capacity of train trip package is not restricted and the ability to receive and send trains of

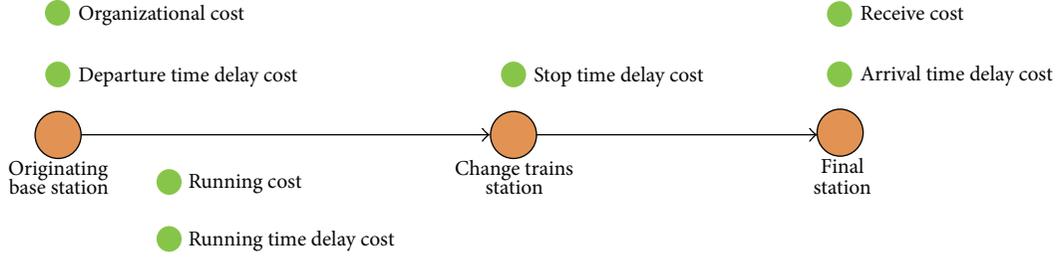


FIGURE 3: Expenses in the service process of train trip package.

each package station can satisfy any traffic intensity; each path can satisfy any traffic intensity.

- (3) Operation details and operation process inside stations are not considered, because the operation of train trip package only reflects the influence on the operation scheme in some important links, such as loading costs, rejection, and hanging operations costs.
- (4) The operation of train trip package depends on the traffic situation and does not consider the constraints of facing operation.

3.2. Parameter Setting

C represents the cost per vehicle kilometer in train trip package transportation;

C_0 is the rejection and hanging costs for a train;

P_{ij} is the train flow leading from region i to region j ;

A_{rs} represents the collection of links that are passed by train trip package between regions r and s ;

L_{rs} is the transport distance between regions r and s ;

x_{ij}^{rs} represents the train flow between regions r and s which are in the path from region i to region j ;

α_{ij}^{rs} is 0-1 variable, if the railway freight flows between regions r and s are carried by train trip packages from region i to region j , $\alpha_{ij}^{rs} = 1$; otherwise, $\alpha_{ij}^{rs} = 0$;

μ_0 represents the number of trains that can be marshaled in a set of grouped trains;

n_{rs} is the number of train trip packages between regions r and s ;

M represents infinity.

3.3. Modeling. All the expenses in the service process of train trip package are shown in Figure 3.

3.3.1. Loading Cost. Train trip package transport is a kind of contractual transport and the contractors are obligated to organize shipments. In principle, the contractors of train trip package need to load and unload packages by themselves. Therefore, the cost of train trip package in loading place is considered as a constant which is denoted by the average loading cost.

3.3.2. Rejection and Hanging Cost. A single set of trains do not need rejection and hanging operations in the middle station. Grouped trains need rejection and hanging operations in the operation stations along the way but do not need loading, unloading, and modifying marshaling operations. Therefore, the rejection and hanging cost only existed in the grouped trains and can be considered as a fixed value.

In the model established in this paper, the total cost of train trip package transportation consists of two parts: one is in-transit cost such as locomotive traction cost and line usage cost which are decided by the cost per vehicle kilometer; the other is the rejection and hanging cost which is decided by the rejection and hanging costs for a train. Because the loading cost of train trip package can be considered as fixed, loading cost is not reflected in the model.

The optimization model of train trip package operation is

$$\begin{aligned} \text{Min } Z = & \sum_{ij} \sum_{rs} \sum_{k \in A_{rs} \cap A_{ij}} CL_k x_{ij}^{rs} \\ & + C_0 \sum_{ij} \left(\sum_{rs} x_{ij}^{rs} - x_{ij}^{ij} \right), \end{aligned} \quad (1)$$

$$\text{s.t. } \sum_{ij} x_{ij}^{rs} = P_{rs} \quad \forall r, s \in S, \quad (2)$$

$$x_{ij}^{rs} - \alpha_{ij}^{rs} \cdot M \leq 0 \quad \forall r, s, i, j \in S, \quad (3)$$

$$\sum_{rs} x_{ij}^{rs} \leq n_{ij} \mu_0 \quad \forall i, j \in S, \quad (4)$$

$$x_{ij}^{rs} \geq 0 \quad \forall r, s, i, j \in S, \quad (5)$$

$$n_{rs} \geq 0, \quad n_{rs} \text{ is integer.} \quad (6)$$

The purpose of objective function (1) is to minimize the total operation cost of all train trip packages.

Constraint (2) is the flow conservation constraint, ensuring that train trip package operation scheme meets the transport demand between each OD pair.

Constraint (3) ensures that the railway freight flows carried by train trip packages are assigned to feasible path.

Constraint (4) ensures that the number of train flows in each service interval does not exceed the carrying capacity the train service can provide.

Constraints (5) and (6) are nonnegative constraints of the decision variables and constraint (6) requires the number of train trip packages to be integers.

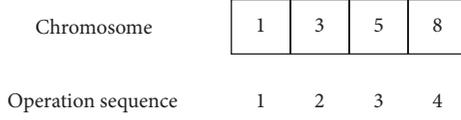


FIGURE 4: Representation of chromosome.

3.4. Genetic Algorithm. Because the formulation of train trip package operation scheme is a linear discrete optimization problem, the traditional method is difficult to get the optimal solution to the problem. Many literatures suggested that heuristic algorithm was often the first choice to solve this kind of complicated transportation optimization problems [18–21]. Among heuristic algorithms, genetic algorithm (GA) is a probabilistic search algorithm for global optimization. The genetic algorithm is formed by simulating biological genetic and evolutionary processes in the natural world. Genetic algorithm has good global search capability and fast calculation speed. It can quickly find an optimal solution from the entire solution space and will seldom fall into the local optimal solution. Currently, genetic algorithm has achieved good application in the path optimization, site selection, and other aspects which has been applied to solve lots of complicated problems [22–25]. Therefore, the genetic algorithm is used in this paper to solve the problem.

Step 1 (coding). The decimal coding is employed to generate an array of numbers, namely, a chromosome. Each chromosome represents an operation sequence of train trip package (Figure 4). A gene in a chromosome represents a station that train trip packages pass by; for example, 1–3–5–8 represents that train trip packages start from station 1 and finally get to station 8, passing by station 3 and station 5.

Step 2 (generating initial populations). According to OD flows between package stations, we successively select each OD pair and find a path between stations to produce a feasible chromosome. Repeat the process until feasible chromosomes between each OD pair have been produced.

Step 3 (calculation of fitness value). Clearly, some chromosomes may not satisfy all of the constraints in the model. Therefore, it is necessary to construct the fitness function. Adaptation function indicates the superiority of chromosomes. For the minimum value problem in this paper, the fitness function is

$$F(x') = N - \sum_{ij} \sum_{rs} \sum_{k \in A_{rs} \cap A_{ij}} CL_k x_{ij}^{rs} - C_0 \sum_{ij} \left(\sum_{rs} x_{ij}^{rs} - x_{ij}^{ij} \right). \quad (7)$$

In the formula, the value of N will be determined according to the size of the problem, to ensure $F(x') \geq 0$.

Step 4 (selection and crossover (Figure 5)). Roulette method is used as a selection mechanism. In the roulette method,

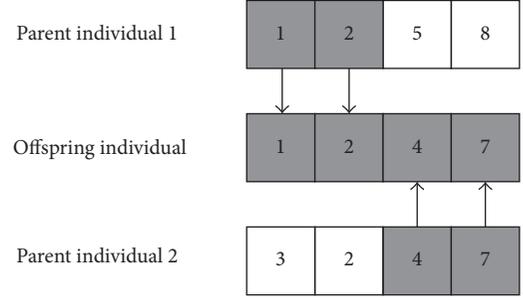


FIGURE 5: Crossover operator of genetic algorithm.

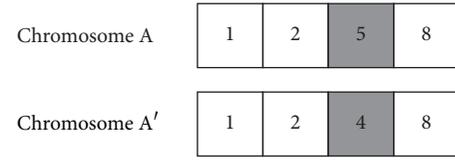


FIGURE 6: Mutation operator of genetic algorithm.

chromosomes will be selected for reproducing a new generation with certain probability. The selection probability of a chromosome is related to its fitness. Generally, the higher the fitness value, the bigger the selection probability. Meanwhile, the elite strategy is also implemented, which means the chromosome with the maximum fitness value in each generation of populations will be copied to the next generation. The optimal policy makes the next generation not worse than the parent generation.

Crossover introduces random changes to the selected chromosomes by crossing two parent individuals to produce offspring individuals with a user-specified probability p_c called crossover rate. In this paper, the single-point crossover is implemented.

Step 5 (mutation). Mutation introduces random changes to the chromosomes by altering the value of a gene with a user-specified probability p_m called mutation rate (Figure 6).

Step 6 (stopping criterion). To compare the computation efficiency of different algorithms, different stopping times are set as the stopping criteria. When getting to the predetermined stopping time, the algorithm ends.

4. Numerical Test

Based on “three vertical and four horizontal” train transportation physical network of China, eight package stations are chosen to build a spatial network graph of package transportation regions (as shown in Figure 7); then design a service network of train trip package. The purpose is to determine the organization form, the running paths, and the operation number of trains.

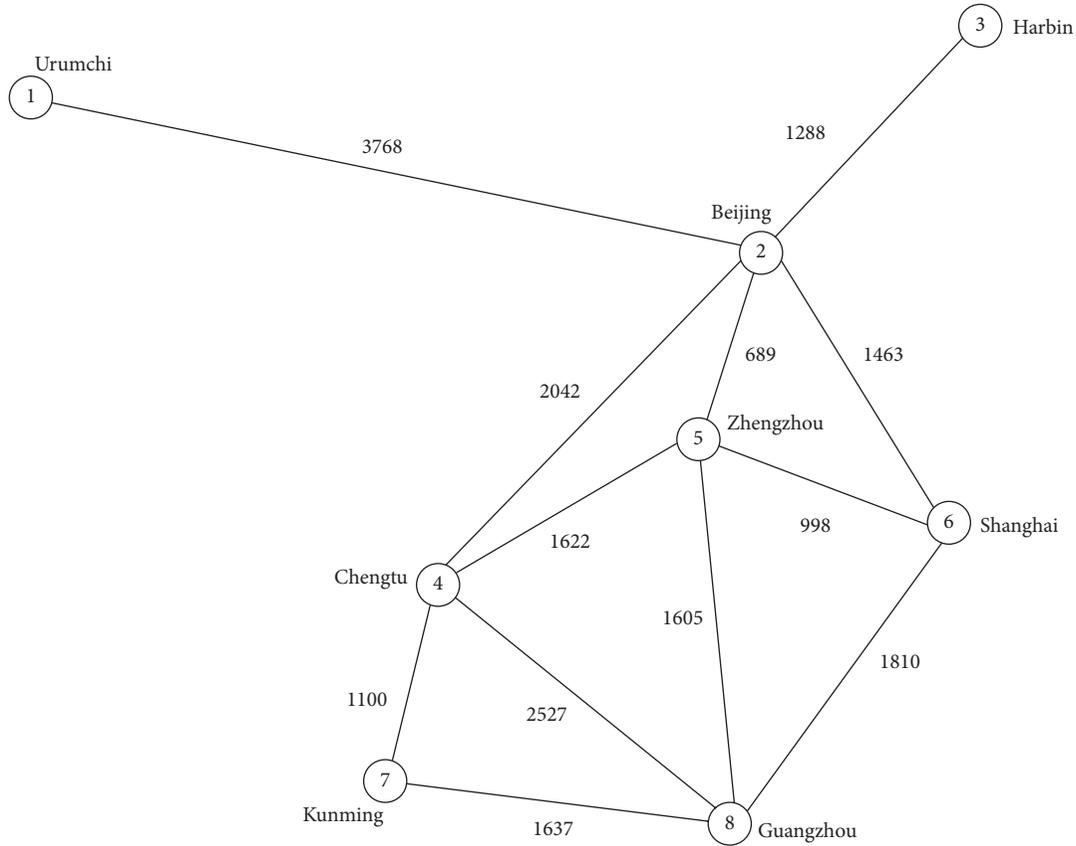


FIGURE 7: Spatial network graph of package transportation regions.

TABLE 1: Space distance of package transportation regions (km).

Region	1	2	3	4	5	6	7	8
1	—	3768	—	—	—	—	—	—
2		—	1288	2042	689	1463	—	—
3			—	—	—	—	—	—
4				—	1622	—	1100	2527
5					—	998	—	1605
6						—	—	1810
7							—	1637
8								—

4.1. Data Preparation

4.1.1. Data of Interregional Distance. As the spatial network graph of package transportation regions shown in Figure 7, the actual length of the links between each node pair can be considered as the distance between center cities of each region. Specific distance data are shown in Table 1.

4.1.2. Data of OD Flows. According to the statistics of China railway transportation, Table 2 shows OD flows by package between 8 package regions, in units of tons per year.

In this paper, assume that the length of the design cycle is one day and the OD flows for each period are constant. It is

known that the OD flows in Table 2 are represented in units of tons per year, so a conversion coefficient is introduced here to obtain daily OD flows in units of trains/day.

Since the volume of packages is difficult to accurately grasp in the actual operation, the deadweight constraint of package trains is considered as an alternative. The deadweight capacity of a package train is averagely 15 tons according to our survey and a year is calculated as 365 days. The result of the conversion is shown in Table 3. Note that the numbers of package trains have to be integer value, so the final converted values are rounded off.

4.1.3. Cost Data. The average operating cost is set to 90 per vehicle kilometer ($C = 90$) and the rejection and hanging operating costs are set to 1000 per train ($C_0 = 1000$).

4.1.4. Other Data. According to the regulation of the number of marshaling train trip packages, $\mu_0 = 20$.

4.2. The Results. Based on the above model parameters, the genetic algorithm is used to calculate model through C++ programming. The crossover probability of GA is set as 0.8 and the mutation probability is set as 0.05. The maximum evolution generation of GA is set as 100. The specific results are shown in Table 4.

TABLE 2: OD flows by package among 8 package regions (tons/year).

Region	1	2	3	4	5	6	7	8
1	—	139790	8284	34410	20431	80648	10753	44088
2	50294	—	62909	136105	57034	140123	67275	108884
3	7710	96678	—	—	—	32028	—	15421
4	88962	172889	—	—	120854	119176	186317	100712
5	13255	46392	—	26510	—	64381	—	31954
6	44764	135649	42155	121667	104032	—	59998	109145
7	20523	79930	—	106933	—	74529	—	73449
8	31014	180416	39662	96321	79025	188766	65606	—

TABLE 3: Daily OD train flows among 8 package regions (train/day).

Region	1	2	3	4	5	6	7	8
1	—	26	2	6	4	15	2	8
2	9	—	11	25	10	26	12	20
3	1	18	—	—	—	6	—	3
4	16	32	—	—	22	22	34	18
5	2	8	—	5	—	12	—	6
6	8	25	8	22	19	—	11	20
7	4	15	—	20	—	14	—	13
8	6	33	7	18	14	34	12	—

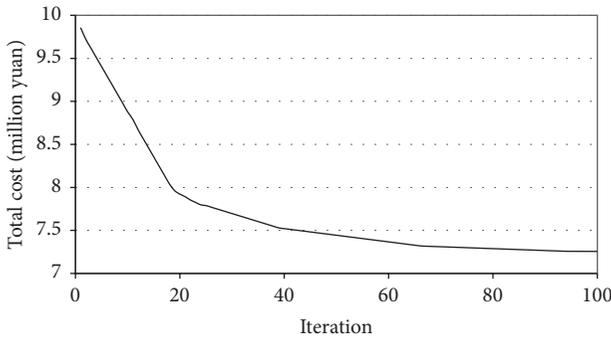


FIGURE 8: Result of each calculation.

As shown in Table 4, parts of OD flows between stations are transported directly from the start station to the final station, without any transit operations. The other parts of OD flows are required to stop at middle stations for rejection and hanging.

4.3. Test Results Discussion. Figure 8 depicts the convergence of the calculation. The results of computational tests show that the algorithm of this paper can achieve convergence within 100 generations, indicating a good convergence and a good adaptability to solve the proposed model.

It is known that the initial operation scheme consists of 48 single sets of trains. All OD flows are transported directly from the start station to the final station, without any operation in transit. Target cost of initial operation scheme is 9.8574 million yuan.

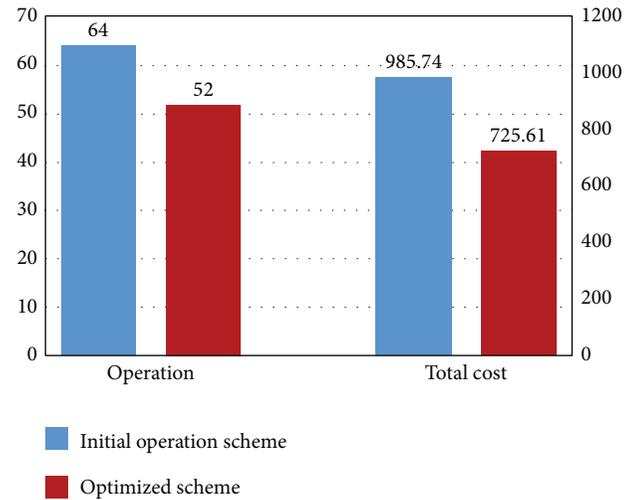


FIGURE 9: Comparison of the results.

After optimization, the sample train transport service network consists of 26 kinds of train trip package transport services. In the optimized operation scheme, there are 52 operations within a day, including 45 operations of single set of trains and 7 of grouped trains. Compared with the initial scheme, the optimized scheme reduces the operation number of single set of trains in long distance and yet increases the operation number of both single set of trains and group trains in short/middle distance, thus reducing the total cost of train transportation. The total cost of the optimized scheme is 7.2561 million yuan, reduced by 26.4% compared with the initial one.

To evaluate the performance of the proposed scheme, the results of initial operation scheme and optimized scheme can be showed in Figure 9. It can be found that the optimized scheme can greatly reduce the total cost. Thus, the proposed optimized scheme is an effective method for train trip package optimization.

5. Conclusions

This paper focuses on how to arrange the train trip package operation scheme to lower the transportation cost. An optimization model is established based on the analysis of

TABLE 4: Calculation results train trip package operation scheme.

Number	Originating station	Final station	Running path	The number of train flows	Operation number	Organizational form
1	1	2	1 → 2	27	2	Single
2	1	4	1 → 4	12	1	Single
3	1	5	1 → 5	8	1	Single
4	2	1	2 → 1	11	1	Single
5	2	3	2 → 3	28	2	Single
6	2	4	2 → 5 → 4	50	3	Single
7	2	6	2 → 6	31	2	Single
8	2	8	2 → 5 → 8	33	2	Grouped
9	3	1	3 → 2 → 1	28	2	Single
10	4	1	4 → 1	16	1	Single
11	4	2	4 → 2	32	2	Grouped
12	4	5	4 → 5	32	2	Single
13	4	6	4 → 5 → 6	48	3	Grouped
14	4	7	4 → 7	48	3	Single
15	4	8	4 → 8	22	2	Single
16	5	1	5 → 1	11	1	Single
17	6	2	6 → 2	32	2	Single
18	6	4	6 → 5 → 4	35	2	Single
19	6	5	6 → 5	19	2	Single
20	6	8	6 → 8	31	2	Single
21	7	4	7 → 4	23	2	Single
22	7	8	7 → 8	27	2	Single
23	8	2	8 → 5 → 2	63	4	Single
24	8	4	8 → 4	18	1	Single
25	8	6	8 → 6	48	3	Single
26	8	7	8 → 7	23	2	Single

the organizational forms and characteristics of the train trip package transport, as well as the existing research results and experiences of passenger train plan. Computational tests show that the proposed model can effectively reduce the cost of transportation. The model can also optimize the operation number of train trip packages in the experiment, indicating that it has better applicability.

An issue of future research is the consideration of more constraints, such as the inconsistent loading cost between single set of trains and group trains and the constraints of facing operation. In further studies, these constraints will be considered in order to get closer to the actual situation.

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 Railway Ministry Science and Technology Management Project (2013X014-C), China Railway Corporation Project (2014X010-A), and Fundamental Research Funds for the Central Universities (Beijing Jiaotong University) under Grant no. 2014JBZ008.

References

- [1] A. A. Assad, "Models for rail transportation," *Transportation Research Part A: General*, vol. 14, no. 3, pp. 205–220, 1980.
- [2] A. A. Assad, "Modelling of rail networks: toward a routing/makeup model," *Transportation Research Part B*, vol. 14, no. 1–2, pp. 101–114, 1980.
- [3] C. Barnhart, N. Krishnan, D. Kim, and K. Ware, "Network design for express shipment delivery," *Computational Optimization and Applications*, vol. 21, no. 3, pp. 239–262, 2002.
- [4] J. M. Farvolden and W. B. Powell, "Subgradient methods for the service network design problem," *Transportation Science*, vol. 28, no. 3, pp. 256–272, 1994.
- [5] Z. Gao, J. Wu, and H. Sun, "Solution algorithm for the bi-level discrete network design problem," *Transportation Research Part B: Methodological*, vol. 39, no. 6, pp. 479–495, 2005.
- [6] H.-Q. Peng and Y.-J. Zhu, "Intercity train operation schemes based on passenger flow dynamic assignment," *Journal of Transportation Systems Engineering and Information Technology*, vol. 13, no. 1, pp. 111–117, 2013.
- [7] M. H. Keaton, "Designing railroad operating plans: a dual adjustment method for implementing lagrangian relaxation," *Transportation Science*, vol. 26, no. 4, pp. 263–279, 1992.
- [8] M. Yaghini, M. Momeni, and M. Sarmadi, "An improved local branching approach for train formation planning," *Applied Mathematical Modelling*, vol. 37, no. 4, pp. 2300–2307, 2013.

- [9] K. R. Smilowitz, A. Atamtürk, and C. F. Daganzo, "Deferred item and vehicle routing within integrated networks," *Transportation Research Part E: Logistics and Transportation Review*, vol. 39, no. 4, pp. 305–323, 2003.
- [10] Y. Sun, C. Cao, and C. Wu, "Multi-objective optimization of train routing problem combined with train scheduling on a high-speed railway network," *Transportation Research Part C: Emerging Technologies*, vol. 44, pp. 1–20, 2014.
- [11] T. G. Crainic, "Service network design in freight transportation," *European Journal of Operational Research*, vol. 122, no. 2, pp. 272–288, 2000.
- [12] C. Barnhart, H. Jin, and P. H. Vance, "Railroad blocking: a network design application," *Operations Research*, vol. 48, no. 4, pp. 603–614, 2000.
- [13] Y.-H. Chang, C.-H. Yeh, and C.-C. Shen, "A multiobjective model for passenger train services planning: application to Taiwan's high-speed rail line," *Transportation Research Part B: Methodological*, vol. 34, no. 2, pp. 91–106, 2000.
- [14] T. G. Crainic and J.-M. Rousseau, "Multicommodity, multi-mode freight transportation: a general modeling and algorithmic framework for the service network design problem," *Transportation Research Part B*, vol. 20, no. 3, pp. 225–242, 1986.
- [15] M. R. Bussieck, P. Kreuzer, and U. T. Zimmermann, "Optimal lines for railway systems," *European Journal of Operational Research*, vol. 96, no. 1, pp. 54–63, 1997.
- [16] B. Yu, H. Zhu, W. Cai, N. Ma, Q. Kuang, and B. Yao, "Two-phase optimization approach to transit hub location—the case of Dalian," *Journal of Transport Geography*, vol. 33, pp. 62–71, 2013.
- [17] H. N. Newton, C. Barnhart, and P. H. Vance, "Constructing railroad blocking plans to minimize handling costs," *Transportation Science*, vol. 32, no. 4, pp. 330–345, 1998.
- [18] B. Z. Yao, B. Yu, P. Hu, J. J. Gao, and M. H. Zhang, "An improved particle swarm optimization for carton heterogeneous vehicle routing problem with a collection depot," *Annals of Operations Research*, 2014.
- [19] 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: Emerging Technologies*, vol. 43, pp. 233–248, 2014.
- [20] 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.
- [21] B. Yu, N. Ma, W. J. Cai, T. Li, X. T. Yuan, and B. Z. Yao, "Improved ant colony optimisation for the dynamic multi-depot vehicle routing problem," *International Journal of Logistics Research and Applications*, vol. 16, no. 2, pp. 144–157, 2013.
- [22] Y. D. Zhang, S. H. Wang, G. L. Ji, and Z. C. Dong, "Genetic pattern search and its application to brain image classification," *Mathematical Problems in Engineering*, vol. 2013, Article ID 580876, 8 pages, 2013.
- [23] Y. D. Zhang, S. H. Wang, and G. L. Ji, "A rule-based model for bankruptcy prediction based on an improved genetic ant colony algorithm," *Mathematical Problems in Engineering*, vol. 2013, Article ID 753251, 10 pages, 2013.
- [24] B. Yu, Z. Z. Yang, X. S. 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.
- [25] H. Derbel, B. Jarboui, S. Hanafi, and H. Chabchoub, "Genetic algorithm with iterated local search for solving a location-routing problem," *Expert Systems with Applications*, vol. 39, no. 3, pp. 2865–2871, 2012.

Research Article

Modifying Regeneration Mutation and Hybridising Clonal Selection for Evolutionary Algorithms Based Timetabling Tool

Thatchai Thepphakorn,^{1,2} Pupong Pongcharoen,² and Chris Hicks³

¹Faculty of Industrial Technology, Pibulsongkram Rajabhat University, Phitsanulok 65000, Thailand

²Industrial Engineering Department, Centre of Operations Research and Industrial Applications (CORIA), Faculty of Engineering, Naresuan University, Phitsanulok 65000, Thailand

³Newcastle University Business School, Newcastle University, Newcastle upon Tyne NE1 7RU, UK

Correspondence should be addressed to Pupong Pongcharoen; pupongp@nu.ac.th

Received 19 September 2014; Accepted 16 December 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Thatchai Thepphakorn 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 outlines the development of a new evolutionary algorithms based timetabling (EAT) tool for solving course scheduling problems that include a genetic algorithm (GA) and a memetic algorithm (MA). Reproduction processes may generate infeasible solutions. Previous research has used repair processes that have been applied after a population of chromosomes has been generated. This research developed a new approach which (i) modified the genetic operators to prevent the creation of infeasible solutions before chromosomes were added to the population; (ii) included the clonal selection algorithm (CSA); and the elitist strategy (ES) to improve the quality of the solutions produced. This approach was adopted by both the GA and MA within the EAT. The MA was further modified to include hill climbing local search. The EAT program was tested using 14 benchmark timetabling problems from the literature using a sequential experimental design, which included a fractional factorial screening experiment. Experiments were conducted to (i) test the performance of the proposed modified algorithms; (ii) identify which factors and interactions were statistically significant; (iii) identify appropriate parameters for the GA and MA; and (iv) compare the performance of the various hybrid algorithms. The genetic algorithm with modified genetic operators produced an average improvement of over 50%.

1. Introduction

Metaheuristics are a class of approximation methods that solve complex optimisation problems that are beyond the scope of classical heuristics and optimisation methods [1]. They have been widely used to solve nondeterministic polynomial (NP) hard problems within acceptable computational time [2]. However, metaheuristic methods are stochastic and cannot guarantee an optimal solution [3]. Evolutionary algorithms (EA) are particularly popular metaheuristics and have been widely applied in the literature. There are three types of EA: evolutionary programming, evolutionary strategies, and genetic algorithms (GA) [4]. Evolutionary programming and evolutionary strategies have been used to solve continuous optimisation problems whilst GA have been mainly used for solving discrete optimisation problems [5].

GA are population based, stochastic search approaches that were inspired by biological evolution. GA include crossover and mutation genetic operations, which are artificial processes for producing new chromosomes. Chromosome selection mimics natural evolution to select a new population for next generation based on individual fitness [6]. GA have been widely applied to solve various optimisation problems [7] including production scheduling [8], course timetabling [9], examination timetabling [10], container packing [11], travelling salesman [12], bankruptcy prediction [13], and machine layout [14]. However, the simple GA may not be effective for solving problems with a very large solution space and many constraints [5].

The term memetic algorithm (MA) is used to describe evolutionary algorithms in which local search is used to a large extent [15]. MAs have received considerable attention

from researchers in many fields [5] including job shop scheduling [16], vehicle routing [17], exam timetabling [18], and nurse scheduling [19]. The MA has also been applied to solve course timetabling problems [20–25].

Genetic operations frequently produce infeasible solutions, which can be (i) discarded; (ii) penalised; or (iii) repaired [6]. However, discarding infeasible solutions or applying a high penalty is only an option when a large proportion of the chromosomes are feasible [9]. Gen and Cheng [6] recommended the repair option. In the algorithms adopted by previous research, the mutation and crossover processes produce a population that includes feasible and infeasible chromosomes. The infeasible chromosomes are then identified and repaired. However, for very large problems that are subject to numerous constraints, the repair process is likely to be highly complex and difficult to design [6]. A complex repair process may be very time consuming [5]. The literature has not considered the development of modified crossover and mutation operators that only produce feasible chromosomes. Such a strategy would likely be more computationally efficient, which would make it possible to conduct more searches within a given execution time.

The performance of evolutionary algorithms is dependent upon the parameters used (such as the population size, number of generations, and the probabilities of crossover and mutation). It is important to identify appropriate values for the parameters in order to obtain the best solutions [26]. There are four experimental strategies: (i) the best-guess approach; (ii) the trial and error approach; (iii) the one factor at a time experimental strategy; and (iv) the factorial experiment [27]. Montgomery [27] suggested that the factorial experiment is the best approach for dealing with several factors. The strategy is to systematically vary the factors together, instead of one at a time. Thus, it is best to use a factorial experiment when investigating appropriate parameter settings for metaheuristic methods. The approach is more reliable, leads to better results, and is more efficient than the alternatives [28].

Artificial immune systems (AIS) are metaheuristics that were inspired by the immune system in biology [29]. There are four main variants of the AIS: danger theory, immune network algorithm (INA), negative selection algorithm (NSA), and clonal selection algorithm (CSA) [30]. AIS have been successfully applied in three application areas: (i) learning; (ii) anomaly detection; and (iii) optimisation [31]. There is only a limited literature on the use of AIS for timetabling. He et al. [32] applied CSA to solve university course timetabling problems in Singapore and benchmark problems. The CSA produced better timetables than GA for all of the problems considered. Malim et al. [33] applied the INA, NSA, and CSA to solve course timetabling problems. The INA produced timetables with the best average fitness, whereas CSA was best in terms of average execution time. Bhaduri [34] was the only researcher to develop a hybrid AIS for timetabling, called GAIN, which included the INA and GA. The GAIN was able to produce optimal feasible timetables faster than GA. However, other researchers have used AIS hybrids in other domains. For example, Zhang et al. [35] combined AIS, the chaos operator, and particle swarm optimisation (PSO), to

produce CIPSO, which was used for transportation planning. The approach outperformed GA and PSO in respect of route optimality and convergence time.

The objectives of this paper were to

- (i) briefly review the literature on evolutionary algorithms and course timetabling;
- (ii) explain the development, process, and features of a novel timetabling tool that incorporates genetic algorithms, local search, the clonal selection algorithm, roulette wheel selection, and the elitist strategy;
- (iii) outline a new modified regeneration mutation operator (MRMO) that is based on roulette wheel selection;
- (iv) describe the development of a novel local search (LS) algorithm that guarantees the feasibility of new chromosomes generated with the MRMO. This hybrid is called the modified memetic algorithm (MMA);
- (v) explain experiments that demonstrated that performance can be improved by the MRMO and MMA using an elitist strategy (ES);
- (vi) outline new hybrid algorithms that include the clonal selection algorithm, MRMO+CSA, and MMA+CSA;
- (vii) describe the testing of the tool using widely used benchmark problems;
- (viii) explain the experimental design and analysis used to investigate the significance of GA parameters and interactions and to identify appropriate parameter settings;
- (ix) provide a comparison of the performance of the proposed hybridisations and the other hybridisations of EA which were used to find the best timetables using 14 benchmarking obtained from the literature.

The next section of this paper briefly reviews course timetabling problems, which is followed by a detailed outline of the development of the evolutionary algorithms based timetabling tool and its features, the experimental programme, results, analysis and conclusions.

2. Course Timetabling Problems

“Timetabling is the allocation, subject to constraints, of given resources to objects being placed in space time, in such a way as to satisfy as nearly as possible a set of desirable objectives” [36, page 266]. There are many types of timetabling problems including employee timetabling, sports timetabling, transportation timetabling, and educational timetabling [28]. Course timetabling arises every academic year in educational institutions (such as high schools, colleges, or universities) and is solved by academic or administrative staff with or without an automated timetabling tool. Course timetabling is known to be a NP-hard problem [37], in which the computational time required to find a solution increases exponentially with problem size [9].

Timetabling problems include hard constraints that must be satisfied in order to produce a feasible timetable and soft constraints, which are desirable but may be violated

[9, page 903]. In the case of course timetabling, it is necessary for a timetable to be feasible for students, lecturers, and classrooms [23]. In a university, a degree programme comprises a set of modules that must be completed by the students registered on the programme. Di Gaspero et al. [38] adopted the following constraints.

- (i) Hard constraints.
 - (a) *Lectures*: all lectures within a module must be assigned to regular periods. All lectures must be scheduled (HC_1).
 - (b) *Room occupancy*: only one lecture can take place in a room at a given time (HC_2).
 - (c) *Conflicts*: students and staff can only attend one lecture at a time (HC_3).
 - (d) *Availabilities*: lecturers must be available for a lecture to be scheduled (HC_4).
- (ii) Soft constraints.
 - (a) *Room capacity*: the room must have sufficient seats for the students on the module (SC_1).
 - (b) *Minimum working days between lectures*: for a particular module there should be a minimum amount of time between lectures (SC_2).
 - (c) *Curriculum compactness*: students on a degree programme should have lectures that are consecutive with no gaps (SC_3).
 - (d) *Room stability*: all lectures of a module should be given in the same room (SC_4).

Another issue is that events or courses may have differing priorities; the generation of infeasible solutions can be avoided by scheduling the highest priority activities first [39].

3. Evolutionary Algorithms Based Timetabling (EAT) Tool

The aim of this research was to generate timetables for lecturers, students, and classrooms that must satisfy all of the hard constraints and minimise the number of violations of the soft constraints proposed by Di Gaspero et al. [38].

The Evolutionary Algorithm based Timetabling (EAT) program was coded using the Tool Command Language and Toolkit (Tcl/Tk) [40]. It was developed in order to construct effective course timetables by using a genetic algorithm (GA) [41] and a memetic algorithm (MA) [42]. Both methods are population based and perform multiple directional search, which achieves a greater diversity than conventional optimisation methods that conduct a single directional search [6]. The MA and GA chromosomes have different components. For MAs, the chromosomes consist of a set of memes, whereas with GAs the chromosomes comprise a set of genes [43]. The key difference is that the memes used by the MA can be self-adapting based upon local search and refinement, whereas genes do not have this capability [6].

The artificial immune system (AIS) was initially proposed in the mid 1980s by Farmer et al. [44]. The clonal selection

algorithm is a well-known variants of the AIS that is based upon two immune system principles: clonal selection and affinity maturation [45]. Each antibody (candidate solution) would be cloned proportionally to its antigenic affinity (fitness) value, in which the higher antigenic affinity would have the higher number of cloned antibodies [46]. Affinity maturation is related to hypermutation and receptor editing [46]. The regulation of hypermutation is a rapid accumulation of mutations that depend upon receptor affinity, in which the cell receptor with the higher affinity is mutated by using a mutation rate that is lower than for solutions with lower fitness [46]. Receptor editing provides a mechanism for escaping from the local optima, which increases the diversity of solutions [46]. The elimination percentage %B specifies how many low affinity antibodies are eliminated from the receptors.

The main procedures within the evolutionary algorithms based timetabling tool are shown in Figure 1. The first step is to the represent events within the timetable as memes/genes. The second step is to combine memes/genes to produce an initial population that represents a set of possible timetables. This part of the algorithm is designed to ensure that all of the candidate solutions are feasible. This is followed sequentially by genetic algorithms, local search, and a clonal selection algorithm, which is repeated for the required number of generations. The GA operators, LS, and CSA are designed to ensure that all of the chromosomes produced are feasible. There is an elitist strategy selection mechanism after the local search processes that selects the chromosomes for the CSA algorithm and also remembers the good solutions in its memory. There is a subsequent roulette wheel selection process after the CSA, which produces a population of chromosomes. A further elitist replacement process substitutes weaker solutions within the population with solutions remembered by the elitist strategy if they are better. The following subsections describe these processes in more detail.

3.1. Meme/Gene Representation. This research used the same data structures for genetic algorithms, memetic algorithms, and the clonal selection algorithm. The terminology used to describe the data structures varies according to the algorithm. With a genetic algorithm a chromosome comprises a set of genes. With a memetic algorithm a chromosome comprises a set of Memes. The clonal selection algorithm described in Section 3.7 uses an identical structure to represent antibodies.

A meme/gene can be encoded using either numeric (binary, integer, or real) or alphanumeric characters [9]. In this work, an integer encoded meme consists of three coded numbers: classrooms ($r = 1, 2, 3, \dots, R$); days per week ($d = 1, 2, 3, \dots, D$); and periods or timeslots per day ($t = 1, 2, 3, \dots, T$). Each meme/gene contains a reference to a classroom, a day, and a timeslot; for example, $\{1, 2, 4\}$ represents an event in the first classroom that takes place on the second days in the fourth timeslot. A chromosome comprises a set of memes/genes that represent a complete timetable.

3.2. Chromosome Initialisation. The chromosome initialisation process takes the following steps.

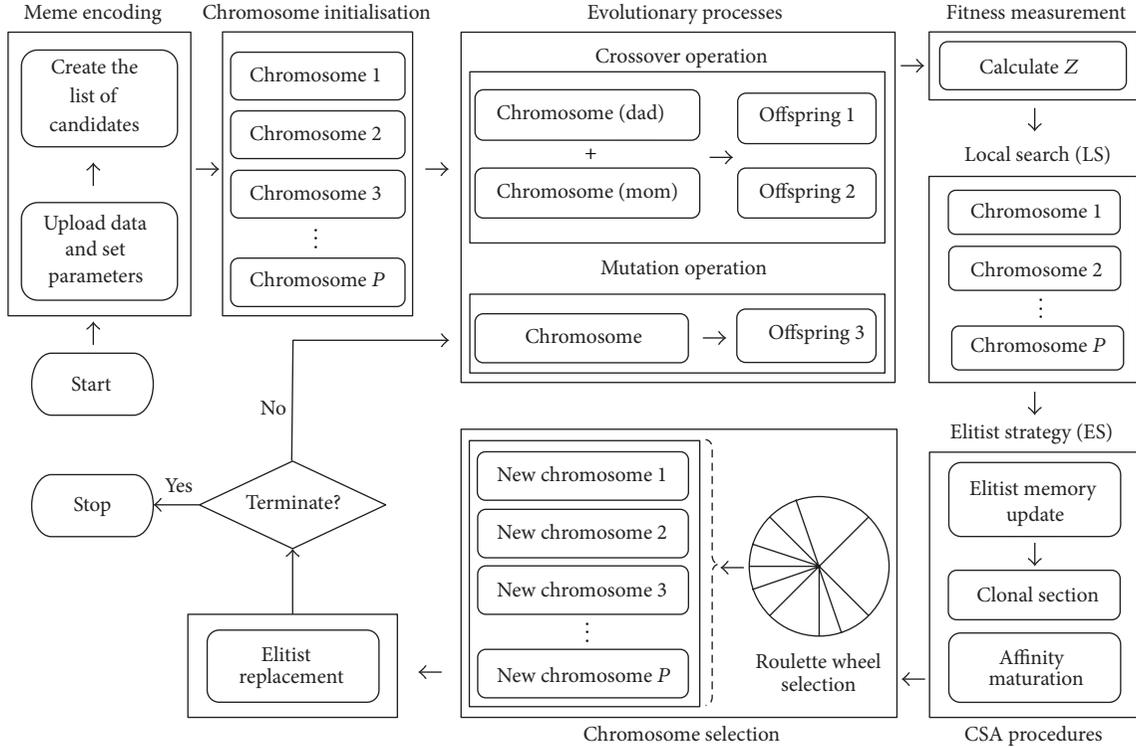


FIGURE 1: Main procedures of evolutionary algorithms based timetabling tool.

- (1) The length of the chromosome required is calculated taking into account the number of degree programmes, modules, and their associated classes.
- (2) An empty chromosome is generated with the appropriate length.
- (3) The modules are then sorted based upon their relative importance.
- (4) The highest priority module is scheduled first: this entails generating memes/genes for all of the classes and randomly assigning them to the chromosome. Before a meme/gene is added a check is made to ensure that the hard constraints are not violated. If there is a violation the algorithm sequentially looks for the next meme/gene that does not contravene the constraints (taking into account the modules in priority order); the process is then repeated in priority order until all the modules have been scheduled.

3.3. Evolutionary Processes. The parent chromosomes are randomly selected for the crossover and mutation genetic operations according to the probabilities of crossover (P_C) and mutation (P_M). The selection of these parameters determines the balance between exploration and exploitation. The crossover operation (COP) produces offspring chromosomes from two parent chromosomes, whereas the mutation operation produces random meme/gene changes in one chromosome. The number of memes/genes within the chromosome that are changed is determined by the mutation rate M_R . Thus, the selection of chromosomes is related to P_M

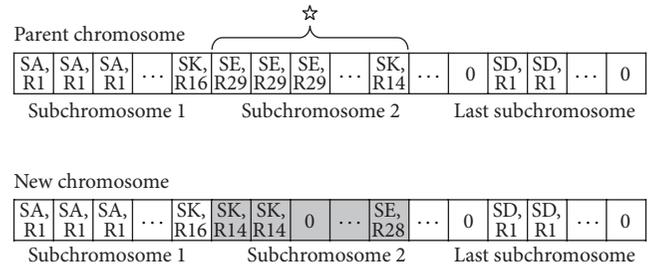


FIGURE 2: Regeneration mutation operator [9].

whereas the selection within the chromosome is related to the parameter M_R .

The EAT includes three types of crossover operation: one-point crossover (OP), two-point crossover (TP) [47], and position based crossover (PB) [48], which were modified to ensure that only feasible chromosomes can be produced. The modified version of the regeneration mutation [9] was developed to ensure feasible solutions.

Figure 2 illustrates the regeneration mutation operator. It includes three steps. First, a chromosome is randomly selected from the population. Secondly, a section (subchromosome) is selected for regeneration. Finally, a new subchromosome is generated randomly. The remaining genes within the chromosome are inherited from the parent.

Their modified regeneration mutation operator made four modifications to the operator: (i) some memes from the parent chromosome are randomly regenerated so that

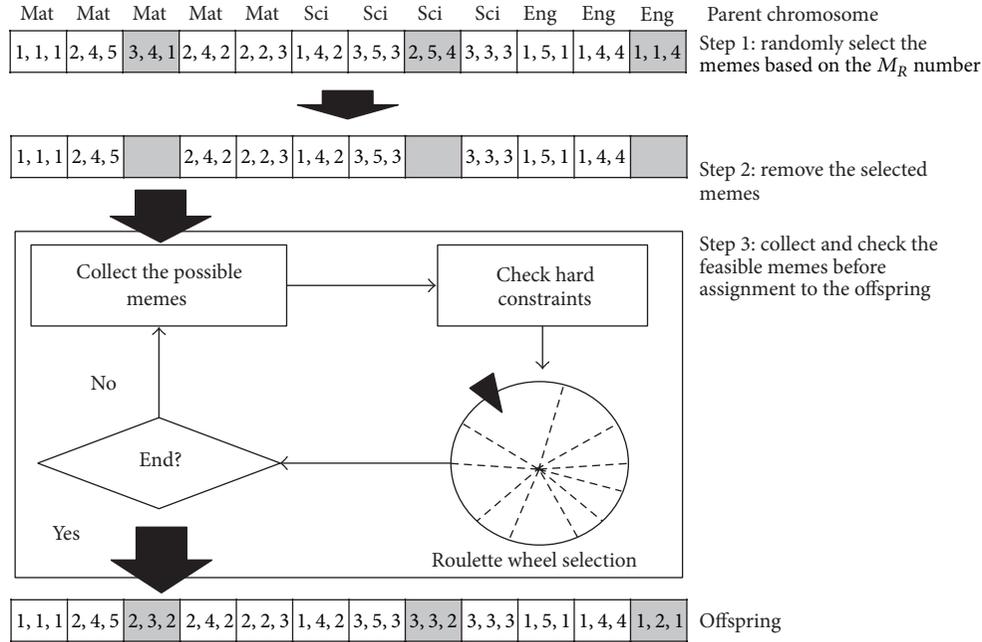


FIGURE 3: The modified regeneration mutation operation.

beneficial memes are inherited by the offspring; (ii) new feasible memes are assigned into the empty chromosome positions by using roulette wheel selection; (iii) all of the offspring are guaranteed to be feasible chromosomes because of the hard constraint checking before memes which are inserted into the empty positions of a new chromosome; and (iv) the parameter M_R specifies the percentage of memes to be regenerated. A higher setting of M_R increases the amount of exploration, but this may result in beneficial memes from the parent being lost. The modified regeneration mutation procedure is illustrated in Figure 3.

3.4. Fitness Measurement. The total violation index (Z) for a timetable may be calculated using (1) [49].

$$\text{Minimise } Z = \sum_{i=1}^S W_i SC_i \quad (1)$$

$$\text{Subject to: } HC_j = 0, \quad \forall j, \quad (2)$$

where i is an index relating to the i_{th} soft constraint ($i = 1, 2, 3, \dots, S$), where S is the number of soft constraints; j is the index for the j_{th} hard constraint ($j = 1, 2, 3, \dots, H$), where H is the number of hard constraints. SC_i is a variable used to count the number of violations of the i_{th} soft constraint. HC_j is the variable used to count the number of violations of the j_{th} hard constraint. For a timetable to be feasible HC_j must be zero for all the hard constraints. The user can specify the relative importance of the soft constraints by adjusting the weightings W_i for each soft constraint. Higher weightings indicate higher priority of the associated soft constraints. In this work, the weights (W_1 – W_4) were set at 1, 5, 2, and 1, respectively, as recommend by Di Gaspero et al. [38].

The GA and MA measure the quality of each chromosome using the objective function from (1) to calculate the total violation index (Z). As the objective is to minimise the number of violations the fitness value, which is determined by [23]

$$\text{Fitness value} = \frac{1}{1 + Z}. \quad (3)$$

3.5. Local Search (LS). The objective of the local search (LS) within the MA is to (i) improve the quality of chromosome or solution, through increased exploitation and (ii) increase the opportunity to quickly discover the global best solution. In this work, two hill-climbing LS heuristics, LS1 and LS2, were adopted from previous work by Thepphakorn et al. [28], as it had been demonstrated that they improved chromosome quality and prevented the generation of infeasible chromosomes. The aim of LS1 is to reduce the number of violations of the first and the fourth soft constraints (SC_1 and SC_4), whilst the LS2 aims to reduce the number of violations of the second and the third soft constraints (SC_2 and SC_3). After the LS1 and LS2 procedures, the total violation index (Z) and the fitness values for the new chromosomes are measured again before performing chromosome selection.

3.6. Elitist Strategy (ES). The ES aims to maintain high quality chromosomes from one generation to the next. The ES helps GAs to reach convergence more quickly [13]. This ES is divided into two subprocesses: elitist memory updating, which records the best solutions (with no duplicates) and elitist replacement, which substitutes the worst chromosomes with those remembered if they are better. The elitist replacement process takes place after chromosome selection. The proportion of chromosomes remembered by the ES is

TABLE 1: Characteristics of course timetabling problems [38].

Problems	Characteristics of course timetabling problems						
	Number of modules	Number of events	Number of classrooms	Number of periods/week	Number of Lecturers	Number degree programmes	Unavailability constraints
1	30	160	6	30	24	14	53
2	82	283	16	25	71	70	513
3	72	251	16	25	61	68	382
4	79	286	18	25	70	57	396
5	54	152	9	36	47	139	771
6	108	361	18	25	87	70	632
7	131	434	20	25	99	77	667
8	86	324	18	25	76	61	478
9	76	279	18	25	68	75	405
10	115	370	18	25	88	67	694
11	30	162	5	46	24	13	94
12	88	218	11	36	74	150	1368
13	82	308	19	25	77	66	468
14	85	275	17	25	68	60	486

determined by a user specified parameter %ES. Previous research had indicated that the most appropriate value for this parameter is 75% [50].

3.7. Clonal Selection Algorithms (CSA). In this research, the memory of the ES is used to produce the hybridisations for (i) the genetic algorithm combined with the CSA and (ii) the memetic algorithm (which had been modified to include hill climbing local search) which was combined with the CSA. When chromosomes are assigned to the elitist memory, the procedure attempts to further improve them through the application of the CSA. The elitist memory is updated if the resultant chromosome is better after the application of the CSA.

All of the chromosomes (antibodies) in the elitist memory are sorted in accordance to their affinities (fitnesses); the chromosome with the highest affinity is assigned the highest rank $k = 1$, ($k = 1, 2, 3 \dots, n$), whilst the chromosome with the lowest affinity is ranked $k = n$. The total number of antibodies (n) for cloning is equal to the number of chromosomes in the ES memory. In the following step, each k rank of antibodies contained in the elitist memory is cloned according to [46]

$$N_c = \sum_{k=1}^n \text{round} \left(\frac{\beta \cdot P}{k} \right), \quad (4)$$

where N_c is the total number of cloned antibodies, $\text{round}(\cdot)$ is an operator for changing real values into integers, β is the multiplying factor, and P is the population size. In the last step, all of the cloned antibodies are generated using affinity maturation. The regeneration mutation operator is used in this process together with a variable mutation rate M_{Rv} that is an adaptive setting based upon an antibody's ranking. The initial setting for M_{Rv} is determined by the parameter M_R .

Antibodies with a lower affinity (ranking) require a value of M_{Rv} that is greater than those with higher ranking [46].

3.8. Chromosome Selection. The classical roulette wheel approach [41] was used in this research. The general concept of the roulette wheel selection is to randomly select which chromosomes in the current population survive into the next population in such a way that their probability of survival depends upon their fitness. This process is terminated when the desired population size has been generated.

4. Experimental Results and Analysis

The objective of the EAT is to construct course timetables with the lowest number of soft constraint violations (Z). The aims of the computational experiments were to (i) identify which main factors and their interactions were statistically significant for the GA; (ii) identify and verify the best parameter settings; (iii) explore the performance of the GA with modified regeneration mutation (called MRMO); and (iv) explore the performance of the proposed hybridisations including MRMO+CSA and MMA+CSA.

The research considered fourteen course timetabling problems that were provided by the third track of ITC2007 [38]. These are summarized in Table 1. The experiments were performed on a personal computer with Intel 2.67 GHz Core 2 Duo CPU and 4 GB of RAM.

4.1. Screening Experiment. The screening experiment had two objectives to identify which factors and first level interactions were statistically significant and to identify the best settings for these factors. The experimental design, shown in Table 2, was used together with data from timetabling problem 1 (a small problem). The factors included (i) the combination of population size and the number of generations (PG), which

TABLE 2: Experimental factors and levels for the GA.

Factors	Levels	Factor values		
		Low (-1)	Medium (0)	High (+1)
PG	3	25 * 100	50 * 50	100 * 25
P_C	3	0.6	0.75	0.9
P_M	3	0.1	0.2	0.3
COP	3	One-point (OP)	Two-point (TP)	Position-based (PB)
M_R	3	0.1	0.5	0.9

TABLE 3: One-third fraction 3^{5-1} experimental design.

Runs	PG	P_C	P_M	COP	M_R
1	-1	-1	-1	-1	-1
2	-1	-1	-1	0	0
3	-1	-1	-1	1	1
4	-1	-1	0	-1	0
5	-1	-1	0	0	1
⋮	⋮	⋮	⋮	⋮	⋮
79	1	1	1	-1	-1
80	1	1	1	0	0
81	1	1	1	1	1

determines the total number of chromosomes generated, which determines the amount of search and influences the execution time. In the computational experiments the value was fixed at 2,500 to limit the time taken for computational search; (ii) the probability of crossover (P_C); (iii) the probability of mutation (P_M); (iv) the crossover operation (COP); and (v) the mutation rate (M_R).

The total number of runs required for a full factorial experiment based on the design in Table 2 would consider all the combinations of the factors in each replication. The total number of runs would therefore be the number of factors times the number of levels times the number of replications, which would be $3^5 = 243$ runs per replication. When resources are limited it is common for researchers to use fractional factorial designs, which use a carefully chosen subset (fraction) of the experimental runs required for a full factorial design. This approach is based upon the sparsity of effects principle that states that a system is usually dominated by main effects and low order interactions [27].

In this experiment, the one-third fraction 3^{5-1} experimental design shown in Table 3 was adopted for the screening experiment, which decreased the number of computational runs by 66.67% per replication compared to the full factorial approach. The first instant problem (see Table 1) was selected and replicated five times using different random seeds. The computational results obtained from the 405 ($3^{5-1} * 5$) runs were analysed using a general linear model form of analysis of variance (ANOVA). Table 4 shows the ANOVA table, which shows the source of variation (*Source*), degrees of freedom (DF), sum of square (SS), mean square (MS), F value, and P value. ANOVA was used to test the null hypothesis that there was no effect (H_0) and the alternative hypothesis (H_1) that

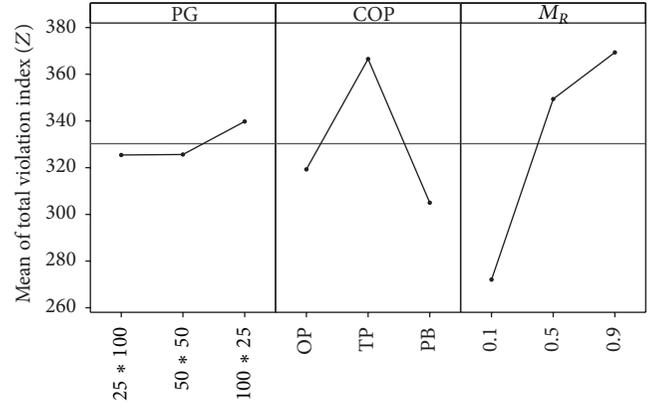


FIGURE 4: Main effect plots of PG, COP, and M_R factors.

there is an effect for each factor and interaction [27]. If a 95% confidence interval is used H_1 is accepted if $P \leq 0.05$, but H_0 cannot be rejected if $P > 0.05$.

Table 4 shows the GA parameters in terms of the main effect and first level interactions. PG, COP, M_R , $PG * P_C$, $PG * COP$, $PG * M_R$, and $P_M * M_R$ were statistically significant with a 95% confidence interval. The random seed number (*seeds*) did not statistically affect the GA performance. However, it is best not to discard parameters having a P value more than 0.05 but less than 0.2 in a screening experiment [26]. Moreover, the most influential factor in this experiment was M_R because it had the highest F value followed by the COP factor.

4.2. Multiple Comparison Analysis. The experimental design considered three different levels for each factor. The alternative hypothesis (H_1) obtained from the ANOVA only identifies that at least one level of a factor has a statistically different mean, but it is not known whether the other levels are significant [27]. Thus, in some cases it is not possible to select the appropriate parameter settings from the ANOVA because it is not known which pairs of results are significantly different. After the screening experiment, appropriate parameter settings for the GA were determined by using the lowest mean obtained from main effect and interaction plots. These are shown in Figures 4 and 5 for the statistically significant GA factors. Figure 4 indicates that the best settings are $PG = 25 * 100$, $COP = PB$, and $M_R = 0.1$. Figure 5 shows the best combinations for the interactions, which are $PG = 25 * 100$ and $M_R = 0.1$; $PG = 50 * 50$

TABLE 4: ANOVA analysis of GA parameters.

Source	DF	SS	MS	F value	P value
PG	2	18,274	9,137	7.65	0.001
P_C	2	4,362	2,181	1.83	0.163
P_M	2	4,589	2,295	1.92	0.148
COP	2	280,473	140,236	117.36	0.000
M_R	2	712,893	356,446	298.29	0.000
Seeds	4	7,413	1,853	1.55	0.187
PG * P_C	4	12,304	3,076	2.57	0.038
PG * P_M	4	4,341	1,085	0.91	0.459
PG * COP	4	15,613	3,903	3.27	0.012
PG * M_R	4	44,420	11,105	9.29	0.000
P_C * P_M	4	3,419	855	0.72	0.582
P_C * COP	4	4,544	1,136	0.95	0.435
P_C * M_R	4	3,259	815	0.68	0.605
P_M * COP	4	3,971	993	0.83	0.506
P_M * M_R	4	14,475	3,619	3.03	0.018
COP * M_R	4	8,257	2,064	1.73	0.143
Error	350	418,238	1,195		
Total	404	1,560,845			

TABLE 5: Pairwise comparisons using Tukey's method for the significant main effects.

Factors	(j)	(i)	Mean difference (i - j) Z	T value	P value
PG	25 * 100	50 * 50	0.21	0.05	0.999
	25 * 100	100 * 25	14.36	3.41	0.002
	50 * 50	100 * 25	14.14	3.36	0.002
COP	OP	TP	47.30	11.24	0.000
	OP	PB	-14.28	-3.39	0.002
	TP	PB	-61.58	-14.64	0.000
M_R	0.1	0.5	77.33	18.38	0.000
	0.1	0.9	97.28	23.12	0.000
	0.5	0.9	19.95	4.74	0.000

with COP = PB; PG = 50 * 50 with $P_C = 0.75$; and $M_R = 0.1$ with $P_M = 0.2$.

Tukey's method [27] is a statistical analysis tool that may be used for multiple or pairwise comparisons. The hypothesis testing used by Tukey's method can be defined as follows: H_0 cannot be rejected if the means of pair $i - j$ are equal; that is, the mean difference between the pair $i - j$ is zero (P value > 0.05); otherwise, H_1 will be accepted for pair $i - j$ with P value ≤ 0.05 [27]. Many statisticians prefer to use this approach because the overall error rate is controlled [27]. Tukey's method was therefore applied to detect significant differences in pairs of means in terms of the main and interaction effects. The results obtained with Tukey's comparison are shown in Tables 5 and 6, each of which consists of the significant factors, the pairs of factor levels between i and j considered, the mean difference between i and j (Mean dif. $i - j$), and the T value and the P value.

The comparative results obtained from using Tukey's method for the significant main effects shown in Table 5 indicated that the mean (mean difference $i - j$) difference

in penalty Z for the factor PG between 25 * 100 and 50 * 50 was not statistically different (H_0 cannot be rejected) whilst the means of other pairs were different (H_1 will be accepted). Moreover, the means obtained from the pairs of the COP and M_R factors were statistically different from each other pair. As there were many pairs of interactions, only the pairs that had the lowest mean in the interaction plots shown in Figure 5 were selected for pairwise comparisons. The analysis of the results obtained with Tukey's method for the selected significant interactions is shown in Table 6. The mean obtained with the M_R factor at 0.1 was not statistically different when the PG factor was set at either 25 * 100 or 50 * 50. This indicates that all three levels of the PG factor are appropriate for use with the PB crossover. The P_C parameter settings were practicable at all levels when the PG factor was set at either 25 * 100 or 50 * 50. The means obtained for all of the levels of the P_M factor were usable when the M_R factor was set at 0.1. Therefore, the appropriate parameter setting for the GA factors PG, COP, M_R , P_C , and P_M were established as 25 * 100 or 50 * 50, PB, 0.1, 0.6-0.9, and 0.1-0.3, respectively.

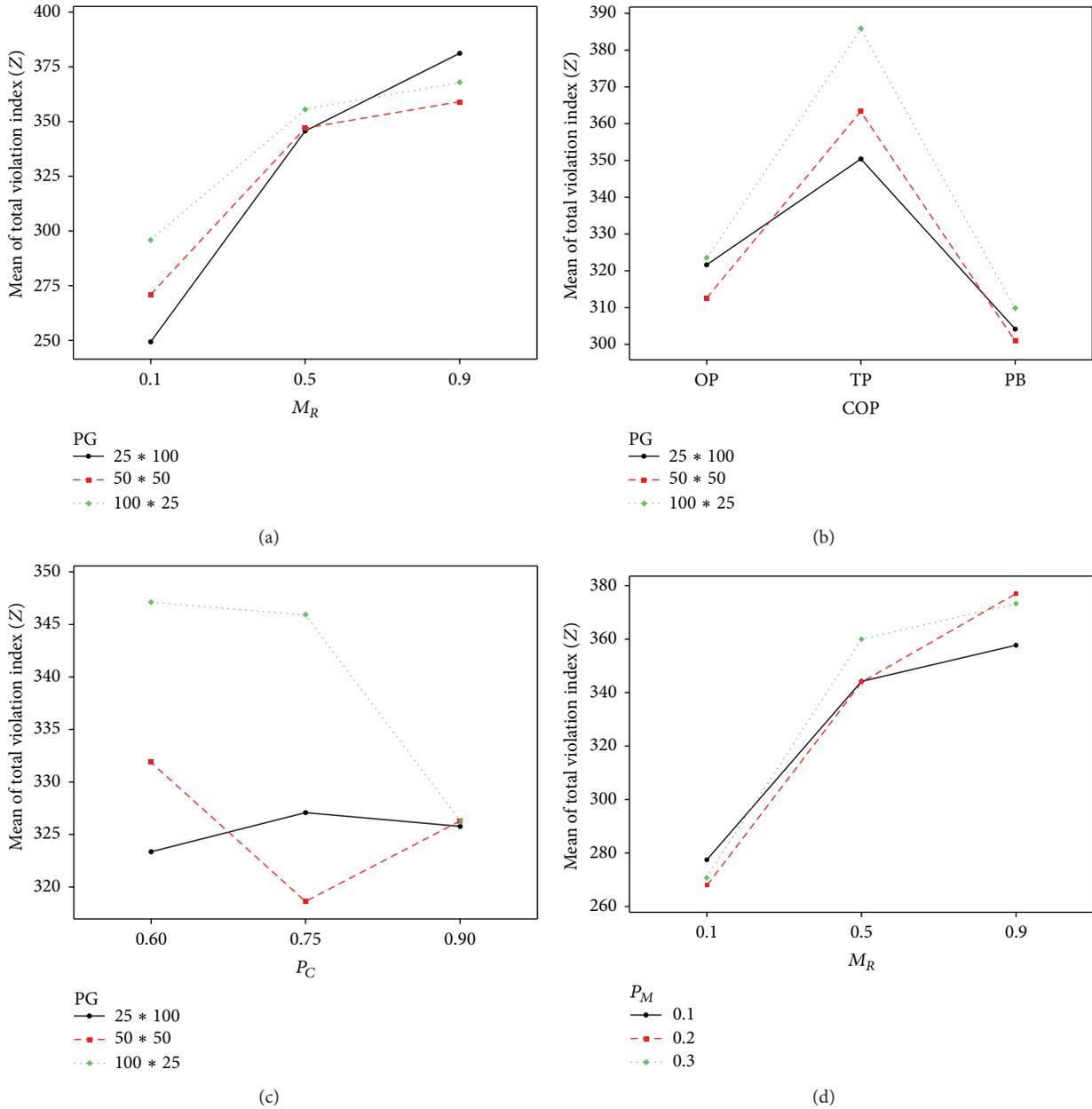


FIGURE 5: Interaction plots of PG * P_C , PG * COP, PG * M_R , and P_M * M_R .

4.3. *Verifying Appropriate Parameter Settings.* The significant factors, interactions, and important differences in the means were investigated by the screening experiment and pairwise comparison. Montgomery [27] suggested that the region for significant factors leading to the best possible response should be explored by conducting a second optimisation experiment after the screening experiment. The COP factor was a discrete GA parameter. The previous experiment identified PB as the best setting. PG settings between 25 * 100 and 50 * 50 showed little difference after using the pairwise analysis (see in Table 5), except for the M_R factor. Therefore, the region of M_R around 0.1 should be verified by using an experimental design before carrying out a comparative study.

Five levels of M_R were therefore considered 0.02, 0.06, 0.1, 0.14, and 0.18. The appropriate settings identified by the previous experiments were used for the other parameters. The first instant problem from the ITC2007 was selected and repeated ten times using different random seed numbers. The results obtained from the computational runs for the best so far solution were statistically analysed in terms of minimum (Min), maximum (Max), and average (Avg) penalty value Z as well as the standard deviation (SD) and the execution time (Time) (hour unit).

The experimental results are shown in Table 7. The M_R setting of 0.1 produced the best performance with the lowest average, minimum, and maximum values. The higher values

TABLE 6: Pairwise comparisons using Tukey's method for the selected significant interaction.

Factors	(j)		(i)		Mean difference (i - j) Z	T value	P value
PG * P_C	25 * 100	0.75	50 * 50	0.75	-8.42	-1.16	0.965
	25 * 100	0.6	25 * 100	0.75	3.73	0.51	0.999
	25 * 100	0.75	25 * 100	0.9	-1.31	-0.18	1.000
	50 * 50	0.6	50 * 50	0.75	-13.24	-1.82	0.671
	50 * 50	0.75	50 * 50	0.9	7.62	1.05	0.981
PG * COP	25 * 100	PB	50 * 50	PB	-3.27	-0.45	1.000
	25 * 100	PB	100 * 25	PB	5.67	0.78	0.998
	50 * 50	PB	100 * 25	PB	8.93	1.23	0.951
PG * M_R	25 * 100	0.1	50 * 50	0.1	21.67	2.97	0.073
	25 * 100	0.1	100 * 25	0.1	46.51	6.38	0.000
	50 * 50	0.1	100 * 25	0.1	24.84	3.41	0.019
P_M * M_R	0.1	0.1	0.2	0.1	-9.42	-1.29	0.933
	0.1	0.1	0.3	0.1	-6.73	-0.92	0.992
	0.2	0.1	0.3	0.1	2.69	0.37	1.000

TABLE 7: Verifying the optimal setting for M_R .

Factor levels	Significant factor	Best so far solution				
	M_R	Min Z	Max Z	Avg Z	SD	Time (hrs)
Lowest (-2)	0.02	228	312	260.60	24.29	0.11
Low (-1)	0.06	216	286	255.20	21.18	0.12
Medium (0)	0.10	163	254	222.30	25.77	0.12
High (1)	0.14	193	328	260.40	37.33	0.13
Highest (2)	0.18	224	314	256.70	25.32	0.14

of M_R required more computational time than the lower parameter settings. This analysis verified that the optimal setting of M_R for the GA is 0.1.

4.4. Performances of GA with/without the Modified Regeneration Mutation. The objective of this experiment was to explore and compare the performance of GA with/without the modified regeneration mutation operator (MRMO) in terms of the speed of convergence and the quality of the solutions. The appropriate parameter settings for the GA with the MRMO for PG, COP, M_R , P_C , and P_M were found to be 25 * 100, PB, 0.1, 0.75, and 0.2, respectively. The benchmark problems adopted from the third track of the ITC2007 (14 instances) [38] were used to test and compare the performance of the proposed algorithms to find the course timetable with the lowest penalty Z. The computational run for each instance was repeated ten times by using different random seeds. The computational results were analysed in terms of Avg, SD, time (hour unit), and the percentage improvement achieved by the GA with MRMO (%Imp). A *t*-test was used to compare the means.

Table 8 shows that the performance differences achieved by the GA with/without the MRMO were all statistically significant with a 95% confidence interval using *t*-test analysis (P value ≤ 0.05) for all of the problems. It means that the GA with the MRMO outperformed the GA without the MRMO for all instances, each of which also had the lower Avg and SD values. Moreover, the %Imp value for each

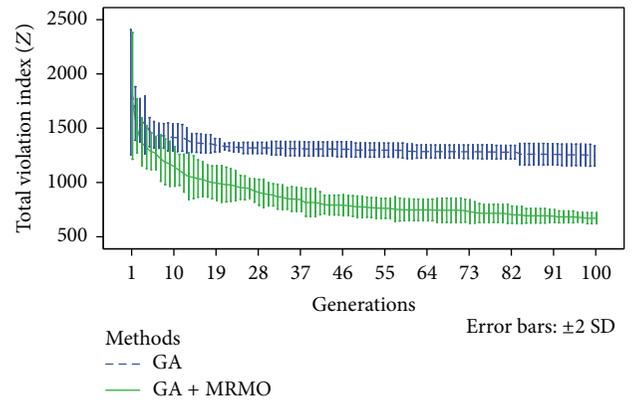


FIGURE 6: A comparison of convergence achieved by the GA with/without MRMO for problem number 7.

problem was distributed between 35.55% and 87.56% but with longer execution time. The average of improvement was up to 51.88%. A comparison of the convergence speed for the proposed methods to investigate the best so far solution is shown in Figures 6 and 7 by using problems number 7 and number 14 from the ITC2007 datasets.

The GA with MRMO converged more quickly for problems number 7 and number 14 than the GA without the MRMO (see Figures 6 and 7). Therefore, it can be concluded that the new regeneration mutation based upon roulette

TABLE 8: Comparative study between GA with/without MRMO.

Problems	GA			GA with MRMO			<i>t</i> -test analysis		%Imp
	Avg Z	SD	Time (hrs)	Avg Z	SD	Time (hrs)	<i>T</i> value	<i>P</i> value	
1	222.30	25.77	0.12	90.60	16.59	0.12	13.59	0.000	59.24
2	710.30	37.97	1.51	416.90	26.74	1.52	19.98	0.000	41.31
3	683.50	28.20	0.67	370.20	26.77	0.68	25.48	0.000	45.84
4	724.70	24.98	0.62	318.70	19.99	0.62	40.13	0.000	56.02
5	1549.60	154.74	0.57	998.70	113.07	0.58	9.09	0.000	35.55
6	1087.70	26.90	0.82	555.70	25.85	0.82	45.09	0.000	48.91
7	1244.50	46.14	1.23	670.50	25.43	1.23	34.45	0.000	46.12
8	689.90	19.17	0.73	300.70	14.48	0.71	51.22	0.000	56.41
9	717.90	24.19	0.87	373.10	21.48	0.85	33.70	0.000	48.03
10	839.20	38.41	0.92	408.70	19.92	0.90	31.47	0.000	51.30
11	403.40	78.19	0.19	50.20	16.43	0.19	13.98	0.000	87.56
12	1397.00	52.02	0.63	854.40	49.25	0.63	23.95	0.000	38.84
13	806.60	37.29	0.71	362.00	23.32	0.71	31.97	0.000	55.12
14	705.90	27.59	0.58	310.30	17.48	0.58	38.30	0.000	56.04
							Avg %Imp		51.88

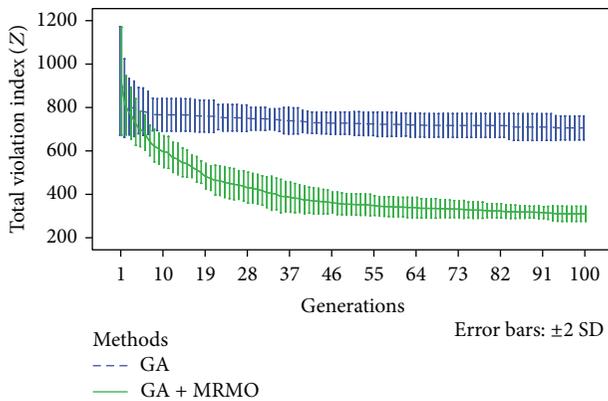


FIGURE 7: A comparison of convergence achieved by the GA with/without MRMO for problem number 14.

wheel selection was able to improve the GA’s performance in terms of solution quality and speed.

4.5. *Analysing the Performance Evolutionary Algorithm Hybridisations.* The objective of this experiment was to explore the performance of (i) the MRMO with/without local search (the modified memetic algorithm (MMA)); (ii) the addition of the elitist strategy (MRMO+ES and MMA+ES); and (iii) the use of the clonal selection algorithm (MRMO+CSA and MMA+CSA) both in terms of convergence speed and solution quality. The appropriate parameter settings for the MRMO and the MMA were adopted from the previous experiments. The benchmark problems adopted from the third track of the ITC2007 [38] were again used to test and compare the performance of the proposed algorithms to find the course timetable with the lowest penalty Z. The computational run for each instance was repeated ten times by using different random seed numbers. The computational

results obtained were analysed statistically in terms of Avg, SD, and time (hour unit), as shown in Table 9. The percentage improvement (%Imp) achieved by the MRMO with/without hybrid heuristics was calculated, whilst the *T* value obtained by using the *t*-test method and the *P* value are also shown in Table 10.

Table 10 shows that almost all of the comparisons between the results obtained from the MRMO and the other hybridisation approaches were statistically significant with a 95% confidence interval (*P* value ≤ 0.05). For all of the problems the results obtained from the MMA+ES, MRMO+CSA, and MMA+CSA were statistically significant with a 95% confidence interval. Moreover, the MMA+CSA achieved the highest *T* value, %Imp, and Avg %Imp, which indicates that it was the best configuration. However, the negative or positive *T* value in Table 10 indicated that the results obtained from some hybrid approaches did not outperform the MRMO for some problems.

According to Tables 9 and 10, the MMA+CSA outperformed the other methods for all instances because it achieved the maximum Avg %Imp of 49.85% and minimum Avg values. However, it also had the longest execution time. Although the Avg %Imp between the MRMO+CSA and the MMA+ES was nearly equal at 42%, the MRMO+CSA required less computational time than both the MMA+CSA and the MMA+ES; it was up to 6.3 times quicker for some instances. Moreover, the Avg %Imp obtained by the MRMO using CSA was better than that using LS (MMA) and ES by approximately 23–26%. The MRMO’s execution times using CSA were also up to 5.7 times faster than those using LS but slower than those using ES by up to 3.2 times for some instances. Although the Avg %Imp obtained from the MRMO using ES and LS was less than those using the proposed hybrid methods, the performances of the MRMO+ES and the MMA were better than the MRMO without hybridisations (see in Table 10). The average improvement for almost all problems

TABLE 9: Performance explorations for EA with/without ES, LS, and CSA hybridisations.

Methods	Statistical analysis	Problems													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
MRMO + ES	Avg Z	63.6	349.2	325.7	275.4	856.1	442.6	531.6	269.3	322.7	357.9	24.2	805.7	343.7	299.6
	SD	11.1	29.9	14.5	20.7	85.9	31.4	21.7	20.5	13.1	18.0	6.7	57.3	14.9	13.8
	Time (hrs)	0.12	1.53	0.68	0.61	0.58	0.83	1.24	0.72	0.89	0.93	0.20	0.63	0.71	0.59
MMA (MRMO + LS)	Avg Z	46.4	353.5	362.2	249.3	938.9	378.2	423.6	255.7	377.1	340.8	14.9	1032.9	300.1	312.0
	SD	7.5	21.8	25.1	13.8	67.4	19.3	17.4	15.1	22.1	17.1	4.0	59.8	17.3	12.9
	Time (hrs)	0.45	5.09	2.80	2.68	3.64	4.68	7.00	2.94	3.08	4.95	0.60	6.14	3.12	2.93
MMA + ES	Avg Z	33.5	274.7	263.4	176.4	715.9	273.4	297.6	193.7	267.6	240.2	8.0	729.9	232.3	205.8
	SD	5.3	13.1	14.5	13.6	70.1	24.5	17.2	10.4	16.0	11.7	1.8	53.7	9.9	8.3
	Time (hrs)	0.47	5.17	2.74	2.60	4.82	4.60	7.00	3.04	2.91	4.91	0.60	6.81	3.45	2.94
MRMO + CSA	Avg Z	25.1	271.9	248.1	187.1	678.7	293.5	336.7	190.5	251.6	266.8	5.9	678.9	234.5	210.9
	SD	3.4	21.6	18.0	6.7	59.4	22.6	16.2	13.2	15.1	11.7	2.5	65.1	12.6	11.1
	Time (hrs)	0.29	1.89	1.08	1.08	1.05	1.17	1.54	1.12	1.42	1.26	0.64	1.08	1.24	1.00
MMA + CSA	Avg Z	21.2	242.3	231.6	146.6	569.3	230.5	260.0	170.0	242.5	212.5	4.4	672.4	200.5	181.1
	SD	2.7	24.6	20.0	7.3	43.5	19.4	13.4	10.8	12.5	14.0	1.6	65.3	9.2	13.4
	Time (hrs)	0.58	5.47	3.06	3.19	6.20	5.05	7.39	3.27	3.90	5.25	1.03	6.79	3.76	3.25

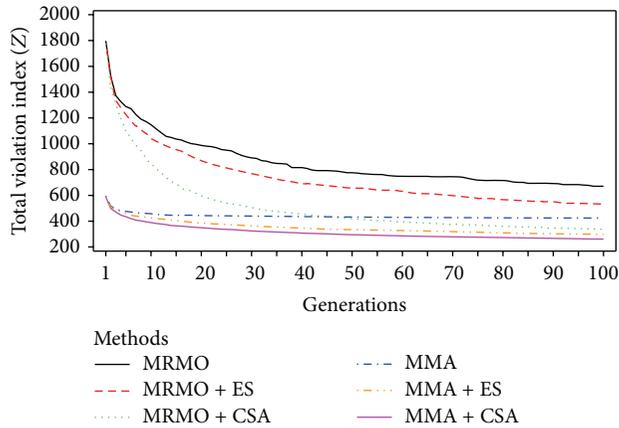


FIGURE 8: A comparison of convergence amongst EA hybridisations for problem number 7.

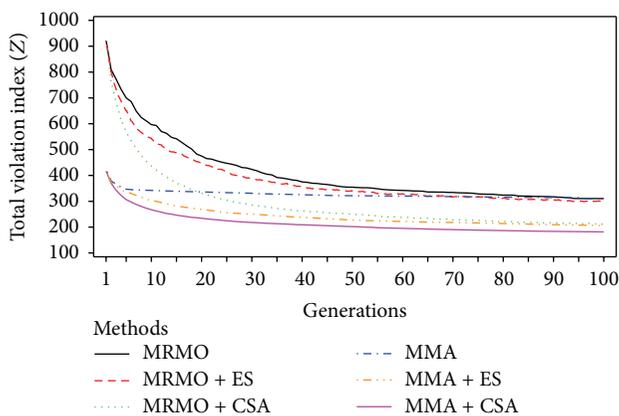


FIGURE 9: A comparison of convergence for EA hybridisations for problem number 14.

was around 16–19%. A comparison of the results in terms of average convergence speeds of the proposed hybrid methods to find the best so far solution is shown in Figures 8 and 9. These were based upon problem numbers 7 and 14 from the ITC2007 which represent medium and large problem sizes.

The MMA+CSA's converged more quickly than the other algorithms for problems 7 and 14. The next best convergence was achieved by MMA+ES (Figures 8 and 9). The MRMO+CSA had low performance in early generations. However, the average of best so far solutions found in the last generation was close to the average of best so far solutions obtained by the MMA+CSA. Moreover, the LS strategy hybridisation in the MRMO including the MMA+CSA, the MMA+ES, and the MMA was able to find a better average of best so far solutions in early generations than the other methods without LS. Therefore, it can be concluded that the LS, ES, and CSA strategies were able to improve the MRMO's performance in terms of the solution quality and speed.

5. Conclusions

The evolutionary algorithms based timetabling (EAT) tool was developed to use genetic algorithms (GA) and memetic

algorithms (MA) to solve university course timetabling problems. The work made a number of significant research contributions. A common problem with genetic algorithms is that many chromosomes within a population may represent infeasible solutions. This work developed new one-point, two-point, and position-based crossover operators and a modified regeneration mutation operators that guaranteed that all of the chromosomes generated represented feasible solutions. Likewise the chromosome initialisation process was designed to produce feasible chromosomes. The research also developed novel hybrids that included genetic algorithms, local search, and a clonal selection algorithm together with roulette wheel and elitist selection. The tool was tested using 14 datasets obtained from the third track of ITC2007 [38], which have been widely used by previous researchers.

The experimental work adopted a sequential experimental design. The screening experiment used a one-third fraction of the 3^{k-1} experimental design [27] with five factors, each of which had three levels. The factors PG, COP, M_R , $PG * P_C$, $PG * COP$, $PG * M_R$, and $P_M * M_R$ were statistically significant with a 95% confidence interval. Main effect plot analysis found the best settings to be $PG = 25 * 100$, $COP = PB$, and $M_R = 0.1$. The best combinations for the interactions were $PG = 25 * 100$ and $M_R = 0.1$; $PG = 50 * 50$ with $COP = PB$; $PG = 50 * 50$ with $P_C = 0.75$; and $M_R = 0.1$ with $P_M = 0.2$.

A further analysis using pairwise comparison found the appropriate parameter setting for PG, COP, M_R , P_C , and P_M to be $25 * 100$ or $50 * 50$, PB, 0.1, 0.6–0.9, and 0.1–0.3, respectively. A further experiment verified that the best setting for M_R was 0.1, as it produced the best performance with the lowest average, minimum and maximum penalty values.

The comparative results indicated that the MRMO outperformed the GA for all problems, with an average improvement of up to 51.88%. The MRMO converged more quickly than GA. In terms of hybrid comparisons, the MMA+CSA outperformed all the other methods; there was an average improvement of 49.85% compared to the MRMO. The second best hybrid was the MRMO+CSA. The MMA+CSA also converged more quickly and the best so far solutions were better than for all the other hybrid methods for all generations. Although the performance of the MRMO+CSA was the second rank in terms of an average of %Imp, it required up to 6.3 times less computational time less than the MMA+CSA. The ES, LS, and CSA embedded within the EAT tool were able to improve the EA's performances in terms of solution quality and its convergence but at the expense of longer execution time.

Thus, the development of novel hybrids has been shown to be an effective approach to solve a wide range of timetabling problems. The proposed approaches have been shown to provide good solutions quickly.

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 acknowledge the Naresuan University for financial support on publication.

References

- [1] I. H. Osman and G. Laporte, "Metaheuristics: a bibliography," *Annals of Operations Research*, vol. 63, pp. 513–623, 1996.
- [2] R. Lewis, "A survey of metaheuristic-based techniques for university timetabling problems," *OR Spectrum*, vol. 30, no. 1, pp. 167–190, 2008.
- [3] E.-G. Talbi, *Metaheuristics: From Design to Implementation*, John Wiley & Sons, 2009.
- [4] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: overview and conceptual comparison," *ACM Computing Surveys*, vol. 35, no. 3, pp. 268–308, 2003.
- [5] S. Salcedo-Sanz, "A survey of repair methods used as constraint handling techniques in evolutionary algorithms," *Computer Science Review*, vol. 3, no. 3, pp. 175–192, 2009.
- [6] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, NY, USA, 1997.
- [7] S. S. Chaudhry and W. Luo, "Application of genetic algorithms in production and operations management: a review," *International Journal of Production Research*, vol. 43, no. 19, pp. 4083–4101, 2005.
- [8] P. Pongcharoen, C. Hicks, and P. M. Braiden, "The development of genetic algorithms for the finite capacity scheduling of complex products, with multiple levels of product structure," *European Journal of Operational Research*, vol. 152, no. 1, pp. 215–225, 2004.
- [9] P. Pongcharoen, W. Promtet, P. Yenradee, and C. Hicks, "Stochastic optimisation timetabling tool for university course scheduling," *International Journal of Production Economics*, vol. 112, no. 2, pp. 903–918, 2008.
- [10] Z. N. Azimi, "Hybrid heuristics for examination timetabling problem," *Applied Mathematics and Computation*, vol. 163, no. 2, pp. 705–733, 2005.
- [11] P. Thapatsuwan, P. Pongcharoen, C. Hicks, and W. Chainate, "Development of a stochastic optimisation tool for solving the multiple container packing problems," *International Journal of Production Economics*, vol. 140, no. 2, pp. 737–748, 2012.
- [12] P. Pongcharoen, W. Chainate, and S. Pongcharoen, "Improving artificial immune system performance: inductive bias and alternative mutations," in *Artificial Immune Systems*, vol. 5132 of *Lecture Notes in Computer Science*, pp. 220–231, Springer, 2008.
- [13] Y. Zhang, S. Wang, and G. Ji, "A rule-based model for bankruptcy prediction based on an improved genetic ant colony algorithm," *Mathematical Problems in Engineering*, vol. 2013, Article ID 753251, 10 pages, 2013.
- [14] S. Vitayasak and P. Pongcharoen, "Machine selection rules for designing multi-row rotatable machine layout considering rectangular-to-square ratio," *Journal of Applied Operational Research*, vol. 5, no. 2, pp. 48–55, 2013.
- [15] E. K. Burke and J. D. Landa Silva, "The design of memetic algorithms for scheduling and timetabling problems," in *Recent Advances in Memetic Algorithms*, W. Hart, J. E. Smith, and N. Krasnogor, Eds., pp. 289–311, Springer, Berlin, Germany, 2005.
- [16] J.-H. Yang, L. Sun, H. P. Lee, Y. Qian, and Y.-C. Liang, "Clonal selection based memetic algorithm for job shop scheduling problems," *Journal of Bionic Engineering*, vol. 5, no. 2, pp. 111–119, 2008.
- [17] R. Tavakkoli-Moghaddam, A. R. Saremi, and M. S. Ziaee, "A memetic algorithm for a vehicle routing problem with backhauls," *Applied Mathematics and Computation*, vol. 181, no. 2, pp. 1049–1060, 2006.
- [18] S. Abdullah, H. Turabieh, B. McCollum, and P. McMullan, "A tabu-based memetic approach for examination timetabling problems," in *Rough Set and Knowledge Technology*, vol. 6401 of *Lecture Notes in Computer Science*, pp. 574–581, Springer, Berlin, Germany, 2010.
- [19] E. Özcan, "Memes, self-generation and nurse rostering," in *Practice and Theory of Automated Timetabling VI*, vol. 3867 of *Lecture Notes in computer Science*, pp. 85–104, Springer, 2006.
- [20] E. Özcan, A. J. Parkes, and A. Alkan, "The interleaved constructive memetic algorithm and its application to timetabling," *Computers and Operations Research*, vol. 39, no. 10, pp. 2310–2322, 2012.
- [21] S. Abdullah and H. Turabieh, "On the use of multi neighbourhood structures within a Tabu-based memetic approach to university timetabling problems," *Information Sciences*, vol. 191, pp. 146–168, 2012.
- [22] M. Basikhasteh and M. A. Movafaghpour, "Hybridizing genetic algorithm with biased chance local search," *World Academy of Science, Engineering and Technology*, vol. 80, pp. 354–359, 2011.
- [23] R. Lewis and B. Paechter, "Finding feasible timetables using group-based operators," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 3, pp. 397–413, 2007.
- [24] D. Qaurooni and M.-R. Akbarzadeh-T, "Course timetabling using evolutionary operators," *Applied Soft Computing Journal*, vol. 13, no. 5, pp. 2504–2514, 2013.
- [25] B. T. Tesfaldet, "Automated lecture timetabling using a memetic algorithm," *Asia-Pacific Journal of Operational Research*, vol. 25, no. 4, pp. 451–475, 2008.
- [26] P. Pongcharoen, D. J. Stewardson, C. Hicks, and P. M. Braiden, "Applying designed experiments to optimize the performance of genetic algorithms used for scheduling complex products in the capital goods industry," *Journal of Applied Statistics*, vol. 28, no. 3–4, pp. 441–455, 2001.
- [27] D. C. Montgomery, *Design and Analysis of Experiments*, John Wiley & Sons, New York, NY, USA, 8th edition, 2012.
- [28] T. Thepphakorn, P. Pongcharoen, and C. Hicks, "An ant colony based timetabling tool," *International Journal of Production Economics*, vol. 149, pp. 131–144, 2014.
- [29] L. N. de Castro and J. Timmis, "An artificial immune network for multimodal function optimization," in *Proceedings of the Congress on Evolutionary Computation (CEC '02)*, pp. 699–704, IEEE, 2002.
- [30] D. Dasgupta, "Advances in artificial immune systems," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 40–49, 2006.
- [31] E. Hart and J. Timmis, "Application areas of AIS: the past, the present and the future," *Applied Soft Computing Journal*, vol. 8, no. 1, pp. 191–201, 2008.
- [32] Y. L. He, S. C. Hui, and E. M. K. Lai, "Automatic timetabling using artificial immune system," in *Algorithmic Applications in Management*, N. Megiddo, Y. F. Xu, and B. H. Zhu, Eds., vol. 3521 of *Lecture Notes in Computer Science*, pp. 55–65, Springer, Berlin, Germany, 2005.
- [33] M. R. Malim, A. T. Khader, and A. Mustafa, "Artificial immune algorithms for university timetabling," in *Proceedings of the 6th International Conference on Practice and Theory of Automated Timetabling (PATAT '06)*, pp. 234–245, 2006.

- [34] A. Bhaduri, "University time table scheduling using genetic artificial immune network," in *Proceedings of the International Conference on Advances in Recent Technologies in Communication and Computing*, pp. 289–292, IEEE, Kottayam, India, October 2009.
- [35] 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.
- [36] E. K. Burke and S. Petrovic, "Recent research directions in automated timetabling," *European Journal of Operational Research*, vol. 140, no. 2, pp. 266–280, 2002.
- [37] R. Qu, E. K. Burke, and B. McCollum, "Adaptive automated construction of hybrid heuristics for exam timetabling and graph colouring problems," *European Journal of Operational Research*, vol. 198, no. 2, pp. 392–404, 2009.
- [38] L. Di Gaspero, B. McCollum, and A. Schaerf, "The second international timetabling competition (ITC-2007): curriculum-based course timetabling track," in *Proceedings of the 14th RCRA Workshop on Experimental Evaluation of Algorithms for Solving Problems with Combinatorial Explosion*, M. Gavaneli and T. Mancini, Eds., Rome, Italy, 2007.
- [39] E. K. Burke and J. P. Newall, "Solving examination timetabling problems through adaption of heuristic orderings," *Annals of Operations Research*, vol. 129, pp. 107–134, 2004.
- [40] J. K. Ousterhout and K. Jones, *Tcl and the Tk Toolkit*, Addison-Wesley, Reading, Mass, USA, 2nd edition, 2009.
- [41] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading, Mass, USA, 1989.
- [42] P. Moscato and M. G. Norman, "A memetic approach for the travelling salesman problem—implementation of a computational ecology for combinatorial optimisation on message-passing systems," in *Proceedings of the International Conference on Parallel Computing and Transputer Applications*, pp. 177–186, 1992.
- [43] E. Elbeltagi, T. Hegazy, and D. Grierson, "Comparison among five evolutionary-based optimization algorithms," *Advanced Engineering Informatics*, vol. 19, no. 1, pp. 43–53, 2005.
- [44] J. D. Farmer, N. H. Packard, and A. S. Perelson, "The immune system, adaptation, and machine learning," *Physica D: Nonlinear Phenomena*, vol. 22, no. 1–3, pp. 187–204, 1986.
- [45] D. Dasgupta, S. Yu, and F. Nino, "Recent advances in artificial immune systems: models and applications," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1574–1587, 2011.
- [46] L. N. de Castro and F. J. von Zuben, "Learning and optimization using the clonal selection principle," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 3, pp. 239–251, 2002.
- [47] T. Murata, H. Ishibuchi, and H. Tanaka, "Genetic algorithms for flowshop scheduling problems," *Computers & Industrial Engineering*, vol. 30, no. 4, pp. 1061–1071, 1996.
- [48] G. Syswerda, "Scheduling optimisation using genetic algorithm," in *Handbook of Genetic Algorithms*, pp. 332–349, 1991.
- [49] T. Thepphakorn and P. Pongcharoen, "Heuristic ordering for ant colony based timetabling tool," *Journal of Applied Operational Research*, vol. 5, no. 3, pp. 113–123, 2013.
- [50] P. Thapatsuwan, J. Sepsirisuk, W. Chainate, and P. Pongcharoen, "Modifying particle swarm optimisation and genetic algorithm for solving multiple container packing problems," in *Proceedings of the International Conference on Computer and Automation Engineering (ICCAE '09)*, pp. 137–141, IEEE, Bangkok, Thailand, March 2009.

Research Article

Color Image Encryption Algorithm Based on TD-ERCS System and Wavelet Neural Network

Kun Zhang and Jian-bo Fang

School of Mathematics and Statistics, Chuxiong Normal University, Chuxiong, Yunnan 675000, China

Correspondence should be addressed to Kun Zhang; zhangkunpost@qq.com

Received 9 August 2014; Revised 14 November 2014; Accepted 26 November 2014

Academic Editor: Vishal Bhatnaga

Copyright © 2015 K. Zhang and J.-b. Fang. 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 order to solve the security problem of transmission image across public networks, a new image encryption algorithm based on TD-ERCS system and wavelet neural network is proposed in this paper. According to the permutation process and the binary XOR operation from the chaotic series by producing TD-ERCS system and wavelet neural network, it can achieve image encryption. This encryption algorithm is a reversible algorithm, and it can achieve original image in the rule inverse process of encryption algorithm. Finally, through computer simulation, the experiment results show that the new chaotic encryption algorithm based on TD-ERCS system and wavelet neural network is valid and has higher security.

1. Introduction

With the rapid growth and application of multimedia based on the Internet system, image security becomes an important issue. Since the size of digital image is always much greater than text file and the digital images contents are strongly correlated, the traditional encryption algorithms such as Data Encryption Standard (DES), International Data Encryption Algorithm (IDEA), and Advanced Encryption Standard (AES) have the weakness of low-level efficiency [1, 2].

Chaos is a ubiquitous phenomenon in nature and chaotic system is also a complex nonlinear, nonequilibrium dynamic process. Chaos theory is established since 1970s from many different research areas, such as physics, mathematics, biology, and chemistry [3]. The chaotic systems are characterized by sensitive dependence on initial conditions and system parameters, similarity to random behavior, no periodicity, and continuous broadband power spectrum [4]. Most properties can meet some requirements of encryption. Therefore, image encryption technique based on chaotic system has more useful and practical applications in the recent years [5, 6]. However, many of the proposed schemes show some deficiencies, such as small key space and weak security [7]. TD-ERCS system is a new class of discrete chaotic map systems based on the physical model of ellipse

reflecting activity. Many researchers have proposed that TD-ERCS system is a discrete chaotic system with the steady complexity, and the pseudorandom sequences generated by TD-ERCS are suitable for use in information encryption [8, 9].

In computer science field, an artificial neural network (ANN) is massively parallel distributed processor made up of simple processing units that has a natural propensity for storing experiential knowledge and making it available for use. As ANNs have many important properties, such as massively parallel, highly connected structures consisting of a number of simple and nonlinear processing elements [10], artificial neural networks (ANNs) have already been applied to solve the image encryption. Many encryption methods based on ANNs have been suggested in research literature and can deal with the intractable problem of fast and highly secure encryption [11, 12].

A new color image encryption algorithm is designed in the paper. In Section 2, the mapping equation of TD-ERCS system is given. In Section 3, wavelet neural networks algorithm is presented in detail. In Section 4, the image encryption algorithm based on TD-ERCS system and wavelet neural network is proposed. Finally, experimental results and some conclusions are given in Section 5.

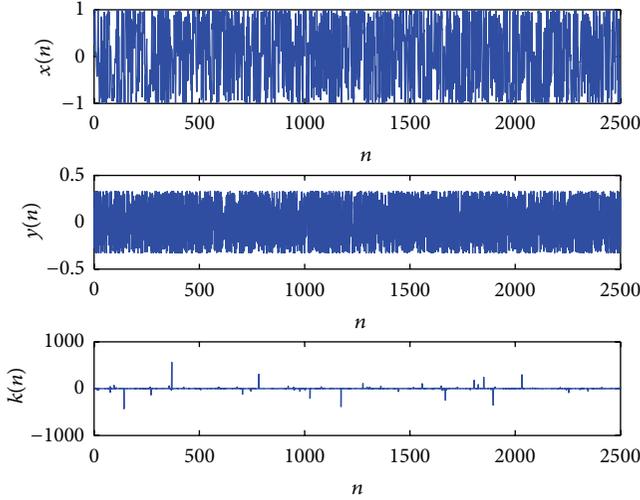


FIGURE 1: Time responses of TD-ERCS for seed parameters ($\mu = 0.3324$, $x_0 = 0.2456$, $\alpha = 2.143$, and $m = 2$).

2. Tangent-Delay Ellipse Reflecting Cavity-Map System

In 2004, Li-Yuan et al. found a new chaotic system bearing the name of Tangent-Delay Ellipse Reflecting Cavity-Map System (TD-ERCS) [13]. TD-ERCS system is a discrete chaotic system and has many properties such as the maximum Lyapunov exponent which is over zero, unchangeable equiprobability distribution, and zero correlation in total field [14]. TD-ERCS is described by

$$\begin{aligned}
 x_n &= \frac{-[2k_{n-1}y_{n-1} + x_{n-1}(u^2 - k_{n-1}^2)]}{u^2 + k_{n-1}^2}, \\
 k_n &= \frac{2k'_{n-m} - k_{n-1} + k_{n-1}(k'_{n-m})^2}{1 + 2k_{n-1}k'_{n-m} - (k'_{n-m})^2}, \\
 k'_{n-m} &= \begin{cases} -\frac{x_{n-1}}{y_{n-1}}\mu^2 & n < m \\ -\frac{x_{n-m}}{y_{n-m}}\mu^2 & n \geq m, \end{cases} \\
 y_n &= k_{n-1}(x_n - x_{n-1}) + y_{n-1}, \\
 k'_0 &= -\frac{x_0}{y_0}\mu^2, \\
 k_0 &= -\frac{\tan \alpha + k'_0}{1 - k'_0 \tan \alpha},
 \end{aligned} \tag{1}$$

where (μ, x_0, α, m) are called TD-ERCS seed parameters. In TD-ERCS seed parameters, $\mu \in (0, 1]$, initial $x_0 \in [-1, 1]$, $\alpha \in [0, \pi]$, and tangent-delay parameters $m = 2, 3, 4, 5, \dots$. Figure 1 shows the time responses of TD-ERCS

chaotic systems, for TD-ERCS seed parameters ($\mu = 0.3324$, $x_0 = 0.2456$, $\alpha = 2.143$, $m = 2$), iterations $n = 2500$ in (1).

3. Wavelet Neural Networks

3.1. The Wavelet Base Function. A cluster of functions which are got by time shift and scale stretch-out and drawback of mother wavelet function $u(t)$ are called the wavelet base functions. The wavelet base function is shown as follows [15]:

$$u_{a,b}(t) = \frac{1}{\sqrt{a}}u\left(\frac{t-b}{a}\right), \tag{2}$$

where a, b are the scale and translation parameters, respectively. The mother wavelet function $u(t)$ is satisfied using the following equation:

$$C_u = \int_{-\infty}^{+\infty} \frac{|U(f)|}{|f|} df < \infty, \tag{3}$$

where $U(f)$ is Fourier transform of $u(t)$.

3.2. The Model of Wavelet Neural Networks. Wavelet neural networks (WNNs) have emerged as new feed forward neural network based on wavelet transform, in which discrete wavelet function is used as the node activation function [16]. Since wavelet neural networks combine self-studying of neural network and the function of time-frequency localization of wavelet transform, they have strong ability to approximate and robust [17].

The architecture of the WNNs is presented in Figure 2. Wavelet neural networks commonly consist of three layers: input layer, hidden layer, and output layer. All the neurons in the layer are connected to the neurons in the next layer [18].

The output of three-layer WNNs is calculated in the form

$$y_k \approx \sum_{j=1}^n \omega_{jk} \psi\left(\frac{\sum_{i=1}^r u_{ij}x_i - b_j}{a_j}\right), \quad k = 1, 2, \dots, m, \tag{4}$$

where r is the neurons number in the input layer, m is the neurons number in the output layer, n is the neurons number in the hidden layer, and ω_{jk} is the connection weight between the j th neuron of hidden layer and the k th neuron of output layer. u_{ij} is the connection weight between the i th neuron of input layer and the j th neuron of hidden layer. b_j is the translation factor of the j th neuron in hidden layer. a_j is the expansion and contraction factor of the j th neuron in hidden layer. x_i is the i th neuron of input layer; y_k is the k th neuron of output layer. In the hidden layer, the activation function of neuron is Morlet wavelet function in [19]

$$\psi(t) = \cos(1.75t) e^{-t^2/2}. \tag{5}$$

Let us define error function $e(t)$ as

$$e(t) = y_e(t) - y(t), \tag{6}$$

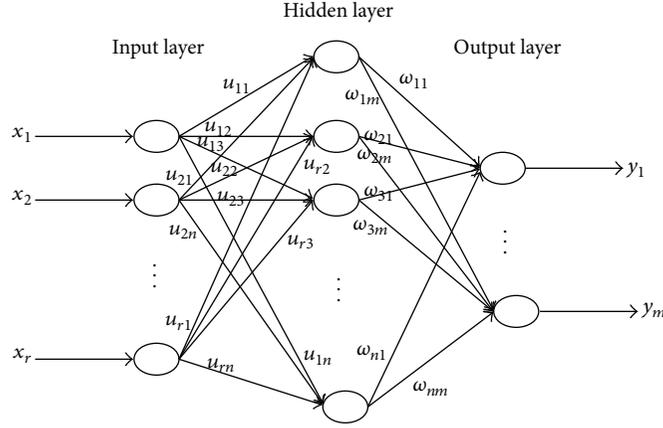


FIGURE 2: The architecture of wavelet neural networks.

where $y_e(t)$ is the model actual output and $y(t)$ is the desired output at time t . Then the cost function E can be defined as

$$E = \frac{1}{2} \sum_{i=1}^N (y_e(t) - y(t))^2 = \frac{1}{2} \sum_{i=1}^N (e(t))^2. \quad (7)$$

Using the gradient descent algorithms, the weight vector for every neuron in wavelet neural networks is updated as follows [20]:

$$\begin{aligned} \Delta \omega_{jk}(l+1) &= -\eta \frac{\partial E}{\partial \omega_{jk}(l)} + \alpha \Delta \omega_{jk}(l), \\ \Delta u_{ij}(l+1) &= -\eta \frac{\partial E}{\partial u_{ij}(l)} + \alpha \Delta u_{ij}(l), \\ \Delta a_j(l+1) &= -\eta \frac{\partial E}{\partial a_j(l)} + \alpha \Delta a_j(l), \\ \Delta b_j(l+1) &= -\eta \frac{\partial E}{\partial b_j(l)} + \alpha \Delta b_j(l), \\ \omega_{jk}(l+1) &= \omega_{jk}(l) + \Delta \omega_{jk}(l+1), \\ u_{ij}(l+1) &= u_{ij}(l) + \Delta u_{ij}(l+1), \\ a_j(l+1) &= a_j(l) + \Delta a_j(l+1), \\ b_j(l+1) &= b_j(l) + \Delta b_j(l+1), \end{aligned} \quad (8)$$

where l represents the backward step number and η and α are the learning and the momentum constants, differing in the ranges 0.01 to 0.1 and 0.1 to 0.9, respectively.

4. Image Encryption Based on TD-ERCS and Wavelet Neural Networks

4.1. Process of Permutation Cryptography. Let S be a real set: $S = \{s_1, s_2, \dots, s_n\}$. A permutation process of order n refers to

the operation of replacing an arrangement $\{p_i \mid p_i = i, i = 1, 2, \dots, n\}$ by a second arrangement $\{q_i \mid q_i = \pi(s_i)\}$, $\pi(s_i)$ denoted by the s_i numerical order in the S set in ascending or descending order, and $\pi(s_i) \in N$. Permutations group is represented as

$$\phi = \begin{pmatrix} 1 & 2 & \dots & n \\ \pi(s_1) & \pi(s_2) & \dots & \pi(s_n) \end{pmatrix}. \quad (9)$$

The reverse of this permutation process is specified as

$$\phi^{-1} = \begin{pmatrix} \pi(s_1) & \pi(s_2) & \dots & \pi(s_n) \\ 1 & 2 & \dots & n \end{pmatrix} \quad (10)$$

which retrieves the original arrangement.

Definition 1. A permutation π of X is a bijective function from X to X .

Based on Definition 1, the permutation cryptography process can be defined as follows.

Definition 2 (see [21]). If any data matrix X is transformed to a cipher-matrix $\varphi_z = \theta_z(X)$ where θ_z is any permutation operation, then the original matrix X can be obtained again from φ_z with the inverse operation of θ_z on it; that is, $\theta_z^{-1}(\varphi_z) = \theta_z^{-1}(\theta_z(X)) = X$, as $\theta_z^{-1}\theta_z$ forms an identity operator.

For example, suppose that plaintext G is

$$G = [0.234 \quad 1.222 \quad 0.007 \quad 0.145 \quad 0.201 \quad 0.341]. \quad (11)$$

The G permutations group in ascending order is

$$\phi_G = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 5 & 1 & 6 & 2 \end{bmatrix}. \quad (12)$$

Permutation cipher H is

$$H = [0.004 \ 0.421 \ 0.107 \ 0.423 \ 0.007 \ 0.221]. \quad (13)$$

The H permutations group in ascending order is

$$\phi_H = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 5 & 3 & 6 & 2 & 4 \end{bmatrix}. \quad (14)$$

Then

$$\begin{aligned} \phi_M &= \phi_G \phi_H = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 5 & 1 & 6 & 2 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 5 & 3 & 6 & 2 & 4 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 6 & 2 & 1 & 4 & 5 \end{bmatrix}. \end{aligned} \quad (15)$$

The ciphertext M is

$$M = [0.234 \ 0.201 \ 0.007 \ 0.341 \ 1.222 \ 0.145]. \quad (16)$$

The decryption process is

$$\begin{aligned} \phi_G &= \phi_M \phi_H = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 6 & 2 & 1 & 4 & 5 \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 5 & 3 & 6 & 2 & 4 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 5 & 1 & 6 & 2 \end{bmatrix}. \end{aligned} \quad (17)$$

4.2. Encryption Process. In the paper, the image encryption based TD-ERCS and WNNs mainly consist of two stages. The first stage of whole encryption system is chaotic sequence generation. In the chaotic sequence generation, a new chaotic sequence is generated by wavelet neural networks and TD-ERCS. The second stage of whole encryption system is the confusion stage. Since images are digital, a map is defined to transform the chaotic sequence to another sequence which consists of integers. Then the image can be encrypted by use of permutation operation and XOR with the integer sequence [22].

4.3. The Chaotic Sequence Generator. In this paper, we used chaotic sequences generated by TD-ERCS system and wavelet neural networks. Since wavelet neural networks have good capacity to approach arbitrary nonlinear mapping, they can possess chaotic state through studying of TD-ERCS chaotic sequence and modeling. The model system structure of chaotic sequences generator based on wavelet neural networks is shown in Figure 3. The feedback from the WNNs output end to input end shaped closed-loop structure and makes the output chaotic sequence feedback to input end as the initial value for next output sequence, so as to output the chaotic sequences continuously. The training data of WNNs are given by the TD-ERCS sequence. After the weights of WNNs are determined by learning algorithm, chaotic sequence $x(n)$ has been produced.

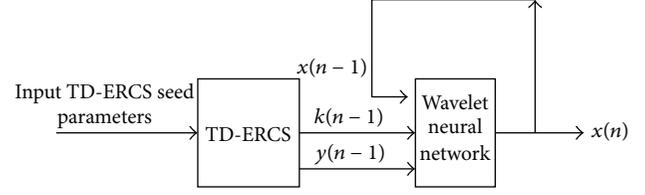


FIGURE 3: The model system structure of chaotic sequences generator based on WNNs.

4.4. Encryption Algorithm. Assume a square image H consists of $M \times N$ pixels as

$$H = \{h_{ij} \mid 1 \leq i \leq M, 1 \leq j \leq N\}, \quad (18)$$

where h_{ij} denotes pixel (i, j) of H image. The steps involved in the implementation of our encryption algorithm based on TD-ERCS and WNNs can be summarized as shown below.

Step 1. Set TD-ERCS seed parameters (μ, x_0, α, m) and generate TD-ERCS chaotic sequences x_n, y_n, z_n .

Step 2. Utilize WNNs to practice the chaotic sequences x_n, y_n, z_n and determine the weights of WNNs. The weights of WNNs and TD-ERCS seed parameters (μ, x_0, α, m) are the key and are transmitted to the receiver by secure channel.

Step 3. Calculate the size of the image $m \times n$, and image is changed to one-dimensional vector in the order of rank h_k [23]:

$$k = (i - 1) \cdot n + j \quad 1 \leq k \leq m \cdot n. \quad (19)$$

Step 4. Use the chaotic sequence generator to produce chaotic sequence $x'(n)$.

Step 5. Select $x'(n)$ as permutation ciphers and calculate image vector C_n by the permutation cryptography process.

Step 6. Generate a new chaotic sequence β_n :

$$\beta_n = \frac{\arccos(x'_n)}{\pi}, \quad 0 \leq \beta_n \leq 1, \quad (20)$$

where β_n binary forms are $\beta_n = (0.b_{n1}b_{n2} \cdots b_{np})$, $b_{ij} = \{0, 1\}$, and use β_k sequence to produce B_k :

$$B_k = b_{k1}b_{k2} \cdots b_{kp}. \quad (21)$$

Step 7. Calculate encrypted image vector H_k :

$$H_k = C_k \oplus B_k, \quad (22)$$

where the \oplus means XOR. The encryption process is finished.

The encryption algorithm is a reversible algorithm and the original image can be obtained by applying the inverse process of the encryption algorithm. The secret

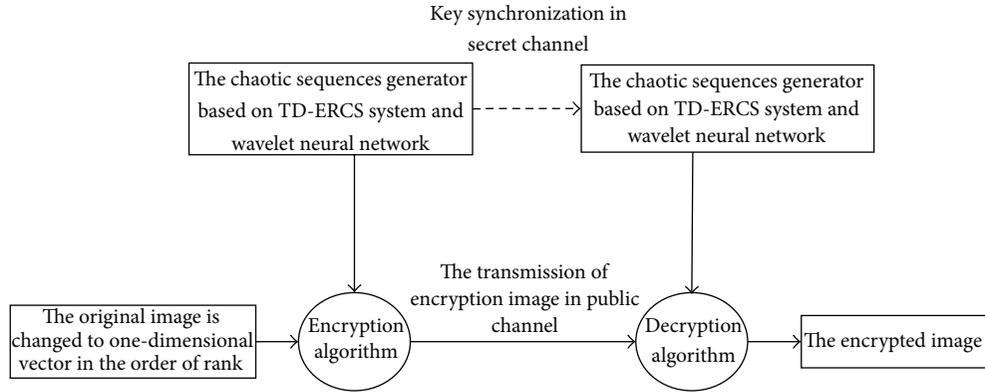


FIGURE 4: The secret communication model of image based on TD-ERCS system and wavelet neural network.

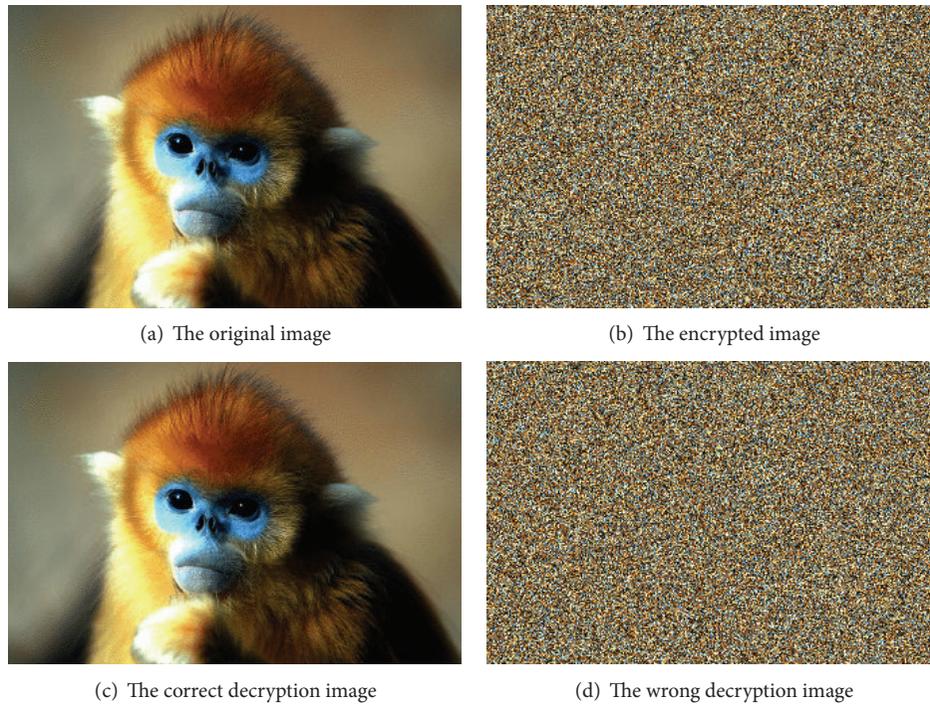


FIGURE 5: The results of color image encryption algorithm.

communication model based on TD-ERCS system and wavelet neural network is shown in Figure 4.

5. Experimental Results and Analysis

In the paper, experimental analysis of the proposed image encryption algorithm has been done. The experiment has been implemented in the Matlab 2009a in Figure 5(a). The original image is 256 level grayscale and 225×336 size.

The structure of WNNs was a three-layer wavelet neural network, in which the number of input layer neurons is three, the number of output layer neurons is one, and the number of hidden layer neurons is six. TD-ERCS system seed parameters are 0.1256, 0.8130, 0.5325, and 2. After 100 times iterations, the cost function E of wavelet neural network was 0.1696. The connection weights of wavelet neural network between input layer and hidden layer are given by

$$W_1 = \begin{bmatrix} 0.5768 & 0.8776 & -0.3325 & -1.0721 & 0.9519 & -0.7115 \\ -0.9919 & -0.5254 & 0.4382 & 0.7774 & -0.0699 & 0.6609 \\ 0.5320 & -1.3329 & 0.0147 & -0.0876 & 0.4870 & -1.106 \end{bmatrix}. \tag{23}$$

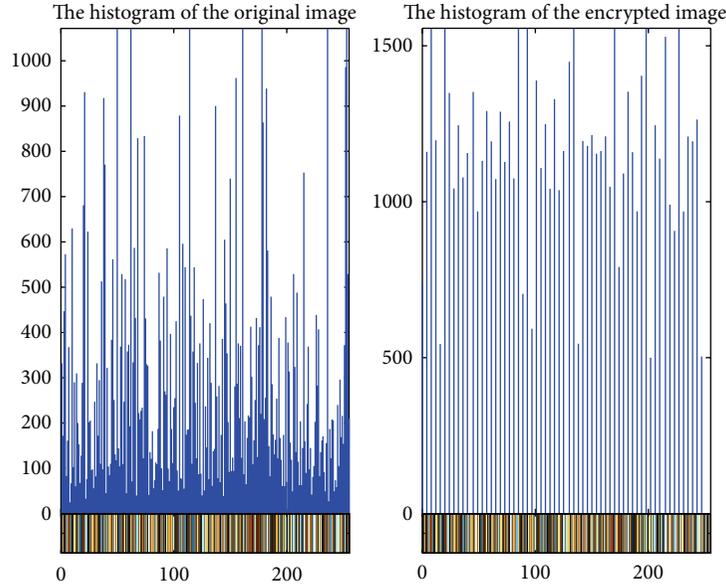


FIGURE 6: The histogram of the original and encrypted image.

The connection weights between hidden layer and output layer are given by

$$W_2 = \begin{bmatrix} -0.0813 \\ 0.7212 \\ 0.4381 \\ 0.2880 \\ 0.2938 \\ -0.2485 \end{bmatrix}. \quad (24)$$

The translation factors vector is given by

$$B = [-0.0679 \quad 0.1035 \quad -0.4452 \quad -1.6843 \quad -0.0160 \quad -0.3100]. \quad (25)$$

The expansion and contraction factors vector is given by

$$A = [0.8826 \quad -0.2703 \quad 0.4648 \quad 0.0897 \quad 1.5915 \quad -0.8423]. \quad (26)$$

Lyapunov exponent is a useful way to characterize and quantify chaotic phenomena arising in dynamical systems, which describe the temporal evolution of small perturbations of the initial conditions [24]. The maximum Lyapunov exponent of chaotic sequence is 0.6312 by Wolf's algorithm, and the result shows that the system is chaotic system [25]. The encrypted image is shown in Figure 5(b), which is rough-and-tumble and unrecognizable. As Figure 5(c) shows the decrypted image using the same encryption key is an exact version of the original image. Figure 5(d) shows the error between the original image and decrypted image, which is zero.

5.1. Key Space Analysis. For a secure image encryption algorithm, the key space should be large enough to make brute-force attacks infeasible [26]. In our algorithm, the TD-ERCS seed parameters and the connection weights of wavelet neural network can be used as keys. If the precision is 10^{-4} in the above test, the key space size is at least 10^{195} . The key space is large enough so that it can resist the exhaustive attack effectively.

5.2. Histograms Analysis. The image histogram is an important statistical characteristic of digital image and can illustrate how pixels in an image are distributed by graphing the number of pixels at each color intensity level [27]. The histograms of the original image and the encrypted image are shown in Figure 6. It is shown that the histogram of the encrypted image is fairly uniform and the encryption algorithm has covered up all the characters of the original image.

5.3. Correlation Coefficient Analysis. In the image data, each pixel is in neighborhood with eight adjacent pixels and is highly correlated with its adjacent pixels [28]. To test the correlation between two vertically adjacent pixels, two horizontally adjacent pixels, and two diagonally adjacent pixels in the encryption image, we randomly select 3015 pairs of two adjacent (in horizontal, vertical, and diagonal direction) pixels from the original and encryption image. Then, calculate their correlation coefficient using the following four formulas [29, 30]:

$$E(x) = \frac{1}{N} \sum_{i=1}^N x,$$

$$D(x) = \frac{1}{N} \sum_{i=1}^N [x - E(x)]^2,$$

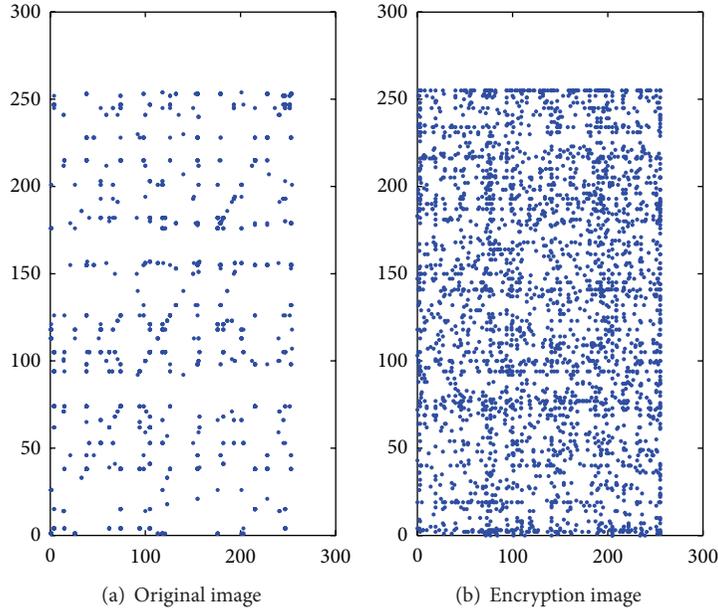


FIGURE 7: The correlation analysis of original and encryption image in horizontal.

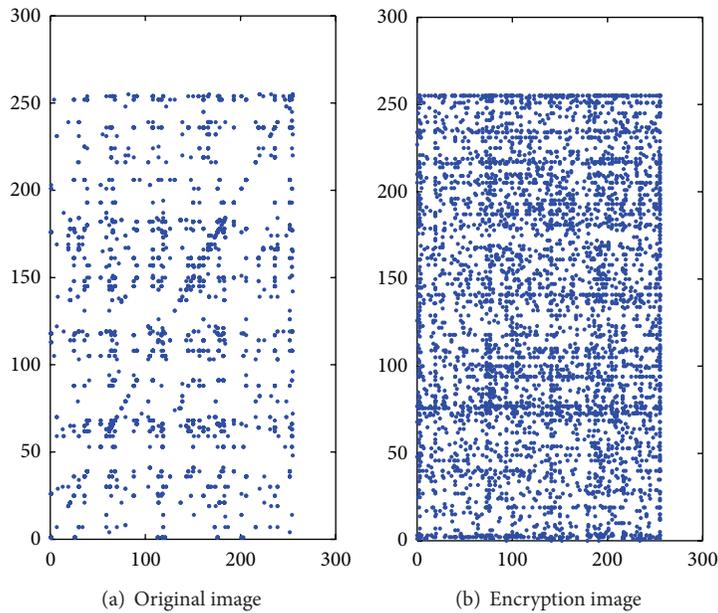


FIGURE 8: The correlation analysis of original and encryption image in vertical.

$$\text{COV}(x, y) = \frac{1}{N} \sum_{i=1}^N [x_i - E(x)] [y_i - E(y)],$$

$$r_{xy} = \frac{\text{COV}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}},$$

(27)

where $E(x)$ is the expected value and N is the number of pixels. $D(x)$ is the estimation of variance of x , and $\text{COV}(x, y)$ is the estimation of covariance between x and y , where x and y are grayscale values of two adjacent pixels in the image. The correlation of two adjacent (in horizontal,

vertical, and diagonal direction) pixels from the original and encryption image is shown in Figures 7, 8, and 9. Table 1 gives the correlation coefficients of the original image and the encrypted image. The experiment data show that there is a high correlation between the adjacent pixels of the original image. The correlation coefficients of the encrypted image are almost zero, and the adjacent pixels of the encrypted image are almost irrelevant.

5.4. Differential Analysis. In image encryption, the cipher resistance to differential attacks is commonly analyzed via

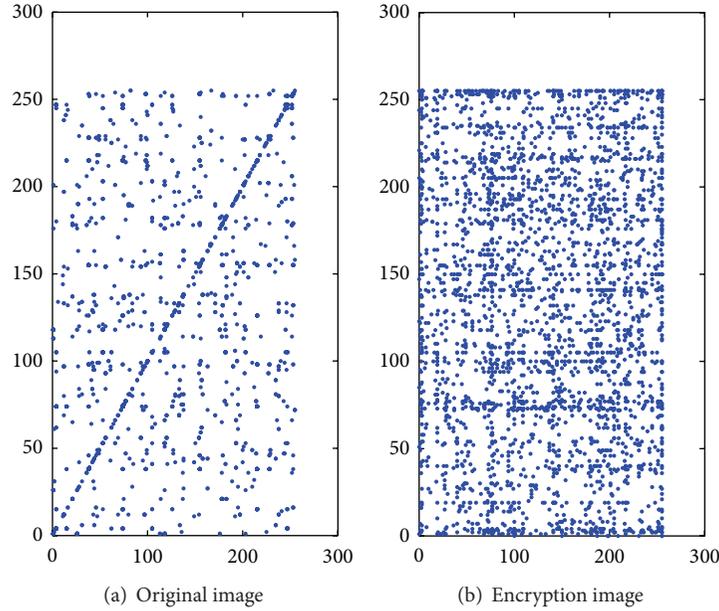


FIGURE 9: The correlation analysis of original and encryption image in diagonal.

TABLE 1: The correlation coefficients of the original image and the encrypted image.

	Horizontal	Vertical	Diagonal
Original image	0.9998	0.9506	0.9469
Encrypted image	0.0015	0.0096	-0.0140

the Number of Pixels Change Rate (NPCR) and the Unified Average Changing Intensity (UACI) tests [31]. NPCR and UACI can be mathematically defined by

$$\begin{aligned}
 D(i, j) &= \begin{cases} 0, & \text{if } C^1(i, j) = C^2(i, j) \\ 1, & \text{if } C^1(i, j) \neq C^2(i, j) \end{cases} \\
 \text{NPCR} &= \frac{\sum_{i=1}^M \sum_{j=1}^N D(i, j)}{M \times N} \times 100\% \\
 \text{UACI} &= \frac{\sum_{i=1}^M \sum_{j=1}^N ((C^1(i, j) - C^2(i, j)) / 255)}{M \times N} \times 100\%, \quad (28)
 \end{aligned}$$

where C^1 , C^2 are ciphertext images before and after one pixel change in a plaintext image, respectively. $C^1(i, j)$ are the pixel value at grid (i, j) in C^1 . $C^2(i, j)$ are the pixel value at grid (i, j) in C^2 . It is clear that NPCR concentrates on the absolute number of pixels which changes value in differential attacks, while the UACI focuses on the averaged difference between two paired ciphertext images [32]. In this paper, the NPCR between the original image and the encryption image is 0.00132%, and the UACI is 0.0000005%. The results demonstrate that the proposed scheme can survive differential attack.

5.5. Key Sensitivity Test. An ideal image encryption scheme has to be key-sensitive, meaning that a tiny change in the key will produce completely different encrypted image. For testing the key sensitivity of encryption algorithm, this paper has performed sensitivity analysis according to the following steps [33].

(1) An original image in Figure 5(a) is encrypted by using TD-ERCS system seed parameters (0.1256, 0.8130, 0.5325, and 2) and the resultant image is referred to as encrypted image A as shown in Figure 5(b).

(2) The same original image is encrypted by making the slight modification in the seed parameters (0.12559, 0.8130, 0.5325, and 2). The encrypted image is shown in Figure 10(a) and the decrypted image of image A using seed parameters (0.12559, 0.8130, 0.5325, and 2) is shown in Figure 9(b).

(3) The same original image is encrypted by making slight modification in the seed parameters (0.1256, 0.81299, 0.5325, and 2) and the encrypted image is shown in Figure 10(c). Figure 10(d) has shown the decrypted image of image A using seed parameters (0.1256, 0.81299, 0.5325, and 2).

(4) The same original image is encrypted by making the slight modification in the seed parameters (0.1256, 0.8130, 0.53249, and 2) and the encrypted image is shown in Figure 10(e). Figure 10(f) has shown the decrypted image of image A using seed parameters (0.1256, 0.8130, 0.53249, and 2).

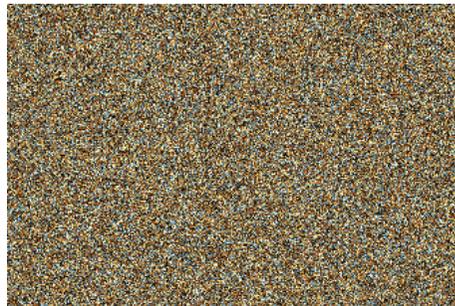
Figure 10 clearly shows that the image A is not correctly decrypted by using TD-ERCS system seed parameters (0.12559, 0.8130, 0.5325, and 2), (0.1256, 0.81299, 0.5325, and 2), and (0.1256, 0.8130, 0.53249, and 2), which has also only one bit difference between the correct key and the wrong key.



(a) The encrypted image using seed parameters (0.1256, 0.8130, 0.5325, and 2)



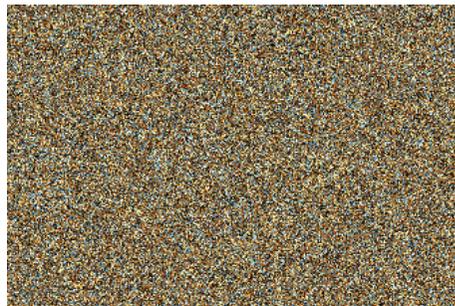
(b) The decrypted image of Figure 5(b) using seed parameters (0.12559, 0.8130, 0.5325, and 2)



(c) The encrypted image using seed parameters (0.1256, 0.81299, 0.5325, and 2)



(d) The decrypted image of Figure 5(b) using seed parameters (0.1256, 0.81299, 0.5325, and 2)



(e) The encrypted image using seed parameters (0.1256, 0.8130, 0.53249, and 2)



(f) The decrypted image of Figure 5(b) using seed parameters (0.1256, 0.8130, 0.53249, and 2)

FIGURE 10: Key sensitive test.

6. Conclusion

The TD-ERCS system is a highly complex nonlinear dynamic system based on the physical model of ellipse reflecting cavity. In this paper, an image encryption scheme based on TD-ERCS system and wavelet neural networks is presented. All parts of the proposed encryption algorithm were simulated using computer code. Theoretical and experimental results indicate that the image encryption algorithm based on TD-ERCS system and wavelet neural networks has advantages of large key space and high-level security, while maintaining acceptable efficiency. Finally, the proposed image encryption is suitable for any size digital image and can be widely applied in other information security fields.

Conflict of Interests

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

Acknowledgment

This research was partially supported by the National Natural Science Foundation of China (no. 11261001).

References

- [1] F. Chiaraluce, L. Ciccarelli, E. Gambi, P. Pierleoni, and M. Reginelli, "A new chaotic algorithm for video encryption," *IEEE Transactions on Consumer Electronics*, vol. 48, no. 4, pp. 838–844, 2002.
- [2] Y. Huang, R. Xu, and W. Lin, "An algorithm for JPEG compressing with chaotic encrypting," in *Proceedings of the International Conference on Computer Graphics, Imaging and Visualisation (CGIV '06)*, pp. 137–140, IEEE, July 2006.
- [3] H. Bai-Lin, *Starting with Parabolas: An Introduction to Chaotic Dynamics*, Shanghai Scientific and Technological Education Publishing House, Shanghai, China, 1993, (Chinese).

- [4] L. Kocarev, M. Sterjev, A. Fekete, and G. Vattay, "Public-key encryption with chaos," *Chaos*, vol. 14, no. 4, pp. 1078–1082, 2004.
- [5] R. Matthews, "On the derivation of a "chaotic" encryption algorithm," *Cryptologia*, vol. 13, no. 1, pp. 29–42, 1989.
- [6] F. Dachselt and W. Schwarz, "Chaos and cryptography," *IEEE Transactions on Circuits and Systems. I. Fundamental Theory and Applications*, vol. 48, no. 12, pp. 1498–1509, 2001.
- [7] G. Alvarez and S. Li, "Breaking an encryption scheme based on chaotic baker map," *Physics Letters A*, vol. 352, no. 1-2, pp. 78–82, 2006.
- [8] S. Ke-Hui, T. Guo-Qiang, and S. Li-Yua, "The complexity analysis of TD-ERCS discrete chaotic pseudo-random sequence," *Acta Physica Sinica*, vol. 57, pp. 3359–3366, 2008.
- [9] L.-Y. Sheng, G.-Q. Li, and Z.-W. Li, "One-way Hash function construction based on tangent-delay ellipse reflecting cavity-map system," *Acta Physica Sinica*, vol. 55, no. 11, pp. 5700–5706, 2006.
- [10] N. F. Güler, E. D. Übeyli, and I. Güler, "Recurrent neural networks employing Lyapunov exponents for EEG signals classification," *Expert Systems with Applications*, vol. 29, no. 3, pp. 506–514, 2005.
- [11] L. Chen and K. Aihara, "Chaotic simulated annealing by a neural network model with transient chaos," *Neural Networks*, vol. 8, no. 6, pp. 915–930, 1995.
- [12] L. Chen and K. Aihara, "Strange attractors in chaotic neural networks," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 47, no. 10, pp. 1455–1468, 2000.
- [13] S. Li-Yuan, S. Ke-Hui, and L. Chuan-Bing, "Study of a discrete chaotic system based on tangent-delay for elliptic reflecting cavity and its properties," *Acta Physica Sinica*, vol. 53, no. 9, pp. 2871–2876, 2004.
- [14] L.-Y. Sheng, L.-L. Cao, K.-H. Sun, and J. Wen, "Pseudo-random number generator based on TD-ERCS chaos and its statistic characteristics analysis," *Acta Physica Sinica*, vol. 54, no. 9, pp. 4031–4037, 2005.
- [15] R. Q. Quiroga and H. Garcia, "Single-trial event-related potentials with wavelet denoising," *Clinical Neurophysiology*, vol. 114, no. 2, pp. 376–390, 2003.
- [16] Y. C. Pati and P. S. Krishnaprasad, "Analysis and synthesis of feedforward neural networks using discrete affine wavelet transformations," *IEEE Transactions on Neural Networks*, vol. 4, no. 1, pp. 73–85, 1993.
- [17] A. Subasi, M. Yilmaz, and H. R. Ozcalik, "Classification of EMG signals using wavelet neural network," *Journal of Neuroscience Methods*, vol. 156, no. 1-2, pp. 360–367, 2006.
- [18] N. Chauhan, V. Ravi, and D. Karthik Chandra, "Differential evolution trained wavelet neural networks: application to bankruptcy prediction in banks," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7659–7665, 2009.
- [19] A. Grossmann and J. Morlet, "Decomposition of Hardy functions into square integrable wavelets of constant shape," *SIAM Journal on Mathematical Analysis*, vol. 15, no. 4, pp. 723–736, 1984.
- [20] Q. Zhao, Z. Sun, F. Sun, and J. Zhu, "Appearance-based robot visual servo via a wavelet neural network," *International Journal of Control, Automation and Systems*, vol. 6, no. 4, pp. 607–612, 2008.
- [21] A. Mitra, Y. V. S. Rao, and S. R. M. Prasanna, "A new image encryption approach using combinational permutation techniques," *World Academy of Science, Engineering and Technology*, vol. 2, pp. 831–835, 2008.
- [22] H. Gao, Y. Zhang, S. Liang, and D. Li, "A new chaotic algorithm for image encryption," *Chaos, Solitons and Fractals*, vol. 29, no. 2, pp. 393–399, 2006.
- [23] K. Zhang and X.-A. Fu, "Color image encryption algorithm based on tangent-delay ellipse reflecting cavity map system," in *Proceedings of the International Conference on Image Analysis and Signal Processing*, pp. 25–27, 2012.
- [24] C. Dellago and H. A. Posch, "Lyapunov exponents of systems with elastic hard collisions," *Physical Review E*, vol. 52, no. 3, pp. 2402–2406, 1995.
- [25] A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, "Determining Lyapunov exponents from a time series," *Physica D: Nonlinear Phenomena*, vol. 16, no. 3, pp. 285–317, 1985.
- [26] T. Gao and Z. Chen, "A new image encryption algorithm based on hyper-chaos," *Physics Letters A*, vol. 372, no. 4, pp. 394–400, 2008.
- [27] Y. Mao, G. Chen, and S. Lian, "A novel fast image encryption scheme based on 3D chaotic Baker maps," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 14, no. 10, pp. 3613–3624, 2004.
- [28] A. N. Pisarchik and M. Zanin, "Image encryption with chaotically coupled chaotic maps," *Physica D: Nonlinear Phenomena*, vol. 237, no. 20, pp. 2638–2648, 2008.
- [29] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," *Chaos, Solitons & Fractals*, vol. 21, no. 3, pp. 749–761, 2004.
- [30] S. Behnia, A. Akhshani, S. Ahadpour, H. Mahmodi, and A. Akhavan, "A fast chaotic encryption scheme based on piecewise nonlinear chaotic maps," *Physics Letters iA*, vol. 366, no. 4-5, pp. 391–396, 2007.
- [31] S. Behnia, A. Akhshani, H. Mahmodi, and A. Akhavan, "A novel algorithm for image encryption based on mixture of chaotic maps," *Chaos, Solitons & Fractals*, vol. 35, no. 2, pp. 408–419, 2008.
- [32] Y. Wu, J. P. Noonan, and S. Agaian, "NPCR and UACI randomness tests for image encryption," *Cyber Journals*, pp. 31–38, 2011.
- [33] G. Chen, Y. Mao, and C. K. Chui, "A symmetric image encryption scheme based on 3D chaotic cat maps," *Chaos, Solitons & Fractals*, vol. 21, no. 3, pp. 749–761, 2004.

Research Article

A Study on Many-Objective Optimization Using the Kriging-Surrogate-Based Evolutionary Algorithm Maximizing Expected Hypervolume Improvement

Chang Luo, Koji Shimoyama, and Shigeru Obayashi

Institute of Fluid Science, Tohoku University, Sendai 980-8577, Japan

Correspondence should be addressed to Chang Luo; luochang33@163.com

Received 25 August 2014; Revised 13 January 2015; Accepted 13 January 2015

Academic Editor: Yudong Zhang

Copyright © 2015 Chang Luo 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.

The many-objective optimization performance of the Kriging-surrogate-based evolutionary algorithm (EA), which maximizes expected hypervolume improvement (EHVI) for updating the Kriging model, is investigated and compared with those using expected improvement (EI) and estimation (EST) updating criteria in this paper. Numerical experiments are conducted in 3- to 15-objective DTLZ1-7 problems. In the experiments, an exact hypervolume calculating algorithm is used for the problems with less than six objectives. On the other hand, an approximate hypervolume calculating algorithm based on Monte Carlo sampling is adopted for the problems with more objectives. The results indicate that, in the unconstrained case, EHVI is a highly competitive updating criterion for the Kriging model and EA based many-objective optimization, especially when the test problem is complex and the number of objectives or design variables is large.

1. Introduction

Evolutionary algorithms (EAs) are a class of metaheuristics inspired by the process of natural evolution, and they have been successfully applied to solve the optimization problems with two or three objectives over the past two decades [1]. Recently, EAs are gradually extended to deal with many-objective problems that involve four or more objectives. Garza-Fabre et al. [2] combined three novel fitness assignment methods with a generic multiobjective evolutionary algorithm (MOEA) and conducted the numerical experiments in 5- to 50-objective DTLZ1, DTLZ3, and DTLZ6 problems [3]. It was found that the proposed three methods were effective in guiding the search in high-dimensional objective spaces. Adra and Fleming [4] investigated two new management mechanisms for promoting diversity in evolutionary many-objective optimization and tested 6- to 20-objective DTLZ2 problems. The results indicated that the inclusion of one of the mechanisms improved the convergence and diversity performances of the existing MOEAs in solving many-objective optimization problems. Hadka and Reed [5] proposed the Borg MOEA for the many-objective

and multimodal optimization using the ϵ -dominance concept and an adaptive population sizing approach. A comparative study was conducted on 18 test problems from DTLZ [3], WFG [6], and CEC 2009 [7] test suites, and the Borg MOEA showed significant advantages over the competing algorithms on many-objective and multimodal problems. Lopez et al. [8] coupled an achievement function and the ϵ -indicator using an alternative preference relation in a MOEA for solving many-objective optimization problems. The experiments were conducted on DTLZ and WFG test suites. The results showed that the proposed approach had a good performance even when using a high number of objectives. Yang et al. [9] proposed a grid-based evolutionary algorithm (GrEA) to solve many-objective optimization problems and performed the numerical experiments in DTLZ test suite. The experimental results indicated that the proposed algorithm showed the effectiveness and competitiveness in balancing convergence and diversity compared with six state-of-the-art MOEAs. Deb and Jain [10] proposed a reference-point based many-objective NSGA-II (called NSGA-III). This algorithm was applied to 3- to 15-objective DTLZ1-4 problems and

compared with two recently suggested versions of MOEA/D. It was found that the proposed NSGA-III obtained satisfactory results on all test problems.

However, the real-world applications of the conventional EA based optimization approaches need high CPU cost due to a large number of expensive performance analyses such as three-dimensional computational fluid dynamics for complex geometries. A common strategy to reduce their CPU cost is to combine surrogate model with EA. A number of surrogate models such as response surface model (RSM) [11], radial basis function (RBF) [12], Kriging model [13], and neural network (NN) [14] have been applied to practical engineering designs. Among these surrogate models, the Kriging model can estimate the deviation between the response model and sample points and automatically adapt to the sample points. Additionally, the Kriging model has a characteristic that an assumption of order of the approximate function is not needed, so it is superior to general RSM. In this study, the Kriging model is adopted as the surrogate model.

For the Kriging model, a few updating criteria have been proposed to determine the locations in the design space where new sample points should be added for improving the model accuracy. Jones et al. [15] suggested that the expected improvement (EI) of an original objective function was maximized to determine the location of a new additional sample point and designated the efficient global optimization algorithm (EGO). Jeong et al. [16] extended EGO for multiobjective optimization problems, in which the EIs in terms of all objective functions were maximized and some of the nondominated solutions in the EIs space were selected as the additional sample points. Emmerich et al. [17] proposed the expected hypervolume improvement (EHVI) as the updating criterion of the Gaussian random field metamodel and evaluated EHVI using the Monte Carlo integration method. Li et al. [18] suggested a domination status based updating criterion to judge whether new sample point was needed in the Kriging metamodel assisted multiobjective genetic algorithm. Shimoyama et al. [19] compared the optimization performances of the EHVI, EI, and estimation (EST) updating criteria in the multiobjective optimization. The results indicated that EHVI kept a good balance between accurate and wide search for nondominated solutions.

However, the study of using EHVI as updating criterion for the optimization of many-objective problems that involve four or more objectives has not been found in the open literature. The present paper is concerned with the many-objective optimization performance of the Kriging-surrogate-based EA approach considering EHVI as the updating criterion. The long term objective is to reduce the CPU cost of the many-objective optimization approach by decreasing the number of expensive performance analyses. In order to further investigate the performance of EHVI updating criterion, two other updating criteria EI and EST are adopted for comparisons. The present comparisons are conducted through numerical experiments in DTLZ test suite [3], which can be extended to an arbitrary number of objective functions.

The structure of this paper is as follows. In Section 2, the terminology and background related to the current study

are reviewed. In Section 3, the Kriging model and EA based approach are outlined. In Section 4, several examples and corresponding results are given. Concluding remarks are presented in Section 5.

2. Background

2.1. Many-Objective Optimization. First, a many-objective problem with M objectives is defined as follows:

$$\begin{aligned} \text{minimize } \mathbf{F}(\mathbf{x}) &= (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})) \\ \text{subject to } \mathbf{x} &\in X, \end{aligned} \quad (1)$$

where \mathbf{x} is a vector of n design variables, $\mathbf{F}(\mathbf{x})$ is a vector of M objective functions, $f_i(\mathbf{x})$ denotes the i th objective function to be minimized, and X is the feasible region delimited by the problem's constraints.

In the many-objective problems, the goal is to find the optimal tradeoff solutions known as the Pareto optimal set. The definition of the Pareto optimal set is based on the domination concept between points in the design space. The dominance relation between two design vectors \mathbf{x}_1 and \mathbf{x}_2 is defined as follows.

Definition 1. A design vector \mathbf{x}_1 dominates another design vector \mathbf{x}_2 , if $f_i(\mathbf{x}_1) \leq f_i(\mathbf{x}_2)$ for all $i \in \{1, \dots, n\}$ and $f_j(\mathbf{x}_1) < f_j(\mathbf{x}_2)$ for at least one $j \in \{1, \dots, n\}$. This is denoted by $\mathbf{x}_1 < \mathbf{x}_2$.

Definition 2. $\mathbf{x} \in X$ is nondominated if and only if there is no vector in X that dominates \mathbf{x} .

A design vector \mathbf{x} is Pareto optimal if the design vector \mathbf{x} is nondominated with respect to the set of all possible design vectors. The set of all possible Pareto optimal is called Pareto optimal set, which forms the Pareto front in the objective space. In general, multiobjective optimization algorithms are designed to capture or closely approximate the Pareto front.

2.2. Kriging Model. The Kriging model has its original applications in mining and geostatistical fields referring to spatially and temporally correlated data [20]. The Kriging model is a combination of global model and localized departures as follows:

$$f(x) = \mu + Z(x), \quad (2)$$

where $f(x)$ denotes an unknown function of interest and μ denotes a known global approximation model. $Z(x)$ is a realization of a stochastic process with mean zero and variance σ^2 , and the covariance matrix of $Z(x)$ is given by

$$\text{Cov}[Z(x^i), Z(x^j)] = \sigma^2 \mathbf{R}, \quad \mathbf{R} = [R(x^i, x^j)]. \quad (3)$$

In (3), \mathbf{R} is an $(n_s \times n_s)$ correlation matrix whose entries are symmetric with respect to the diagonal, $R(x^i, x^j)$ is the correlation function between any two points x^i and x^j among n_s sample points. The type of correlation function needs to

be determined by users. This paper employs the following Gaussian correlation function:

$$R(x^i, x^j) = \exp \left[-\sum_{k=1}^n \theta_k |x_k^i - x_k^j|^2 \right], \quad (4)$$

where n denotes the number of design variables. θ_k is the weight of the distance along the k th design variable.

In the Kriging model, the values of μ , σ^2 , and $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_n]$ are determined by maximizing the likelihood function. First, $\boldsymbol{\theta}$ is obtained by maximizing the concentrated log-likelihood function as

$$\max_{\theta_k > 0} \left(-\frac{n_s}{2} \ln(\sigma^2) - \frac{1}{2} \ln(|\mathbf{R}|) \right), \quad k = 1, 2, \dots, n; \quad (5)$$

μ and σ^2 that maximize the likelihood function are represented in closed form as

$$\mu = \frac{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{f}}{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1}}, \quad (6)$$

$$\sigma^2 = \frac{(\mathbf{f} - \mathbf{1}\mu)^T \mathbf{R}^{-1} (\mathbf{f} - \mathbf{1}\mu)}{n_s}, \quad (7)$$

where $\mathbf{f} = [f(x^1), f(x^2), \dots, f(x^{n_s})]^T$ and $\mathbf{1}$ is an n_s -dimensional unit vector. After $\boldsymbol{\theta}$ is obtained, μ and σ^2 are obtained by (6) and (7), respectively. Thereinto, \mathbf{R} is calculated by (4). Now, (2) can be written to be the Kriging model predictor as

$$\hat{f}(x) = \mu + \mathbf{r}^T \mathbf{R}^{-1} (\mathbf{f} - \mathbf{1}\mu), \quad (8)$$

where \mathbf{r} is an n_s -dimensional vector whose i th element is $\text{Cov}[Z(x), Z(x^i)]$.

The accuracy of the predicted value $\hat{f}(x)$ depends greatly on the distances between predicted point x and the sample points. The mean squared error $\hat{s}^2(x)$ for a predicted point x using the Kriging model predictor is defined by

$$\hat{s}^2(x) = \sigma^2 \left[\mathbf{1} - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} + \frac{(\mathbf{1} - \mathbf{1}^T \mathbf{R}^{-1} \mathbf{r})^2}{\mathbf{1}^T \mathbf{R}^{-1} \mathbf{1}} \right]. \quad (9)$$

2.3. Expected Hypervolume Improvement. EHVI is based on the theory of the hypervolume indicator [21]. Hypervolume indicator is a recently popular metric which is used for comparing the performances of different multiobjective optimizers. The hypervolume of a set of solutions S measures the size of the portion of objective space that is dominated by the set S collectively. In the field of evolutionary multiobjective optimization (EMO), the hypervolume indicator is the only unary indicator that is known to be strictly monotonic with regard to Pareto dominance. This characteristic is of high interest and relevance for the problems with a large number of objectives.

Hypervolume calculation requires high computational effort. Several algorithms have been proposed for calculating hypervolume exactly. Wu and Azarm [22] proposed

the inclusion-exclusion algorithm (IEA) for hypervolume calculation, and the complexity of this algorithm is $O(M2^N)$ for N solutions and M objectives. Fleischer [23] introduced the algorithm based on the Lebesgue measure, and its complexity is $O(N^M)$. While et al. [24] suggested a fast hypervolume by slicing objective (HSO) algorithm, and the complexity is $O(N^{M-1})$. Based on HSO, Fonseca et al. [25] proposed an improved dimension-sweep algorithm for calculating hypervolume. The proposed algorithm achieved $O(N^{M-2} \log N)$ complexity in the worst case. The fastest algorithm yet known for exact hypervolume calculation is the Walking Fish Group (WFG) algorithm proposed by While et al. [26]. Its complexity is $O(2^N - 1)$ in the worst case; however, even relatively small percentages of dominated points can make this algorithm's performance a huge improvement.

On the other hand, some approximate algorithms to calculate hypervolume have also been developed in recent years. Bader and Zitzler [27] proposed the approximate algorithm based on Monte Carlo sampling. In their work, the approximate algorithm was adopted to calculate the hypervolume values of the problems with more than 5 objectives. Bringmann and Friedrich [28] presented a new approximation algorithm also based on Monte Carlo sampling and worked extremely fast for all tested practical instances. Ishibuchi et al. [29] proposed the approximation algorithm using achievement scalarizing functions with uniformly distributed weight vectors.

EHVI is the expected value of hypervolume improvement in the Kriging model. The hypervolume improvement $\text{HVI}[f_1(x), f_2(x), \dots, f_M(x)]$ is defined as the difference of hypervolume between the current sample set and the next sample set, as illustrated in Figure 1, and its expected value $\text{EHVI}[f_1(x), f_2(x), \dots, f_M(x)]$ is expressed as

$$\begin{aligned} \text{EHVI}[f_1(x), f_2(x), \dots, f_M(x)] \\ = \int_{-\infty}^{f_{1\text{ref}}} \int_{-\infty}^{f_{2\text{ref}}} \dots \int_{-\infty}^{f_{M\text{ref}}} \text{HVI}[f_1(x), f_2(x), \dots, f_M(x)] \\ \times \phi_1(F_1) \\ \cdot \phi_2(F_2) \dots \phi_M(F_M) dF_1 dF_2 \dots dF_M, \end{aligned} \quad (10)$$

where F_i denotes the Gaussian random variable $N[\hat{f}_i(x), \hat{s}_i^2(x)]$. $\phi_i(F_i)$ is the probability density function, and $f_{i\text{ref}}$ is the reference value used for calculating hypervolume. The maximization of EHVI is considered as the updating criterion to determine the location of an additional sample point. It is a single objective optimization problem.

In the present optimizations, objective function values are normalized between 0 and 1 for all given sample points, and the reference values are set to be 1.1 for all normalized objective functions. When the number of objectives is small (≤ 5), hypervolume is calculated using the fastest exact algorithm WFG [26]. When the number of objectives is large (> 5), in consideration of the tradeoff between the EHVI accuracy and the overall computing time, hypervolume is calculated using the approximate algorithm as described in [27]. The

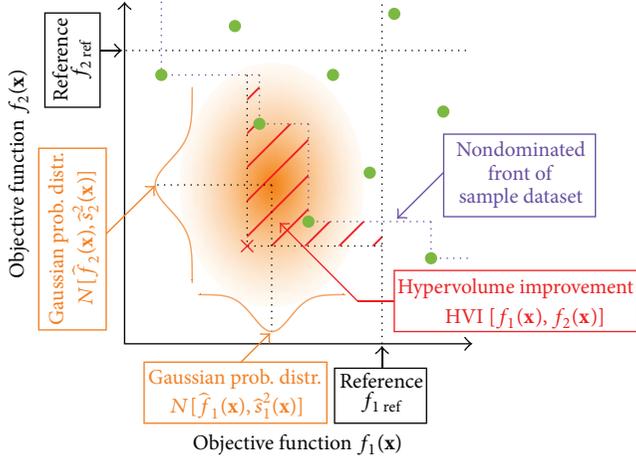


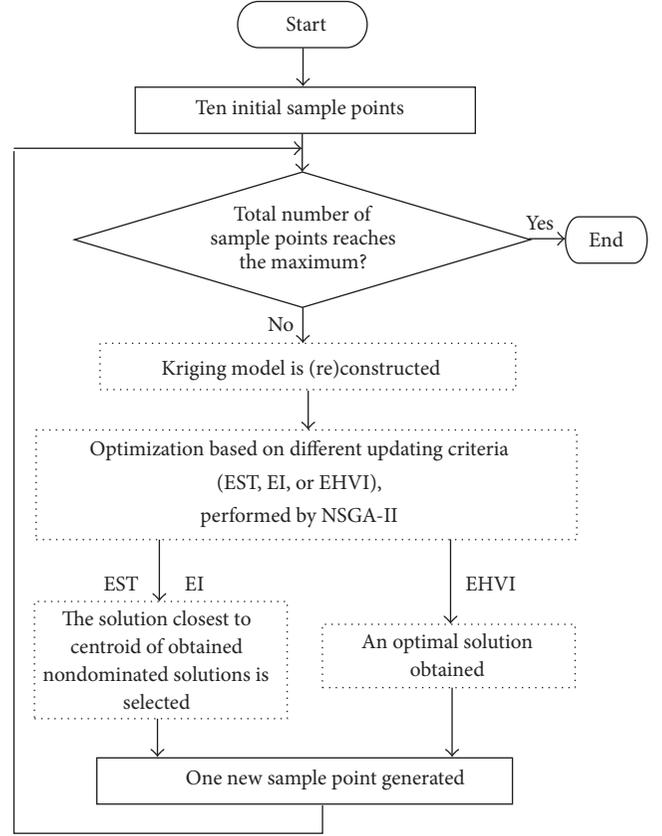
FIGURE 1: EHV1 updating criterion of the Kriging model.

number of samples to estimate an approximated value of hypervolume in this algorithm is set to be 300 for 8- and 10-objective problems and 500 for 12- and 15-objective problems. It is difficult to calculate EHV1 by the analytical integration. As a simple alternative, we use Monte Carlo integration [30] by producing 1,000 random points with Gaussian distribution around the candidate sample point to measure the EHV1.

3. Kriging-Surrogate-Based EA for Many-Objective Optimization

Figure 2 gives the flowchart of the Kriging-surrogate-based EA for many-objective optimization. The initial sample points are generated using Latin hypercube sampling (LHS) method [31] and distribute uniformly in the design space. LHS determines the locations of the points to be sampled taking account of the orthogonality condition, which means that each of the samples does not have overlapping values regarding all the design variables, and LHS can comprehend the whole variable space even with a smaller sample size than the Monte Carlo method. Using the initial samples, the Kriging model is constructed to approximate the responses of objective functions to design variables in the form of simple algebraic functions. These algebraic functions are derived to interpolate the initial sample points with real objective function values. Therefore, the Kriging model can promptly give the estimation function values at other points with unknown objective function values. In order to investigate the performance of EHV1, two other updating criteria EI and EST [19] are adopted for comparing with EHV1 in this study. An EMO algorithm, NSGA-II [32], is adopted to search for the nondominated solutions for EI or EST and the optimal solution for EHV1 on the response surfaces. The parameter values [33] used in NSGA-II are given in Table 1.

In the optimization based on EHV1, a single solution with the maximum EHV1 is obtained and employed as the additional sample point. This indicates that the sample points are added one by one. On the other hand, a set of



□ Real values
 □ Predicted values by Kriging model

FIGURE 2: Flowchart of the Kriging-surrogate-based EA for many-objective optimization.

TABLE 1: Parameter values used in NSGA-II.

Parameters	NSGA-II
Population size	100
Number of generations	200
Simulated binary crossover probability	1
Polynomial mutation probability	1/n
η_c [33]	30
η_m [33]	20

multiple nondominated solutions is obtained in the optimization based on EI or EST. For a fair comparison, only the solution closest to the centroid of the obtained nondominated solutions in the design space is chosen as the additional sample point. When the additional sample point determined in the previous step is added to the set of initial samples, the Kriging model is reconstructed and the optimization is performed again. The termination condition of updating the Kriging model is that the total number of the initial sample points plus the additional sample points reaches 70.

TABLE 2: The test problems and key properties.

Problems	M	n	Properties
DTLZ1	3, 5, 8, 10, 12, 15	$M + 4$	Multimodal, separable
DTLZ2	3, 5, 8, 10, 12, 15	$M + 9$	Concave, separable
DTLZ3	3, 5, 8, 10, 12, 15	$M + 9$	Multimodal, concave, and separable
DTLZ4	3, 5, 8, 10, 12, 15	$M + 9$	Concave, separable
DTLZ5	3, 5, 8, 10, 12, 15	$M + 9$	Concave, separable
DTLZ6	3, 5, 8, 10, 12, 15	$M + 9$	Concave, separable
DTLZ7	3, 5, 8, 10, 12, 15	$M + 19$	Discontinuous, separable
DTLZ7-k10	3, 5, 8, 10, 12, 15	$M + 9$	Discontinuous, separable

M is the number of objectives, and n is the number of variables.

TABLE 3: The size of T with the number of objectives.

M	3	5	8	10	12	15
$ T $	21^{M-1} 441	11^{M-1} 14,641	4^{M-1} 16,384	3^{M-1} 19,683	3^{M-1} 177,147	3^{M-1} 4,782,969

4. Numerical Test

4.1. Test Problems. Three- to fifteen-objective DTLZ1-7 problems [3] are considered in the present numerical experiments. All these problems are nonconstrained and scalable to an arbitrary number of objective functions. Table 2 lists these test problems and their key properties.

4.2. Performance Metric. The inverted generational distance (IGD) [34] is considered as the performance metric in the present numerical experiments. IGD can provide the combined information about the convergence and diversity of the obtained solutions. IGD is defined as follows:

$$\text{IGD}(T, D) = \frac{1}{|T|} \sum_{f \in T} \min_{f' \in D} |f - f'|, \quad (11)$$

where T denotes the set of the points located on the true Pareto optimal front. D is the nondominated solution set obtained by the optimization. Smaller IGD indicates better performance in multiobjective optimization. The solutions in T distribute uniformly on the true Pareto optimal front. The size of T varies with the number of objectives and is shown in Table 3.

4.3. Results and Discussion. For each test problem, 20 runs are carried out with different initial datasets of ten samples points. The IGD values that are given in the following are averaged over 20 runs. Figure 3 gives the IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ1 problems. EST achieves faster reduction of IGD than EHVI and EI except for the case of $M = 5$; EHVI achieves the fastest IGD reduction. In addition, EST always obtains faster reduction of IGD than EI for any number of objectives. It indicates that the updating criterion EST without considering estimation errors works

better than the updating criterion EI considering estimation errors. It is because DTLZ1 problem is simple and easily approximated by the Kriging model.

The IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ2 problems are shown in Figure 4. When $M = 3$ and 5, the optimization performance of using EHVI as the updating criterion is poor. This is because the updating strategy based on EHVI may result in the additional sample points falling into a local optimal region. When $M = 10$, EHVI obtains a little faster IGD reduction than EI and EST. In addition, it is noted that EI obtains much faster reduction of IGD than EHVI and EST when M is larger than 10.

Figure 5 presents the IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ3 problems. EHVI always achieves better convergence and diversity performances than EI and EST for any number of objectives. When $M = 3$, EI obtains faster IGD reduction than EST. When M is larger than 3, the final IGD values obtained by EI and EST are very close. It indicates that the problems are well approximated by the Kriging model, and the estimation errors are relatively small.

Figure 6 shows the IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ4 problems. For any number of objectives, EST without considering estimation errors obtains faster IGD reduction than EI considering estimation errors. As M increases, the convergence and diversity performances of using EHVI as updating criterion become better. When M is larger than 8, EHVI outperforms EI and EST. This indicates that EHVI updating criterion is particularly advantage for the problems with big number of objectives.

The IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ5 problems are shown in Figure 7. When M is smaller than 12, EST obtained faster reduction of IGD than EI. However, EI outperforms EST when $M = 12$ and 15. It indicates that EI considering estimation errors shows positive impact on convergence when the number of objectives is large. On the other hand, when $M = 3$ and 5, the optimization performance of EHVI is poor. However, the optimization performance based on EHVI is gradually improved as M increases. When M is larger than 8, EHVI obtains the fastest reduction of IGD.

Figure 8 presents the IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ6 problems. EHVI always obtains much faster IGD reduction than EI and EST for any number of objectives. The updating criterion EHVI shows the efficient exploitation capacity of the Kriging-surrogate-based EA for many-objective optimization. EI achieves faster reduction of IGD than EST except for the cases of $M = 3$ and 10.

The IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ7 problems are shown in Figure 9. EHVI outperforms EI and EST for any number of objectives. When M is larger than 8, EHVI shows more obvious advantage compared with EI and EST. On the other hand, EI has better optimization performance than EST except for the case of $M = 5$.

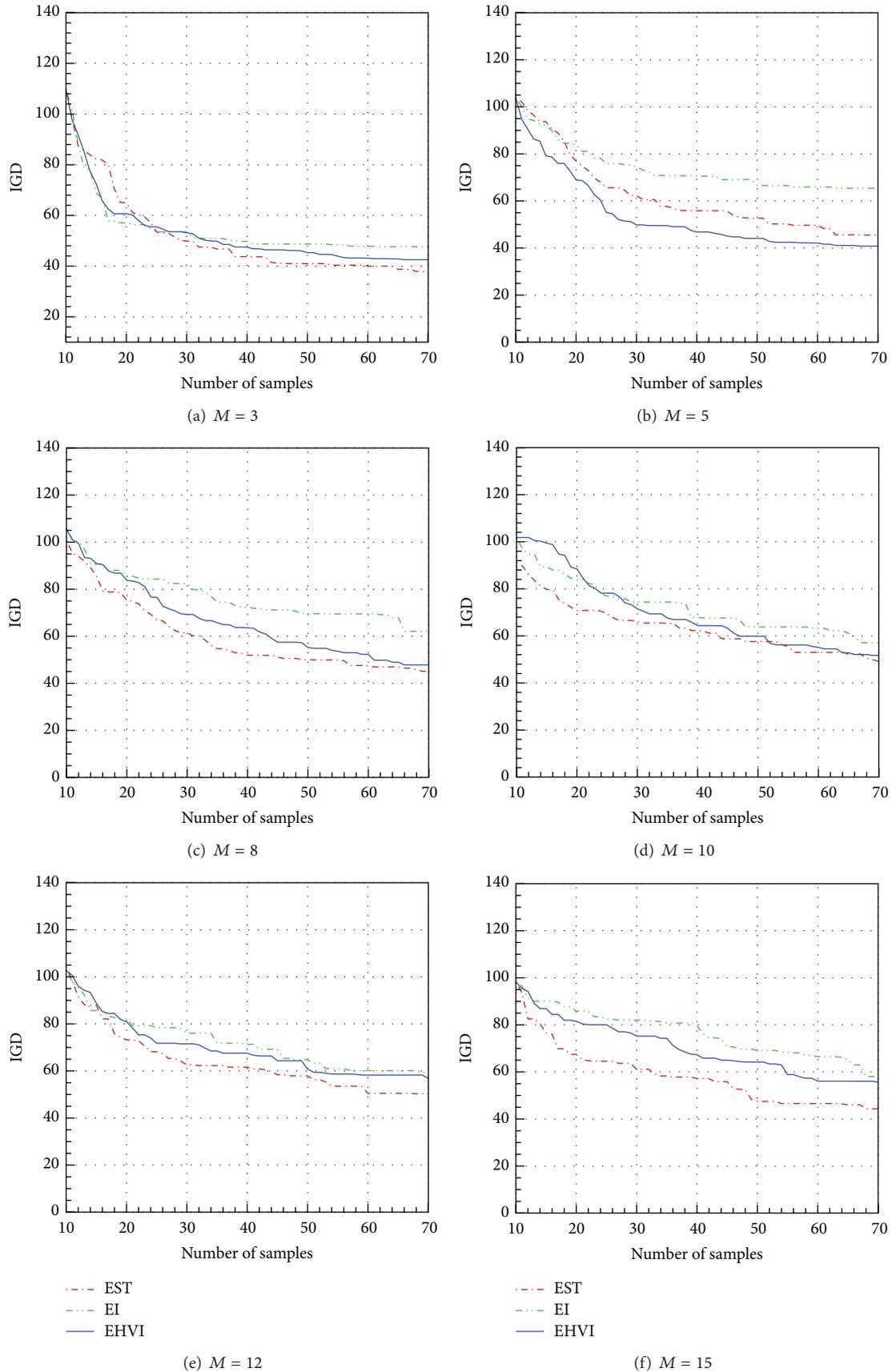


FIGURE 3: IGD histories during the Kriging model update in DTLZ1.

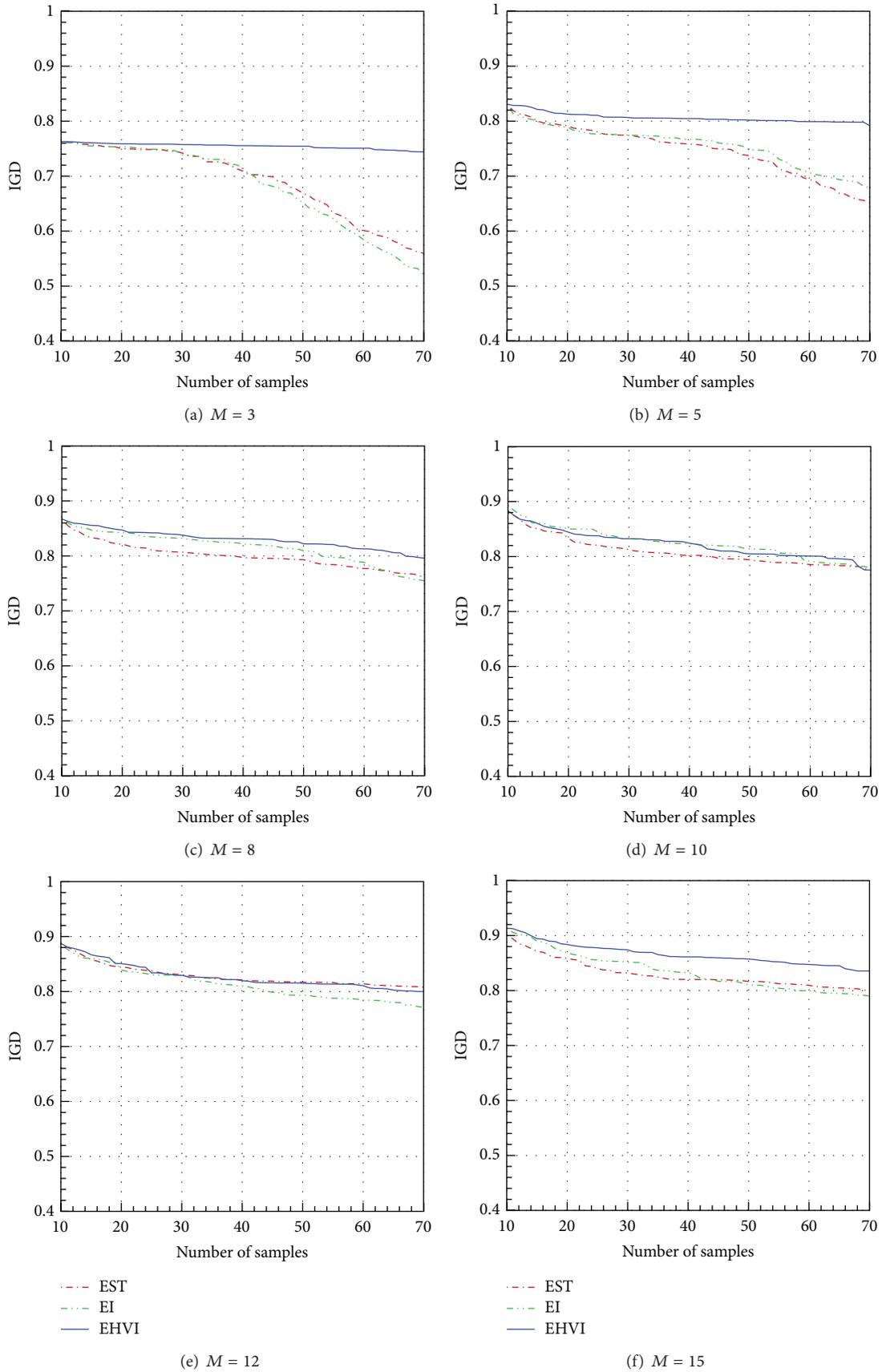


FIGURE 4: IGD histories during the Kriging model update in DTLZ2.

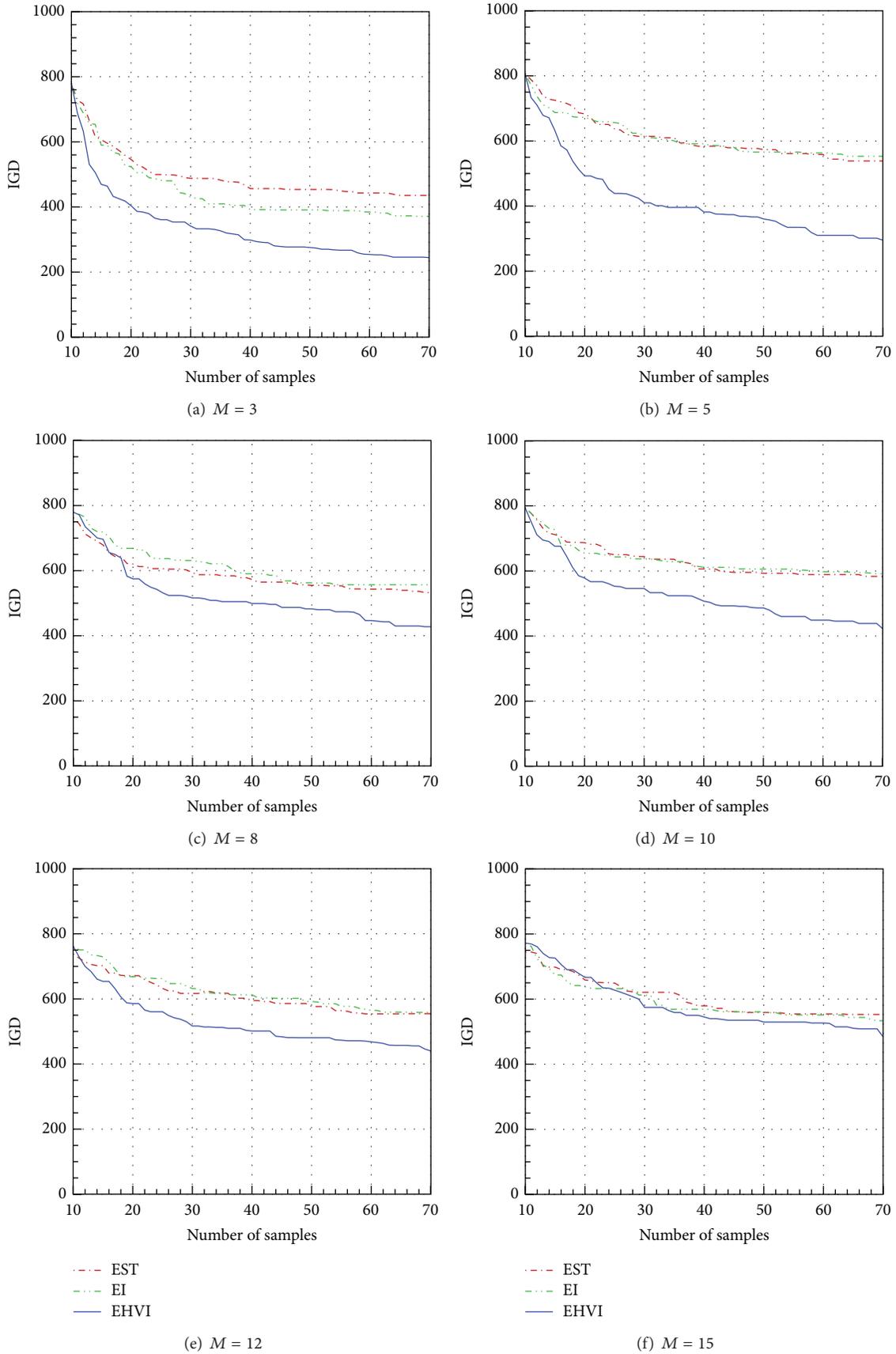


FIGURE 5: IGD histories during the Kriging model update in DTLZ3.

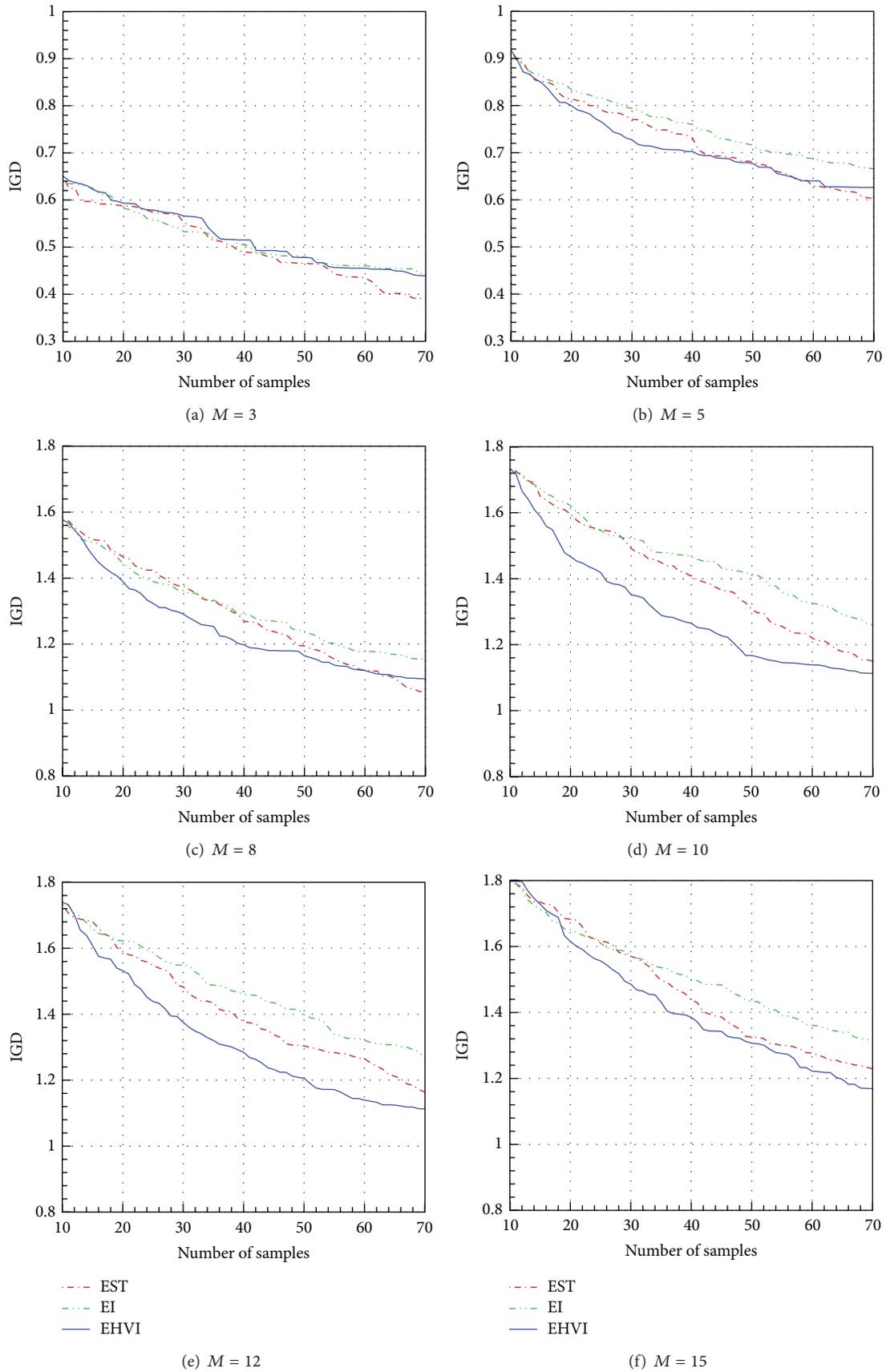


FIGURE 6: IGD histories during the Kriging model update in DTLZ4.

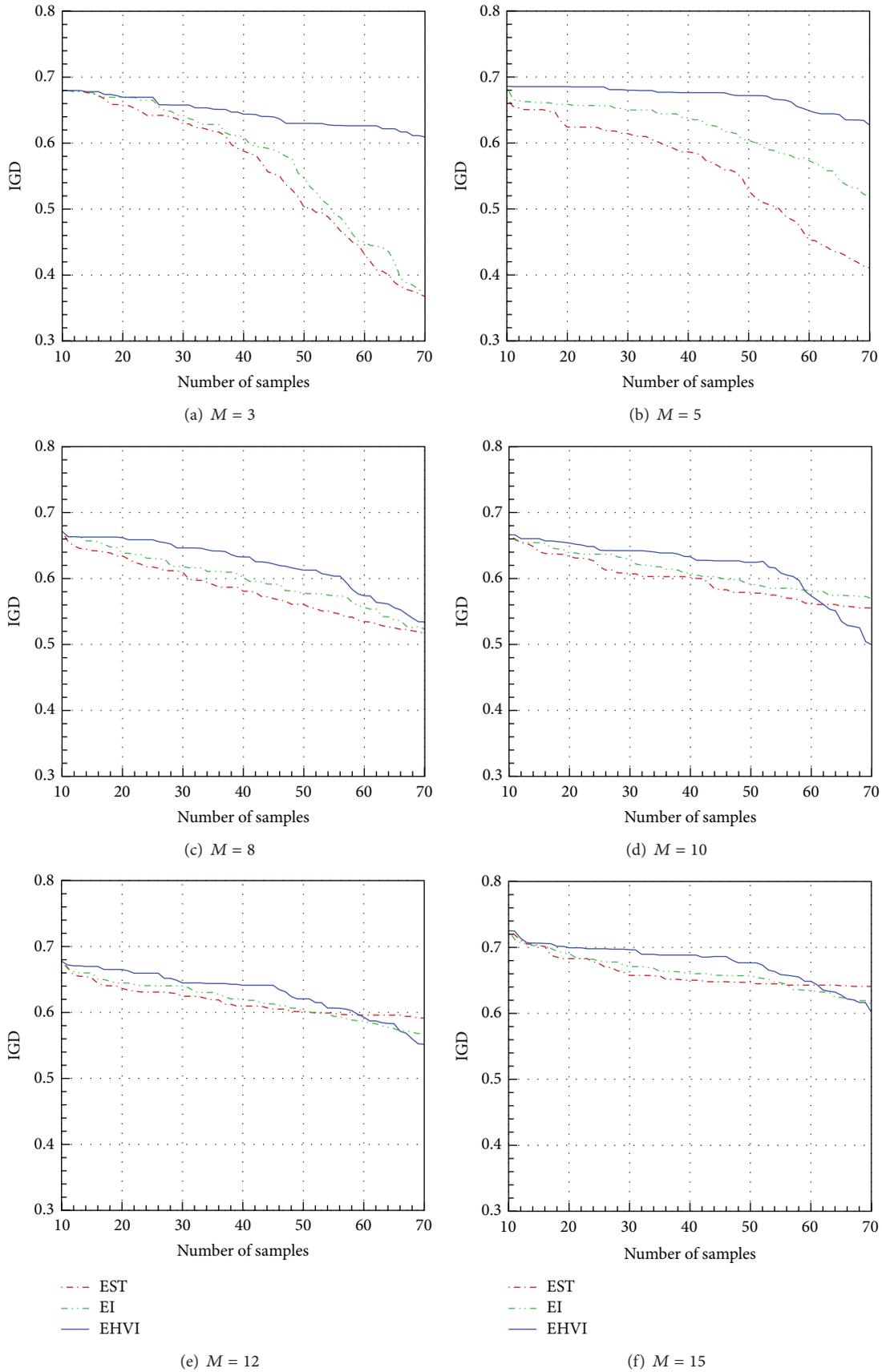


FIGURE 7: IGD histories during the Kriging model update in DTLZ5.

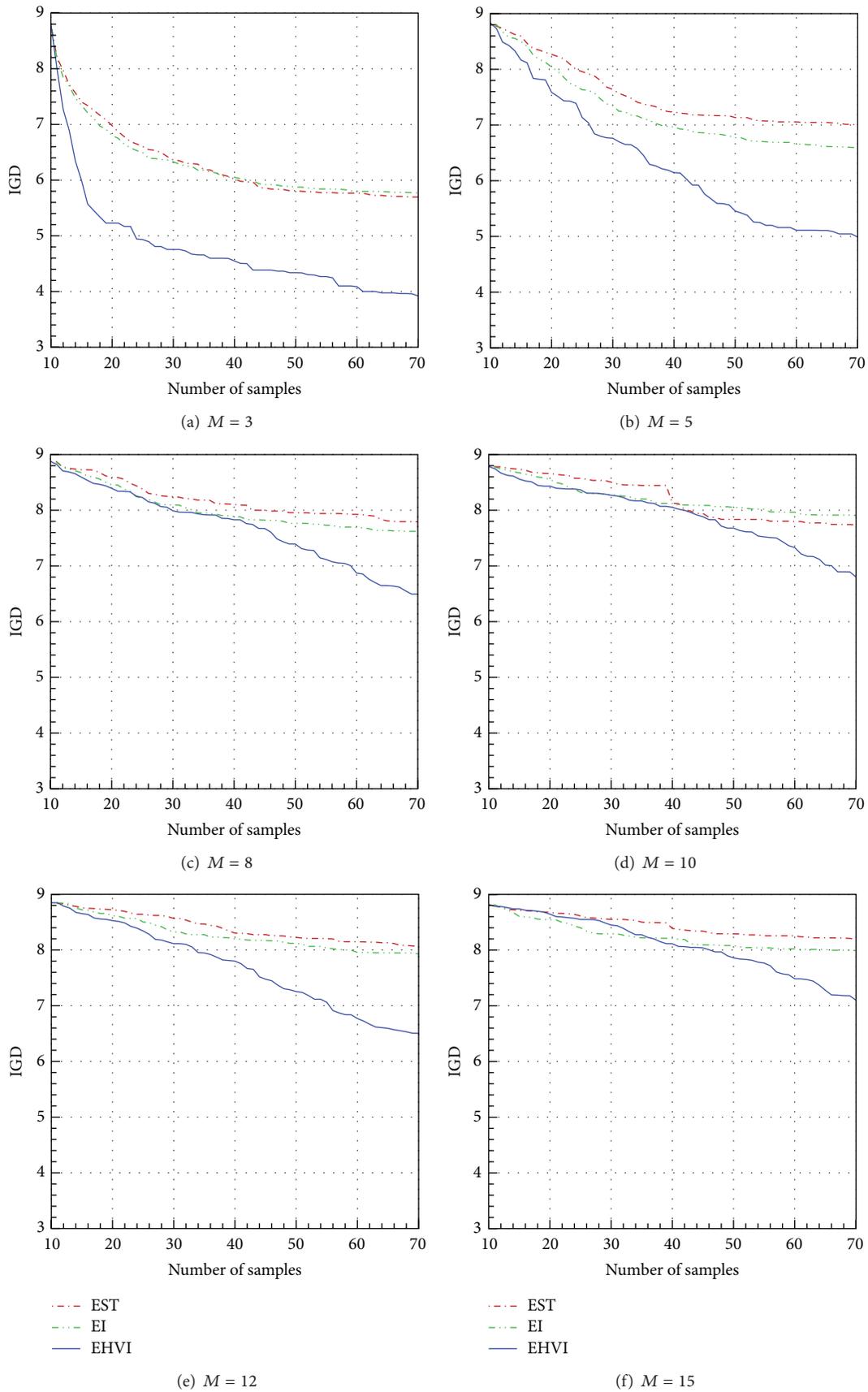


FIGURE 8: IGD histories during the Kriging model update in DTLZ6.

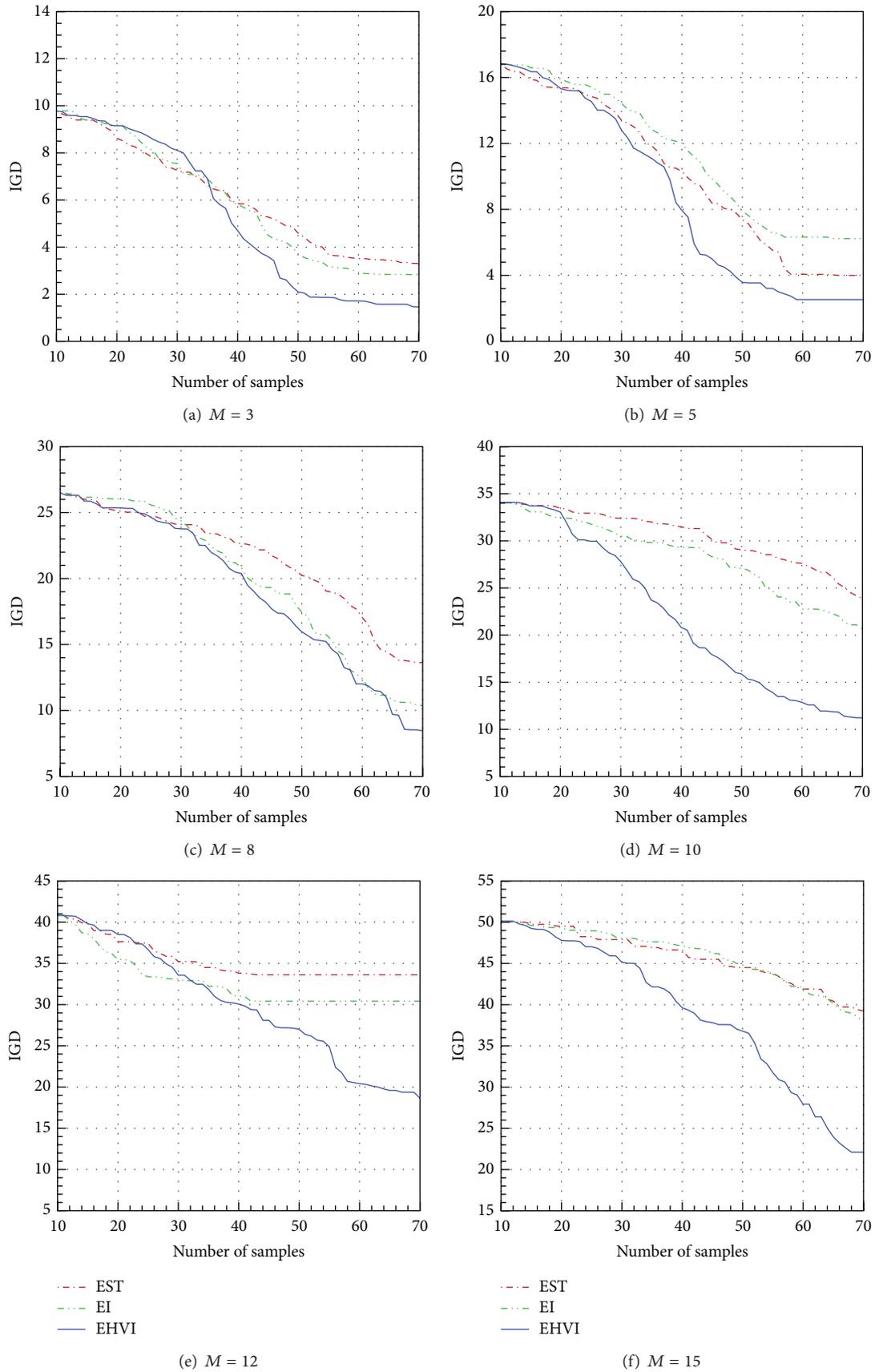


FIGURE 9: IGD histories during the Kriging model update in DTLZ7.

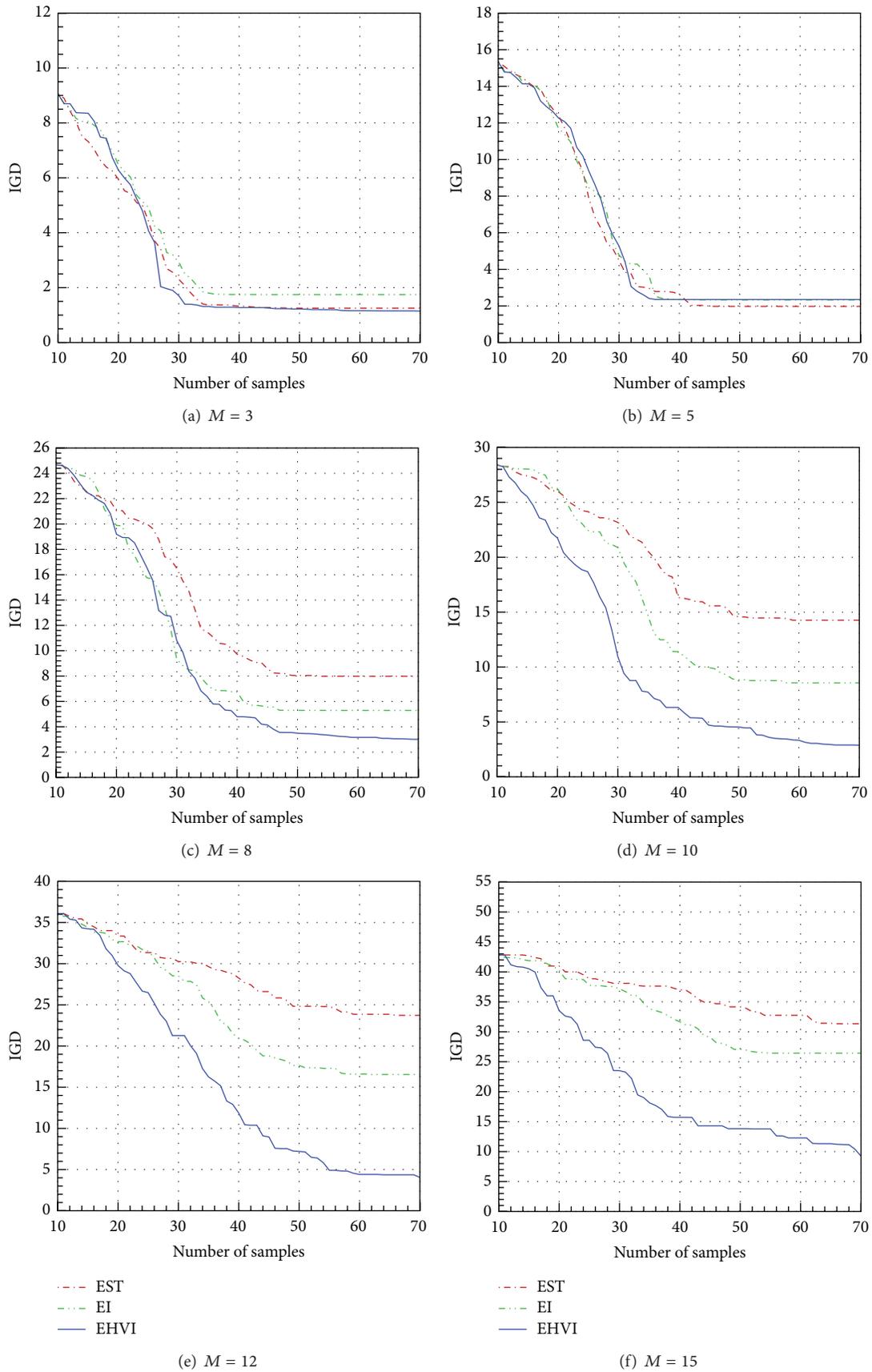


FIGURE 10: IGD histories during the Kriging model update in DTLZ7-k10.

In order to investigate the effect of the number of design variables on the optimization performance of using EHVI as updating criterion, DTLZ7- k 10 is conducted for the many-objective optimization. In the original DTLZ7 [15], the k , the number the required variables which must take any function that is not less than 0, is suggested to be 20. However, the k is set to be 10 in DTLZ7- k 10 problem. It indicates that the number of design variables of DTLZ7- k 10 problem is 10 less than that of the corresponding DTLZ7 problem. The IGD histories during the Kriging model update based on EHVI, EI, and EST criteria in 3- to 15-objective DTLZ7- k 10 problems are shown in Figure 10. EHVI obtains faster reduction of IGD than EI and EST except for the case of $M = 5$; EST achieves the fastest IGD reduction. By comparing the optimization performances in both DTLZ7 and DTLZ7- k 10 problems, the advantage of EHVI over EI and EST in DTLZ7 is more obvious than that in DTLZ7- k 10. It is indicated that EHVI is more suitable for the problems with large number of design variables.

5. Conclusion

The Kriging-surrogate-based EA considering EHVI, EI, and EST as updating criteria was conducted in 3- to 15-objective DTLZ1-7 test problems, and the optimization performances using different updating criteria were compared in this study. Thereinto, the fastest WFG exact algorithm and the approximate algorithm based on Monte Carlo sampling were adopted for the hypervolume calculation.

In DTLZ1 problem, EHVI had faster IGD reduction than EI and EST when $M = 5$, but EHVI did not keep the advantage when M was larger than 5. In DTLZ2 and DTLZ5 problems, when $M = 3$ and 5, the optimization performance of using EHVI as the updating criterion was poor. However, the optimization performance of EHVI was improved when the number of objectives was larger than 5. In DTLZ4 and DTLZ5 problems, the advantage of EHVI was shown gradually as M increased, and EHVI obtained faster reduction of IGD than EST and EI when M was larger than 8. In DTLZ3, DTLZ6, and DTLZ7 problems, EHVI always obtained better convergence and diversity performances than EI and EST for any number of objectives. In DTLZ7- k 10 problem, the advantage of EHVI over EI and EST was reduced compared with that in DTLZ7. On the whole, in the non-constrained case, EHVI shows that it is highly competitive for the updating of the Kriging model compared with EI and EST in many-objective optimization, especially when the test problem is complex and the number of objectives or design variables is large.

On the other hand, EST always obtained faster reduction of IGD than EI for any number of objectives in DTLZ1 and DTLZ4 problems. It is because the two problems are easily approximated by the Kriging model. In most of the other many-objective optimizations, the updating criterion EI considering estimation errors had better optimization performance than EST without considering estimation errors, especially for the problems with big number of objectives.

Conflict of Interests

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

References

- [1] C. A. C. Coello, G. B. Lamont, and D. A. van Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Springer, New York, NY, USA, 2007.
- [2] M. Garza-Fabre, G. T. Pulido, and C. A. Coello Coello, "Ranking methods for many-objective optimization," in *MICAI 2009: Advances in Artificial Intelligence: 8th Mexican International Conference on Artificial Intelligence, Guanajuato, México, November 9-13, 2009. Proceedings*, vol. 5845 of *Lecture Notes in Computer Science*, pp. 633–645, Springer, Berlin, Germany, 2009.
- [3] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler, "Scalable test problems for evolutionary multi-objective optimization," TIK-Technical Report No. 112, 2001.
- [4] S. F. Adra and P. J. Fleming, "Diversity management in evolutionary many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 2, pp. 183–195, 2011.
- [5] D. Hadka and P. Reed, "Borg: an auto-adaptive many-objective evolutionary computing framework," *Evolutionary Computation*, vol. 21, no. 2, pp. 231–259, 2013.
- [6] S. Huband, P. Hingston, L. Barone, and L. While, "A review of multiobjective test problems and a scalable test problem toolkit," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 5, pp. 477–506, 2006.
- [7] Q. Zhang, A. Zhou, S. Zhao, P. N. Suganthan, W. Liu, and S. Tiwari, "Multiobjective optimization test instances for the CEC 2009 special session and competition," Tech. Rep. CES-487, University of Essex, 2009.
- [8] A. Lopez, C. A. Coello Coello, A. Oyama, and K. Fujii, "An alternative performance relation to deal with many-objective optimization problems," in *Evolutionary Multi-Criterion Optimization*, vol. 7811 of *Lecture Notes in Computer Science*, pp. 291–306, Springer, Berlin, Germany, 2013.
- [9] S. Yang, M. Li, X. Liu, and J. Zheng, "A grid-based evolutionary algorithm for many-objective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 5, pp. 721–736, 2013.
- [10] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, part I: solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.
- [11] Y. Lian and M.-S. Liou, "Multiobjective optimization using coupled response surface model and evolutionary algorithm," *AIAA Journal*, vol. 43, no. 6, pp. 1316–1325, 2005.
- [12] H. Fang, M. Rais-Rohani, and M. F. Horstemeyer, "Multiobjective crashworthiness optimization with radial basis functions," in *Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, pp. 2095–2104, Albany, NY, USA, September 2004.
- [13] K. Shimoyama, S. Yoshimizu, S. Jeong, S. Obayashi, and Y. Yokono, "Multi-objective design optimization for a steam turbine stator blade using LES and GA," *Journal of Computational Science and Technology*, vol. 5, no. 3, pp. 134–147, 2011.
- [14] M. Papadrakakis, N. D. Lagaros, and Y. Tsompanakis, "Optimization of large-scale 3-D trusses using evolution strategies

- and neural networks,” *International Journal of Space Structures*, vol. 14, no. 3, pp. 211–223, 1999.
- [15] D. R. Jones, M. Schonlau, and W. J. Welch, “Efficient global optimization of expensive black-box functions,” *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, 1998.
- [16] S. Jeong, Y. Minemura, and S. Obayashi, “Optimization of combustion chamber for diesel engine using Kriging model,” *Journal of Fluid Science and Technology*, vol. 1, no. 2, pp. 138–146, 2006.
- [17] M. T. M. Emmerich, K. C. Giannakoglou, and B. Naujoks, “Single- and multiobjective evolutionary optimization assisted by Gaussian random field metamodels,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 4, pp. 421–439, 2006.
- [18] M. Li, G. Li, and S. Azarm, “A kriging metamodel assisted multi-objective genetic algorithm for design optimization,” *Journal of Mechanical Design, Transactions of the ASME*, vol. 130, no. 3, Article ID 031401, 2008.
- [19] K. Shimoyama, K. Sato, S. Jeong, and S. Obayashi, “Updating Kriging surrogate models based on the hypervolume indicator in multi-objective optimization,” *Journal of Mechanical Design*, vol. 135, no. 9, Article ID 094503, 2013.
- [20] N. A. C. Cressie, *Statistics for Spatial Data*, John Wiley & Sons, New York, NY, USA, 1993, Revision.
- [21] E. Zitzler and L. Thiele, “Multiobjective optimization using evolutionary algorithms—a comparative case study,” in *Proceedings of the 5th International Conference on Parallel Problem Solving from Nature (PPSN '98)*, vol. 1498 of *Lecture Notes in Computer Science*, pp. 292–301, Amsterdam, The Netherlands, 1998.
- [22] J. Wu and S. Azarm, “Metrics for quality assessment of a multiobjective design optimization solution set,” *Journal of Mechanical Design, Transactions of the ASME*, vol. 123, no. 1, pp. 18–25, 2001.
- [23] M. Fleischer, “The measure of pareto optima applications to multi-objective metaheuristics,” in *Evolutionary Multi-Criterion Optimization*, vol. 2632 of *Lecture Notes in Computer Science*, pp. 519–533, Springer, Berlin, Germany, 2003.
- [24] L. While, P. Hingston, L. Barone, and S. Huband, “A faster algorithm for calculating hypervolume,” *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 1, pp. 29–38, 2006.
- [25] C. M. Fonseca, L. Paquete, and M. López-Ibáñez, “An improved dimension-sweep algorithm for the hypervolume indicator,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 1157–1163, July 2006.
- [26] L. While, L. Bradstreet, and L. Barone, “A fast way of calculating exact hypervolumes,” *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 1, pp. 86–95, 2012.
- [27] J. Bader and E. Zitzler, “Hype: an algorithm for fast hypervolume-based many-objective optimization,” TIK-Technical Report 286, 2008.
- [28] K. Bringmann and T. Friedrich, “Approximating the least hypervolume contributor: NP-hard in general, but fast in practice,” in *Evolutionary Multi-Objective Optimization*, vol. 5467 of *Lecture Notes on Computer Science*, pp. 6–20, Springer, Berlin, Germany, 2009.
- [29] H. Ishibuchi, N. Tsukamoto, Y. Sakane, and Y. Nojima, “Hypervolume approximation using achievement scalarizing functions for evolutionary many-objective optimization,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '09)*, pp. 530–537, Trondheim, Norway, May 2009.
- [30] S. Weinzierl, “Introduction to Monte Carlo methods,” Tech. Rep. NIKHEF-00-012, NIKHEF, Theory Group, Amsterdam, The Netherlands, 2000.
- [31] M. D. McKay, R. J. Beckman, and W. J. Conover, “A comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979.
- [32] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multi-objective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [33] K. Deb and R. B. Agrawal, “Simulated binary crossover for continuous search space,” *Complex Systems*, vol. 9, no. 2, pp. 115–148, 1995.
- [34] D. A. Van Veldhuizen and G. B. Lamont, “Multiobjective evolutionary algorithm research: a history and analysis,” Tech. Rep. TR-98-03, Air Force Institute of Technology, Dayton, Ohio, USA, 1998.

Research Article

Neural Network-Based Fault-Tolerant Control of Underactuated Surface Vessels

Bong Seok Park

Division of Electrical, Electronic, and Control Engineering, Kongju National University, Cheonan 31080, Republic of Korea

Correspondence should be addressed to Bong Seok Park; bspark@kongju.ac.kr

Received 1 September 2014; Accepted 27 October 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Bong Seok Park. 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 addresses the problem of trajectory tracking of underactuated surface vessels (USVs) in the presence of thruster failure. Multilayer neural networks (MNNs) are employed to estimate the unknown model parameters and external disturbances. To design a fault-tolerant controller without a fault detection scheme, we use the Nussbaum gain technique. We introduce an additional control to resolve the difficulty arising from having fewer inputs than degrees-of-freedom. Further, an approach angle is proposed to track both a straight and curved path. Stability analysis and simulations are performed to demonstrate the effectiveness of the proposed scheme.

1. Introduction

As land resources are depleted, interest in ocean exploration has increased. There are many marine vehicles designed to explore the ocean. Among these, unmanned marine vehicles (UMVs) such as autonomous underwater vehicles (AUVs) and unmanned surface vessels are useful because they can reduce the cost and the risk of personal injury [1]. Therefore, significant research has been undertaken to perform complex tasks using UMVs.

An important problem in the control of UMVs is failure occurrence on the actuators [2, 3]. If there is no immediate solution for actuator failure, the UMV must abort the mission and return to the base for repair. This results in a waste of time and money. Therefore, to enhance the reliability while performing its mission in the presence of thruster failure, there are more thrusters than the degrees-of-freedom of the UMV. For this reason, the thruster-force allocation problem is important to overcome thruster failure [4]. The allocation of thruster forces of an AUV that can accommodate both thruster faults and saturation was presented in [5, 6]. A fault diagnosis and accommodation system was proposed using weighted pseudoinverse in [7]. Weighted pseudoinverse and quantum particle swarm optimization were introduced for hybrid fault-tolerant control in [8]. The combination of

backstepping technique and fuzzy control was presented for tracking control of fully actuated surface vessels in [9]. However, all these works focused on fully or overactuated systems. This implies that the cost and weight of the system are increased because there must be more devices. Moreover, if a fully actuated system is damaged, we then have an underactuated controller [10]. Therefore, it is necessary to develop a fault-tolerant controller for underactuated systems.

Motivated by these observations, a neural network-based fault-tolerant control is presented for underactuated surface vessels (USVs). The proposed algorithm can be applied to AUVs because the equations of motion for surface vessels are the same as those of AUVs in the horizontal plane. For a practical perspective, we assume that the bow and stern are not symmetric and the model parameters for the hydrodynamic terms are unknown. To estimate the unknown model parameters and external disturbances, we employ multilayer neural networks (MNNs) [11]. Unlike other fault-tolerant control methods [12–14], we do not require a fault detection scheme; we design the fault-tolerant controller using the Nussbaum gain technique [15]. Further, an additional control is introduced to resolve the difficulty of controller design of underactuated systems, and an approach angle is proposed to track any trajectory including both a straight- and curved-line path. It is proved that all

the signals of the closed-loop system are ultimately, uniformly bounded. Simulation results are presented to demonstrate the effectiveness of the proposed scheme.

The main contributions of our paper are as follows. (1) The fault-tolerant controller is designed without the fault detection scheme. (2) For underactuated surface vessels, we develop the additional control input and analyze the stability including the sway dynamics. (3) The approach angle which is composed only of position information of USVs is introduced. With the help of a novel approach angle, we can track any trajectory including both a straight- and curved-line path. (4) For a practical application, the model parameters for the hydrodynamic terms and external disturbances are assumed to be unknown. These uncertainties are estimated by MNNs.

Throughout this paper, the following notations are used: (1) $\|\cdot\|$ denotes any suitable vector norm; (2) $\text{tr}(\cdot)$ is the trace; (3) $\|\cdot\|_F$ is the Frobenius norm defined by $\|A\|_F = \sqrt{\text{tr}(A^T A)}$; (4) W_i and \widehat{W}_i denote bounded ideal neural weights and the estimates of neural weights, respectively; (5) $\widetilde{W}_i = W_i - \widehat{W}_i$ denotes the error between W_i and \widehat{W}_i ; (6) $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ denote the minimum and maximum eigenvalues of matrix A , respectively; (7) $\text{diag}(\cdot)$ is a diagonal matrix.

2. Underactuated Surface Vessel Model

The kinematics and dynamics of USVs are described as follows [16]:

$$\begin{aligned} \dot{\eta} &= J(\psi) \nu, \\ M\dot{\nu} &= -C(\nu) \nu - D(\nu) \nu + \tau_d + \tau, \end{aligned} \quad (1)$$

where $\eta = [x, y, \psi]^T$ denotes position (x, y) and yaw angle ψ of the USV in the earth-fixed frame; $\nu = [u, v, r]^T$ denotes surge, sway, and yaw velocities of the USV in the body-fixed frame, respectively, $\tau_d = [\tau_{d,u}, \tau_{d,v}, \tau_{d,r}]^T$ denotes the external forces such as wind and ocean currents, and $\tau = [\tau_u, 0, \tau_r]^T$ is the control vector of the surge force τ_u and the yaw moment τ_r . In the above equation, matrices $J(\psi)$, $D(\nu)$, $C(\nu)$, and M are given as follows:

$$\begin{aligned} J(\psi) &= \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \\ D(\nu) &= \begin{bmatrix} d_{11}(u) & 0 & 0 \\ 0 & d_{22}(v, r) & d_{23}(v, r) \\ 0 & d_{32}(v, r) & d_{33}(v, r) \end{bmatrix}, \\ C(\nu) &= \begin{bmatrix} 0 & 0 & -m_{22}v - m_t r \\ 0 & 0 & m_{11}u \\ m_{22}v + m_t r & -m_{11}u & 0 \end{bmatrix}, \\ M &= \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{32} & m_{33} \end{bmatrix}, \end{aligned} \quad (2)$$

where

$$\begin{aligned} m_{11} &= m - X_{\dot{u}}, & m_{22} &= m - Y_{\dot{v}}, & m_{23} &= mx_g - Y_{\dot{r}}, \\ m_{32} &= mx_g - N_{\dot{v}}, & m_{33} &= I_z - N_{\dot{r}}, \\ m_t &= \frac{(m_{23} + m_{32})}{2}, & d_{11}(u) &= -(X_u + X_{u|u}|u|), \\ d_{22}(v, r) &= -(Y_v + Y_{|v|v}|v| + Y_{|r|v}|r|), \\ d_{23}(v, r) &= -(Y_r + Y_{|v|r}|v| + Y_{|r|r}|r|), \\ d_{32}(v, r) &= -(N_v + N_{|v|v}|v| + N_{|r|v}|r|), \\ d_{33}(v, r) &= -(N_r + N_{|v|r}|v| + N_{|r|r}|r|). \end{aligned} \quad (3)$$

$X_u, X_{u|u}, Y_v, Y_{|v|v}, Y_{|r|v}, Y_r, Y_{|v|r}, Y_{|r|r}, N_v, N_{|v|v}, N_{|r|v}, N_r, N_{|v|r}$, and $N_{|r|r}$ are the linear and quadratic drag coefficients, m is the mass of the USV, X_u, Y_v, Y_r, N_v , and N_r are the added masses, x_g is the X_b -coordinate of the USV center of gravity in the body-fixed frame, I_z is the inertia with respect to the vertical axis, and M is invertible.

Remark 1. Unlike other works, we assume that the parameters m_{23} and m_{32} are not zero. This means that the sway dynamics is influenced by the yaw moment τ_r and we must control sway and yaw velocities simultaneously using only the yaw moment τ_r . When the parameters m_{23} and m_{32} are not zero, controller design difficulties are increased.

To address the problem stated in Remark 1, we utilize the following state transformations [17]:

$$\bar{x} = x + \epsilon \cos \psi, \quad \bar{y} = y + \epsilon \sin \psi, \quad \bar{v} = v + \epsilon r, \quad (4)$$

where $\epsilon = m_{23}/m_{22}$. The transformed equation (4) implies that the virtual center of mass is positioned by ϵ in the longitudinal direction. Using (4), the USV dynamics (1) can be expressed by defining state variables $z_1 = [z_{11}, z_{12}, z_{13}]^T = [\bar{x}, \bar{y}, \psi]^T$ and $z_2 = [z_{21}, z_{22}, z_{23}]^T = [u, \bar{v}, r]^T$:

$$\begin{aligned} \dot{z}_1 &= J(\psi) z_2, \\ \dot{z}_2 &= \varphi(z_2) + \bar{\tau}, \\ y &= z_1, \end{aligned} \quad (5)$$

where

$$\begin{aligned} \varphi(z_2) &= [\varphi_1, \varphi_2, \varphi_3]^T, \\ \varphi_1 &= \frac{(m_{22}vr + m_t r^2 - d_{11}(u)u + \tau_{d,u})}{m_{11}}, \\ \varphi_2 &= -\frac{m_{11}}{m_{22}}ur - \frac{d_{22}(v, r)}{m_{22}}v - \frac{d_{23}(v, r)}{m_{22}}r + \frac{1}{m_{22}}\tau_{d,v}, \end{aligned}$$

$$\begin{aligned}
\varphi_3 &= \frac{1}{\Delta} \left\{ (m_{11}m_{22} - m_{22}^2)uv + (m_{11}m_{32} - m_t m_{22})ur \right. \\
&\quad - (d_{33}(v, r)r + d_{32}(v, r)v)m_{22} \\
&\quad + (d_{23}(v, r)r + d_{22}(v, r)v)m_{32} \\
&\quad \left. - m_{32}\tau_{d,v} + m_{22}\tau_{d,r} \right\}, \\
\Delta &= m_{22}m_{33} - m_{23}m_{32}, \\
\bar{\tau} &= [\bar{\tau}_1, 0, \bar{\tau}_2]^T = \left[\frac{\tau_u}{m_{11}}, 0, \frac{m_{22}\tau_r}{\Delta} \right]^T.
\end{aligned} \tag{6}$$

In these expressions, we assume that φ_u , φ_v , and φ_r are unknown time-varying functions because they cannot be obtained precisely by measurement or calculation.

When faults occur in the control vector $\bar{\tau}$, we can express them as follows:

$$\bar{\tau} = F_p \tau_n + f_b, \tag{7}$$

where $\tau_n = [\tau_{n,1}, 0, \tau_{n,2}]^T$ is a nominal control vector, $F_p = \text{diag}(f_1, 0, f_2) \in \mathbb{R}^{3 \times 3}$ denotes the partial loss of control vector, and $f_b = [f_{b,1}, 0, f_{b,2}]^T$ is an unknown bias vector. f_1 and f_2 are defined as

$$f_i = \begin{cases} 0 < f_i < 1; & \text{if the thrusters are in partial failure} \\ f_i = 1; & \text{if the thrusters are not in failure.} \end{cases} \tag{8}$$

The control objective is to design a fault-tolerant controller such that it tracks the reference trajectory $\eta_d = [x_d, y_d, \psi_d]$ even though faults exist in the control vector of the USV. The reference trajectory is given as

$$\begin{aligned}
\dot{x}_d &= u_d \cos \psi_d - v_d \sin \psi_d, \\
\dot{y}_d &= u_d \sin \psi_d + v_d \cos \psi_d, \\
\dot{\psi}_d &= r_d,
\end{aligned} \tag{9}$$

where x_d and y_d denote the desired position; ψ_d denotes the desired yaw angle, and u_d , v_d , and r_d are the desired velocities of the USV. To generate the reference trajectory, we can use path planning techniques presented in [18, 19].

Assumption 1. The desired velocities u_d , v_d , and r_d are bounded. Further, the first derivatives \dot{u}_d , \dot{v}_d , and \dot{r}_d are available and bounded.

3. Fault-Tolerant Control

3.1. Neural Network Estimation. In this paper, MNN is used to estimate the unknown function $\varphi(z_2)$. From the universal approximation property for neural networks (NNs) [20], the nonlinear function $\varphi(z_2)$ can be written as

$$\varphi(z_2) = W^T \sigma(V^T z_2) + \varepsilon, \tag{10}$$

where ε is the NN reconstruction error and $\sigma(\cdot)$ is the sigmoid function defined as $\sigma(z) = 1/(1 + e^{-z})$.

Assumption 2. W and V are bounded such that $\|W\|_F \leq W_M$ and $\|V\|_F \leq V_M$ for positive constants W_M and V_M , respectively, and ε is bounded such that $\|\varepsilon\| \leq \varepsilon_M$ for a positive constant ε_M .

The Taylor series expansion of $\sigma(\cdot)$ in (10) for a given z_2 can be written as

$$\sigma(V^T z_2) = \sigma(\bar{V}^T z_2) + \bar{\sigma}(\bar{V}^T z_2) \bar{V}^T z_2 + H(\bar{V}^T z_2)^2, \tag{11}$$

where $\bar{\sigma} = (d\sigma(V^T z_2)/d(V^T z_2))|_{V^T z_2 = \bar{V}^T z_2}$, $H(\bar{V}^T z_2)^2$ denotes higher order terms, and $\bar{V} = V - \bar{V}$. From (10) and (11), we can obtain the following result:

$$\varphi(z_2) - \hat{\varphi}(z_2) = \bar{W}^T \sigma(\bar{V}^T z_2) + \bar{W}^T \bar{\sigma}(\bar{V}^T z_2) \bar{V}^T z_2 + P, \tag{12}$$

where $P = \bar{W}^T \bar{\sigma}(\bar{V}^T z_2) \bar{V}^T z_2 + W^T H(\bar{V}^T z_2)^2 + \varepsilon$.

Property 1. The unknown term P in (12) is bounded such that $\|P\| \leq \delta^T \Theta$, where $\delta = [c_0, c_1, c_2, c_3, c_4]^T$, $\Theta = [1, \|z_2\|, \|\bar{W}\|_F \|z_2\|, \|\bar{V}\|_F \|z_2\|, \|\bar{W}\|_F \|\bar{V}\|_F \|z_2\|]^T$, and c_j ($j = 0, \dots, 4$) is a positive bounding constant.

Proof. See the appendix. \square

3.2. Neural Network-Based Fault-Tolerant Control. In this section, we design the neural network-based fault-tolerant control system. To avoid the difficulty of detecting the faults, the controller is proposed using the Nussbaum gain technique. The following lemma is used to design the neural network-based fault-tolerant control law.

Lemma 2 (see [21]). *Let $V(\cdot)$ and $\zeta(\cdot)$ be smooth functions defined on $[0, t_f)$ with $V(t) \geq 0$, $\forall t \in [0, t_f)$. For $t \in [0, t_f)$, if the following inequality holds,*

$$V(t) \leq \alpha_2 + e^{-\alpha_3 t} \int_0^t g(\lambda) N(\zeta) \dot{\zeta} e^{\alpha_3 \lambda} d\lambda + e^{-\alpha_3 t} \int_0^t \dot{\zeta} e^{\alpha_3 \lambda} d\lambda, \tag{13}$$

where α_2 and α_3 are positive constants, $N(\zeta) = e^{\zeta^2} \cos((\pi/2)\zeta)$ is a Nussbaum-type function, and $g(t)$ is a time-varying parameter which takes values in the closed intervals $\Omega := [\omega^-, \omega^+]$, then $V(t)$, $\zeta(t)$, and $\int_0^t g(\lambda) N(\zeta) \dot{\zeta} d\lambda$ must be bounded on $[0, t_f)$.

We now design the neural network-based fault-tolerant control law using the dynamic surface design approach.

Step 1. Define the following errors:

$$\begin{aligned}
e_1 &= z_{11} - \bar{x}_d, \\
e_2 &= z_{12} - \bar{y}_d, \\
e_3 &= z_{13} - \psi_d,
\end{aligned} \tag{14}$$

where $\bar{x}_d = x_d + \epsilon \cos \psi_d$ and $\bar{y}_d = y_d + \epsilon \sin \psi_d$. An approach angle ψ_a is defined as

$$\psi_a = \arctan\left(\frac{e_2}{e_1}\right) \tanh\left(\frac{D^2}{\gamma_1}\right) + \psi_d \left(1 - \tanh\left(\frac{D^2}{\gamma_1}\right)\right), \quad (15)$$

where $D = \sqrt{e_1^2 + e_2^2}$ and γ_1 is a positive constant.

Remark 3. In this paper, we use an approach angle ψ_a as an alternative for the desired yaw angle ψ_d . Using the approach angle ψ_a , we can track any trajectory including both straight-line and curved-line paths. There are other works using the approach angle; however, our approach angle is different from the other studies because it requires only position information.

Using (5) and (9), the time derivative of (14) is given as

$$\begin{aligned} \dot{e}_1 &= z_{21} \cos z_{13} - z_{22} \sin z_{13} - u_d \cos \psi_d + \bar{v}_d \sin \psi_d, \\ \dot{e}_2 &= z_{21} \sin z_{13} + z_{22} \cos z_{13} - u_d \sin \psi_d - \bar{v}_d \cos \psi_d, \\ \dot{e}_3 &= z_{23} - \dot{\psi}_a. \end{aligned} \quad (16)$$

To stabilize the error dynamics of (16), we choose the virtual controls ϕ_1 , ϕ_2 , and ϕ_3 as

$$\begin{aligned} \phi_1 &= -k_1 e_1 \cos z_{13} - k_1 e_2 \sin z_{13} \\ &\quad + u_d \cos(z_{13} - \psi_d) + \bar{v}_d \sin(z_{13} - \psi_d), \\ \phi_2 &= k_1 e_1 \sin z_{13} - k_1 e_2 \cos z_{13} \\ &\quad - u_d \sin(z_{13} - \psi_d) + \bar{v}_d \cos(z_{13} - \psi_d), \\ \phi_3 &= -k_2 e_3 + \dot{\psi}_a, \end{aligned} \quad (17)$$

where k_1 and k_2 are design parameters.

Step 2. In this step, we use the NNs to estimate uncertain terms φ_1 , φ_2 , and φ_3 . From (12), we can derive the following results:

$$\varphi_i - \hat{\varphi}_i = \bar{W}_i^T \sigma(\bar{V}_i z_2) + \bar{W}_i^T \bar{\sigma}(\bar{V}_i^T z_2) \bar{V}_i^T z_2 + P_i, \quad (18)$$

$$i = 1, 2, 3,$$

where $\hat{\varphi}_i$ is the estimate of φ_i and $P_i = \bar{W}_i^T \bar{\sigma}(\bar{V}_i^T z_2) \bar{V}_i^T z_2 + W_i^T H(\bar{V}_i^T z_2)^2 + \varepsilon_i$. By Property 1, P_i is bounded such that $\|P_i\| \leq \delta_i^T \Theta_i$, where δ_i is the unknown constant vector and $\Theta_i = [1, \|z_2\|, \|\bar{W}_i\| \|z_2\|, \|\bar{V}_i\| \|z_2\|, \|\bar{W}_i\| \|\bar{V}_i\| \|z_2\|]^T$.

Assumption 3. W_i and V_i are bounded such that $\|W_i\| \leq W_{i,M}$ and $\|V_i\| \leq V_{i,M}$ for positive constants $W_{i,M}$ and $V_{i,M}$, respectively, and ε_i is bounded such that $\|\varepsilon_i\| \leq \varepsilon_{i,M}$ for a positive constant $\varepsilon_{i,M}$.

Define the following errors:

$$\begin{aligned} e_4 &= z_{21} - \phi_{1,f} - \gamma_2 \tanh \beta_1, \\ e_5 &= z_{22} - \phi_{2,f} - \gamma_2 \tanh \beta_2, \\ e_6 &= z_{23} - \phi_{3,f} - \gamma_2 \tanh \beta_3, \end{aligned} \quad (19)$$

where γ_2 is a positive constant. $\phi_{1,f}$, $\phi_{2,f}$, and $\phi_{3,f}$ are signals obtained by the following first-order filters:

$$\kappa_i \dot{\phi}_{i,f} + \phi_{i,f} = \phi_i, \quad \phi_{i,f}(0) = \phi_i(0), \quad i = 1, 2, 3, \quad (20)$$

where κ_i is a positive constant. Here, β_1 , β_2 , and β_3 are used to solve the problem that there is no actual control on the sway dynamics as seen from (5) and obtained by the following differential equations:

$$\begin{aligned} \dot{\beta}_1 &= -\frac{\cosh^2 \beta_1}{\gamma_2} T_u \tanh \beta_2, \\ \dot{\beta}_2 &= \frac{\cosh^2 \beta_2}{\gamma_2} (k_3 e_5 - e_1 \sin z_{13} + e_2 \cos z_{13} + \hat{\varphi}_2 \\ &\quad + \hat{\delta}_2^T \Theta_2 \operatorname{sgn}(e_5) - \dot{\phi}_{2,f}), \\ \dot{\beta}_3 &= -\frac{\cosh^2 \beta_3}{\gamma_2} T_r \tanh \beta_2, \end{aligned} \quad (21)$$

where T_u , T_r , $k_3 > 0$ are design parameters and $\hat{\delta}_2$ is the estimate of δ_2 . To design the unified control algorithm from the error dynamics of (19), we introduce a function $h(\rho(t))$ as

$$h(\rho(t)) = \begin{cases} 1, & \|\bar{y}(t)\| > \rho(t) \\ 0, & \|\bar{y}(t)\| \leq \rho(t), \end{cases} \quad (22)$$

where $\rho(t)$ is the threshold which is the criterion for judgment.

Remark 4. The threshold $\rho(t)$ can be derived from the fault detection algorithm. However, the proposed algorithm does not require the threshold $\rho(t)$ because we design the neural network-based fault-tolerant controller under the assumption that the function $h(\rho(t))$ in (22) is unknown.

Using (5), (7), and (22), the time derivative of (19) is represented as

$$\begin{aligned} \dot{e}_4 &= \bar{\varphi}_1 + g_1(t) \tau_{n,1} + T_u \tanh \beta_2 - \dot{\phi}_{1,f}, \\ \dot{e}_5 &= \varphi_2 - k_4 e_5 + e_1 \sin z_{13} - e_2 \cos z_{13} \\ &\quad - \hat{\varphi}_2 - \hat{\delta}_2^T \Theta_2 \operatorname{sgn}(e_5), \end{aligned} \quad (23)$$

$$\dot{e}_6 = \bar{\varphi}_3 + g_2(t) \tau_{n,2} + T_r \tanh \beta_2 - \dot{\phi}_{3,f},$$

where

$$\begin{aligned} \bar{\varphi}_1 &= \varphi_1 + h(\rho(t)) f_{b,1}, \\ \bar{\varphi}_3 &= \varphi_3 + h(\rho(t)) f_{b,2}, \\ g_1(t) &= 1 - h(\rho(t)) + h(\rho(t)) f_1, \\ g_2(t) &= 1 - h(\rho(t)) + h(\rho(t)) f_2. \end{aligned} \quad (24)$$

Property 2. $g_1(t)$ and $g_2(t)$ are bounded such that $0 < g_1(t) \leq 1$ and $0 < g_2(t) \leq 1$.

Proof. See the appendix. \square

From (23), we choose the actual controls $\tau_{n,1}$ and $\tau_{n,2}$ as follows:

$$\begin{aligned} \tau_{n,1} &= N(\zeta_1)\theta_1, \\ \theta_1 &= \widehat{\varphi}_1 + k_3 e_4 - \dot{\phi}_{1,f} + T_u \tanh \beta_2 + e_1 \cos z_{13} \\ &\quad + e_2 \sin z_{13} + \delta_1^T \Theta_1 \operatorname{sgn}(e_4), \\ \dot{\zeta}_1 &= e_4 \theta_1, \\ \tau_{n,2} &= N(\zeta_2)\theta_2, \\ \theta_2 &= \widehat{\varphi}_3 + k_3 e_6 - \dot{\phi}_{3,f} + T_r \tanh \beta_2 + \delta_3^T \Theta_3 \operatorname{sgn}(e_6) + e_3, \\ \dot{\zeta}_2 &= e_6 \theta_2, \end{aligned} \quad (25)$$

where $N(\zeta_1)$ and $N(\zeta_2)$ are Nussbaum-type functions. For the stability analysis, we define the boundary layer error as

$$\omega_i = \phi_{i,f} - \phi_i, \quad i = 1, 2, 3. \quad (26)$$

Let us consider the following Lyapunov functions:

$$\begin{aligned} V_1 &= \frac{1}{2} \left\{ e_1^2 + e_2^2 + e_4^2 + e_5^2 + \sum_{i=1}^2 \omega_i^2 \right. \\ &\quad \left. + \sum_{i=1}^2 \left(\widetilde{W}_i^T \Lambda_i^{-1} \widetilde{W}_i + \operatorname{tr} \left(\widetilde{V}_i^T \Lambda_i^{-1} \widetilde{V}_i \right) + \delta_i^T \Lambda_i^{-1} \delta_i \right) \right\}, \\ V_2 &= \frac{1}{2} \left(e_3^2 + e_6^2 + \omega_3^2 \right. \\ &\quad \left. + \widetilde{W}_3^T \Lambda_3^{-1} \widetilde{W}_3 + \operatorname{tr} \left(\widetilde{V}_3^T \Lambda_3^{-1} \widetilde{V}_3 \right) + \delta_3^T \Lambda_3^{-1} \delta_3 \right), \end{aligned} \quad (27)$$

where Λ_i is positive definite matrix.

Theorem 5. Consider the USV model (5) in the presence of the faults. For any initial conditions $V_1(0) \leq \mu_1$ and $V_2(0) \leq \mu_2$, where μ_1 and μ_2 are any positive constants, if Assumptions 1–3 are satisfied and one applies the controllers (17) and (25) with the adaptation laws defined as

$$\begin{aligned} \dot{\widetilde{W}}_i &= \Lambda_i e_{i+3} \sigma \left(\widetilde{V}_i^T z_2 \right) - \iota_i \Lambda_i \widetilde{W}_i, \\ \dot{\widetilde{V}}_i &= \Lambda_i z_2 \left(\overline{\sigma} \left(\widetilde{V}_i^T z_2 \right) \widetilde{W}_i e_{i+3} \right)^T - \iota_i \Lambda_i \widetilde{V}_i, \\ \dot{\delta}_i &= \Gamma_i \Theta_i |e_{i+3}| - \iota_i \Gamma_i \delta_i, \\ i &= 1, 2, 3, \end{aligned} \quad (28)$$

where ι_i is a positive constant, then one can ensure that

$$\lim_{t \rightarrow \infty} \|\eta - \eta_d\| = \vartheta, \quad (29)$$

where ϑ is a positive constant.

Proof. See the appendix. \square

4. Simulation Results

In this section, we simulate a neural network-based fault-tolerant controller to verify the performance of the proposed control system. The parameters of the underactuated surface vessel are as indicated in [22]. The control gains are chosen as $k_1 = 1.2$, $k_2 = k_3 = 2$, $T_u = T_r = 1$, $\gamma_1 = 0.0001$, $\gamma_2 = 1000$, and $\kappa_i = 0.2$, where $i = 1, 2, 3$. The reference trajectory is generated by the following desired velocities:

$$\begin{aligned} 0 \leq t < 5 : [u_d, v_d, r_d] &= \left[0.5 \sin \left(\frac{\pi t}{10} \right), 0, 0 \right], \\ 5 \leq t < 20 : [u_d, v_d, r_d] &= [0.5, 0, 0], \\ 20 \leq t : [u_d, v_d, r_d] &= [0.5, 0, 0.3]. \end{aligned} \quad (30)$$

The initial postures for the reference USV and the actual USV are $[x_d, y_d, \psi_d] = [0, 0, 0]$ and $[x, y, \psi] = [-0.5, 0.5, 0]$, respectively. In this simulation, we use three MNNs to estimate the uncertainties and the MNNs are composed of four hidden layers and three inputs. In addition, the weight tuning parameters of the MNNs are chosen as $\iota_i = 0.01$, $\Lambda_i = \operatorname{diag}(2, 2, 2)$, and $\Gamma_i = \operatorname{diag}(0.001, 0.001, 0.001, 0.001, 0.001)$, where $i = 1, 2, 3$. To verify the robustness of the proposed neural network-based fault-tolerance controller, we assume that the faults occur after 20 seconds such that $F_p = \operatorname{diag}(0.6, 0, 0.5)$ and $f_b = [1.5, 0, 1]$.

Figure 1(a) presents the trajectory tracking result. The tracking errors are depicted in Figures 1(c) and 1(d). As indicated in Figures 1(c) and 1(d), there are tracking errors when faults occur on $t = 20$ (sec); however these errors are reduced quickly. Further, the tracking errors are bounded for the entire time even though faults exist. From Figure 1(a), we can confirm that the actual USV tracks successfully the reference trajectory including both straight and curved paths. Therefore, we can conclude that the proposed control system possesses robustness for faults, model uncertainties, and external disturbances.

5. Conclusions

This paper presented a neural network-based fault-tolerant controller for an underactuated USV with faults, model uncertainties, and external disturbances. MNNs were employed to estimate the highly nonlinear uncertain terms, and approach angle was implemented to track any reference trajectory including both straight- and curved-line paths. To avoid the difficulty of detecting the faults, the neural network-based fault-tolerant controller was designed using the Nussbaum gain technique. From the simulation results, it was demonstrated that the proposed control system has good tracking performance in the presence of the faults, model uncertainties, and external disturbances. In the future, the actuator saturation problem will be addressed and evaluate the performance through real experiments.

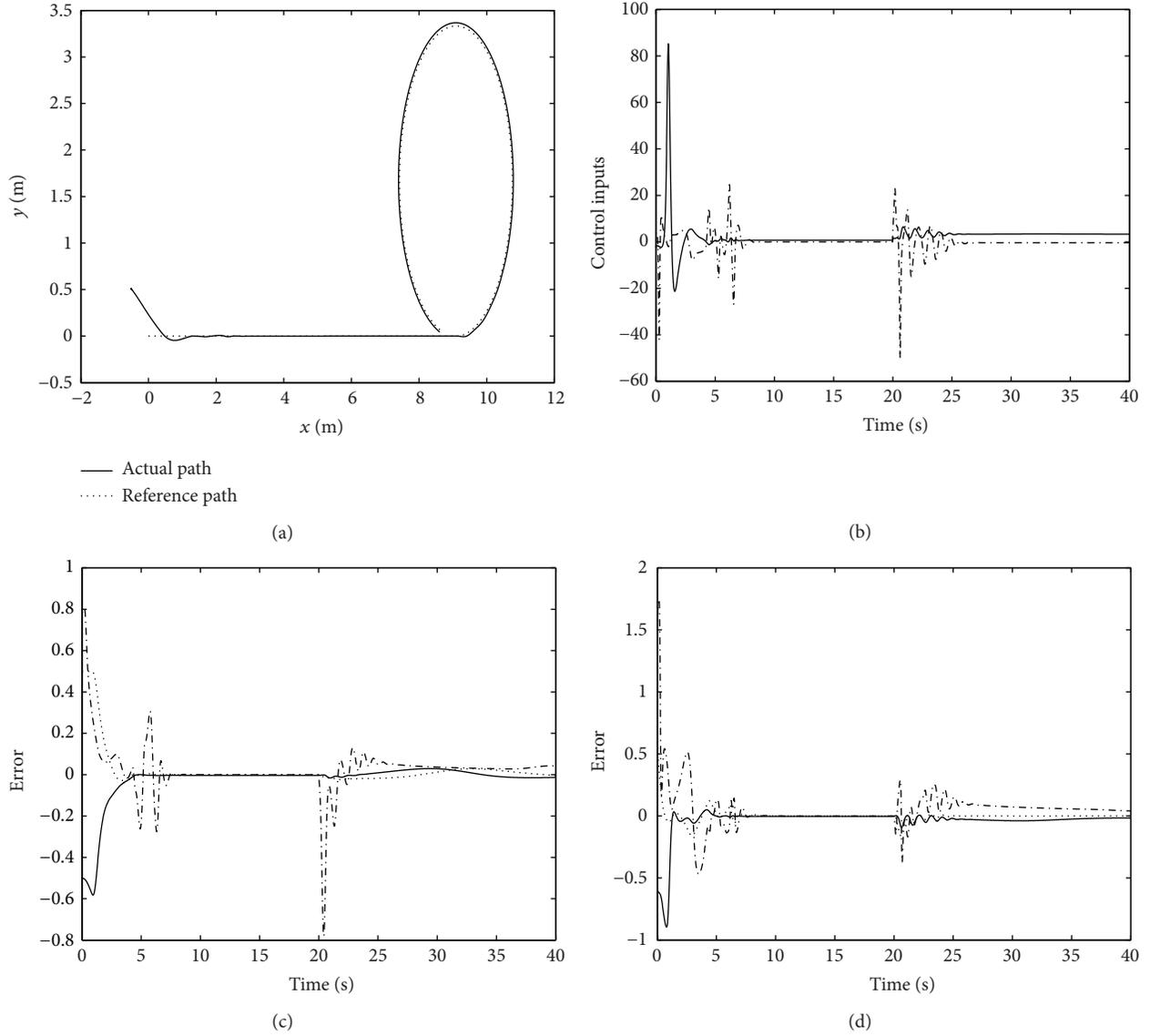


FIGURE 1: Simulation results. (a) Trajectory tracking result. (b) Control inputs (solid: $\bar{\tau}_1$, dash-dot: $\bar{\tau}_2$). (c) Tracking errors (solid: e_1 , dotted: e_2 , and dash-dot: e_3). (d) Tracking errors (solid: e_4 , dotted: e_5 , and dash-dot: e_6).

Appendix

Proof of Property 1. Since $\sigma(\cdot)$ and $\bar{\sigma}(\cdot)$ are bounded, the unknown term P in (12) can be written as

$$\|P\| \leq \alpha_0 \|\widehat{W}\|_F \|\widehat{V}\|_F \|z_2\| + \|W\|_F (\alpha_1 + \|\widehat{V}\|_F \|z_2\|) + \|\varepsilon\|, \quad (\text{A.1})$$

where α_0 and α_1 are positive constants. Using Assumption 2 and triangular inequality, (A.1) is bounded by

$$\begin{aligned} \|P\| &\leq \alpha_0 (W_M + \|\widehat{W}\|_F) (V_M + \|\widehat{V}\|_F) \|z_2\| \\ &\quad + \alpha_1 W_M + \alpha_1 W_M (V_M + \|\widehat{V}\|_F) \|z_2\| + \varepsilon_M \end{aligned}$$

$$\begin{aligned} &= c_0 + c_1 \|z_2\| + c_2 \|\widehat{W}\|_F \|z_2\| + c_3 \|\widehat{V}\|_F \|z_2\| \\ &\quad + c_4 \|\widehat{W}\|_F \|\widehat{V}\|_F \|z_2\| \\ &= \delta^T \Theta, \end{aligned} \quad (\text{A.2})$$

where $c_0 = \alpha_1 W_M (1 + V_M) + \varepsilon_M$, $c_1 = \alpha_0 W_M V_M + \alpha_1 W_M V_M$, $c_2 = \alpha_0 V_M$, $c_3 = \alpha_0 W_M + \alpha_1 W_M$, and $c_4 = \alpha_0$. \square

Proof of Property 2. Since f_1 and f_2 are bounded from (8) such that $0 < f_1 \leq 1$ and $0 < f_2 \leq 1$, one can easily prove that $0 < g_1(t) \leq 1$ and $0 < g_2(t) \leq 1$. \square

Proof of Theorem 5. We prove Theorem 5 in two steps. The first step is to prove boundness of e_1 and e_2 . In the second, we prove boundness of e_3 .

Step 1. The time derivative of V_1 using (16)–(20) and (23)–(26) results in

$$\begin{aligned} \dot{V}_1 = & -k_1 e_1^2 + e_1 \cos z_{13} (\bar{\omega}_1 + \gamma_2 \tanh \beta_1) \\ & - e_1 \sin z_{13} (\bar{\omega}_2 + \gamma_2 \tanh \beta_2) \\ & - k_1 e_2^2 + e_2 \sin z_{13} (\bar{\omega}_1 + \gamma_2 \tanh \beta_1) \\ & + e_2 \cos z_{13} (\bar{\omega}_2 + \gamma_2 \tanh \beta_2) \\ & - k_3 e_4^2 + g_1(t) N(\zeta_1) \dot{\zeta}_1 + \dot{\zeta}_1 - \bar{\delta}_1^T \Theta_1 |e_4| \\ & + e_4 (\bar{W}_1^T \sigma(\bar{V}_1^T z_2) + \bar{W}_1^T \bar{\sigma}(\bar{V}_1^T z_2) \bar{V}_1^T z_2 + P_1) \quad (\text{A.3}) \\ & - k_3 e_5^2 - \bar{\delta}_2^T \Theta_2 |e_5| \\ & + e_5 (\bar{W}_2^T \sigma(\bar{V}_2^T z_2) + \bar{W}_2^T \bar{\sigma}(\bar{V}_2^T z_2) \bar{V}_2^T z_2 + P_2) \\ & - \left(\frac{\bar{\omega}_1^2}{\kappa_1} + \bar{\omega}_1 \Upsilon_1 \right) - \left(\frac{\bar{\omega}_2^2}{\kappa_2} + \bar{\omega}_2 \Upsilon_2 \right) \\ & - \sum_{i=1}^2 \left(\bar{W}_i^T \Lambda_i^{-1} \bar{W}_i + \bar{V}_i^T \Lambda_i^{-1} \bar{V}_i + \bar{\delta}_i^T \Lambda_i^{-1} \bar{\delta}_i \right), \end{aligned}$$

where

$$\begin{aligned} \Upsilon_1 (e_1, e_2, e_3, e_4, e_6, \bar{\omega}_1, \bar{\omega}_3, u_d, \bar{v}_d, r_d, \psi_d, \dot{u}_d, \dot{\bar{v}}_d, k_1, \gamma_2) \\ = & -k_1 \dot{e}_1 \cos z_{13} + k_1 e_1 z_{23} \sin z_{13} - k_1 \dot{e}_2 \sin z_{13} \\ & - k_1 e_2 z_{23} \cos z_{23} + \cos(z_{13} - \psi_d) (\dot{u}_d + \bar{v}_d (z_{23} - r_d)) \\ & + \sin(z_{13} - \psi_d) (\dot{\bar{v}}_d - u_d (z_{23} - r_d)), \\ \Upsilon_2 (e_1, e_2, e_3, e_5, e_6, \bar{\omega}_2, \bar{\omega}_3, u_d, \bar{v}_d, r_d, \psi_d, \dot{u}_d, \dot{\bar{v}}_d, k_1, \gamma_2) \\ = & k_1 \dot{e}_1 \sin z_{13} + k_1 e_1 z_{23} \cos z_{13} - k_1 \dot{e}_2 \cos z_{13} \\ & + k_1 e_2 z_{23} \sin z_{23} + \cos(z_{13} - \psi_d) (\dot{\bar{v}}_d - u_d (z_{23} - r_d)) \\ & - \sin(z_{13} - \psi_d) (\dot{u}_d + \bar{v}_d (z_{23} - r_d)). \quad (\text{A.4}) \end{aligned}$$

On $V_1 = \mu_1$ and $V_2 = \mu_2$, $\Upsilon_1 \leq \Upsilon_{1,M}$ and $\Upsilon_2 \leq \Upsilon_{2,M}$, where $\Upsilon_{1,M}$ and $\Upsilon_{2,M}$ are positive constants. Therefore, using (28) and Young's inequality, (A.3) is written as

$$\begin{aligned} \dot{V}_1 \leq & -(k_1 - 1) e_1^2 - (k_1 - 1) e_2^2 - k_3 e_4^2 - k_3 e_5^2 \\ & - \left(\frac{1}{\kappa_1} - \frac{5}{2} \right) \bar{\omega}_1^2 - \left(\frac{1}{\kappa_2} - \frac{5}{2} \right) \bar{\omega}_2^2 \\ & - \frac{1}{2} \sum_{i=1}^2 t_i (\bar{W}_i^T \bar{W}_i + \bar{V}_i^T \bar{V}_i + \bar{\delta}_i^T \bar{\delta}_i) \\ & + g_1(t) N(\zeta_1) \dot{\zeta}_1 + \dot{\zeta}_1 + \sum_{i=1}^2 \frac{t_i}{2} (W_{i,M}^2 + V_{i,M}^2 + \delta_{i,M}^2) \\ & + 4\gamma_2^2 + \frac{1}{2} (\Upsilon_{1,M}^2 + \Upsilon_{2,M}^2). \quad (\text{A.5}) \end{aligned}$$

Choosing $k_1 = 1 + k_1^*$, $1/\kappa_1 = (5/2) + \kappa_1^*$, and $1/\kappa_2 = (5/2) + \kappa_2^*$ with $k_1^*, \kappa_1^*, \kappa_2^* > 0$ yields

$$\dot{V}_1 \leq -c_1 V_1 + g_1(t) N(\zeta_1) \dot{\zeta}_1 + \dot{\zeta}_1 + c_2, \quad (\text{A.6})$$

where $c_1 = \min[k_1^*, k_3, \kappa_1^*, \kappa_2^*, t_1, t_2]$ and $c_2 = \sum_{i=1}^2 (t_i/2) (W_{i,M}^2 + V_{i,M}^2 + \delta_{i,M}^2) + 4\gamma_2^2 + (1/2)(\Upsilon_{1,M}^2 + \Upsilon_{2,M}^2)$. Upon multiplication of (A.6) by $e^{\varsigma_1 t}$, it becomes

$$\frac{d}{dt} (V_1 e^{\varsigma_1 t}) \leq g_1(t) N(\zeta_1) \dot{\zeta}_1 e^{\varsigma_1 t} + \dot{\zeta}_1 e^{\varsigma_1 t} + c_2 e^{\varsigma_1 t}. \quad (\text{A.7})$$

Integrating it over $[0, t]$, we have

$$V_1(t) \leq c_3 + e^{-\varsigma_1 t} \int_0^t g_1 N(\zeta_1) \dot{\zeta}_1 d\lambda + e^{-\varsigma_1 t} \int_0^t \dot{\zeta}_1 e^{\varsigma_1 \lambda} d\lambda, \quad (\text{A.8})$$

where $c_3 = \sup_{\lambda \in [0, t]} [V_1(0) e^{-\varsigma_1 \lambda} + (c_2/\varsigma_1)(1 - e^{-\varsigma_1 \lambda})]$. Therefore, we can conclude that $V_1(t)$, ζ_1 , and $\int_0^t g_1 N(\zeta_1) \dot{\zeta}_1$ are bounded on $[0, t_f]$ according to Lemma 2.

Step 2. The time derivative of V_2 using (16)–(20) and (23)–(26) results in

$$\begin{aligned} \dot{V}_2 = & -k_2 e_3^2 + e_3 (\bar{\omega}_3 + \gamma_2 \tanh \beta_3) - k_3 e_6^2 + g_2(t) N(\zeta_2) \dot{\zeta}_2 \\ & + \dot{\zeta}_2 - \bar{\delta}_3^T \Theta_3 |e_6| \\ & + e_6 (\bar{W}_3^T \sigma(\bar{V}_3^T z_2) + \bar{W}_3^T \bar{\sigma}(\bar{V}_3^T z_2) \bar{V}_3^T z_2 + P_3) \\ & - \left(\frac{\bar{\omega}_3^2}{\kappa_3} + \bar{\omega}_3 \Upsilon_3 \right) - \bar{W}_3^T \Lambda_3^{-1} \bar{W}_3 \\ & - \bar{V}_3^T \Lambda_3^{-1} \bar{V}_3 - \bar{\delta}_3^T \Lambda_3^{-1} \bar{\delta}_3, \quad (\text{A.9}) \end{aligned}$$

where

$$\begin{aligned} \Upsilon_3 (e_3, e_4, e_5, e_6, \bar{\omega}_1, \bar{\omega}_2, \bar{\omega}_3, u_d, \bar{v}_d, r_d, \dot{u}_d, \dot{\bar{v}}_d, k_2, \gamma_2) \\ = & -k_2 \dot{e}_3 + \ddot{\psi}_a. \quad (\text{A.10}) \end{aligned}$$

On $V_1 = \mu_1$ and $V_2 = \mu_2$, $Y_3 \leq Y_{3,M}$, where $Y_{3,M}$ is a positive constant. Then, using (28) and Young's inequality, (A.3) is written as

$$\begin{aligned} \dot{V}_2 \leq & -(k_2 - 1)e_3^2 - k_3e_6^2 - \left(\frac{1}{\kappa_3} - 1\right)\omega_3^2 + g_2(t)N(\zeta_2)\dot{\zeta}_2 \\ & + \dot{\zeta}_2 - \frac{t_3}{2}(\bar{W}_3^T\bar{W}_3 + \bar{V}_3^T\bar{V}_3 + \bar{\delta}_3^T\bar{\delta}_3) \\ & + \frac{1}{2}(\gamma_2^2 + t_3(W_{3,M}^2 + V_{3,M}^2 + \delta_{3,M}^2) + Y_{3,M}^2). \end{aligned} \quad (\text{A.11})$$

Choosing $k_2 = 1 + k_2^*$ and $1/\kappa_3 = 1 + \kappa_3^*$ with $k_2^*, \kappa_3^* > 0$ yields

$$\dot{V}_2 \leq -c_4V_2 + g_2(t)N(\zeta_2)\dot{\zeta}_2 + \dot{\zeta}_2 + c_5, \quad (\text{A.12})$$

where $c_4 = \min[k_2^*, k_3, \kappa_3^*, t_3]$ and $c_5 = (1/2)(\gamma_2^2 + t_3(W_{3,M}^2 + V_{3,M}^2 + \delta_{3,M}^2) + Y_{3,M}^2)$. Similarly, we have

$$V_2(t) \leq c_6 + e^{-c_4 t} \int_0^t g_2 N(\zeta_2) \dot{\zeta}_2 e^{c_4 \lambda} d\lambda + e^{-c_4 t} \int_0^t \dot{\zeta}_2 e^{c_4 \lambda} d\lambda, \quad (\text{A.13})$$

where $c_6 = \sup_{\lambda \in [0, t_f]} [V_2(0)e^{-c_4 \lambda} + (c_5/c_4)(1 - e^{-c_4 \lambda})]$. Therefore, we can conclude that $V_2(t)$, ζ_2 , and $\int_0^t g_2 N(\zeta_2) \dot{\zeta}_2$ are bounded on $[0, t_f)$ according to Lemma 2.

Since V_1 and V_2 are bounded, one can easily prove from (25) that $\dot{\zeta}_1$ and $\dot{\zeta}_2$ are bounded. Then, $g_1(t)N(\zeta_1)\dot{\zeta}_1 + \dot{\zeta}_1$ and $g_2(t)N(\zeta_2)\dot{\zeta}_2 + \dot{\zeta}_2$ are bounded such that $|g_1(t)N(\zeta_1)\dot{\zeta}_1 + \dot{\zeta}_1| \leq c_7$ and $|g_2(t)N(\zeta_2)\dot{\zeta}_2 + \dot{\zeta}_2| \leq c_8$, where c_7 and c_8 are positive constants. From (A.6) and (A.12), $\dot{V}_1 < 0$ on $V_1 = \mu_1$ when $c_1 > (c_2 + c_7)/\mu_1$, and $\dot{V}_2 < 0$ on $V_2 = \mu_2$ when $c_4 > (c_5 + c_8)/\mu_2$. This means that $V_1 \leq \mu_1$ and $V_2 \leq \mu_2$ are invariant sets. Moreover, the boundedness of V_1 and V_2 implies that e_1 , e_2 , and e_3 are bounded. Therefore, we can conclude from (4) and (15) that the inequality in (29) holds. \square

Remark A.1. The choice of control gains has some suggestions as follows: (i) increasing k_1, k_3, t_1 , and t_2 and decreasing κ_1 and κ_2 help to increase c_1 , subsequently reducing the bound c_2/c_1 of error; (ii) increasing k_2, k_3 , and t_3 and decreasing κ_3 help to increase c_4 , subsequently reducing the bound c_5/c_4 of error.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2012RIA1A1041216).

References

- [1] M. L. Corradini, A. Monteriu, G. Orlando, and S. Pettinari, "An actuator failure tolerant robust control approach for an underwater remotely operated vehicle," in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, pp. 3934–3939, Orlando, Fla, USA, December 2011.
- [2] A. Alessandri, M. Caccia, and G. Veruggio, "Fault detection of actuator faults in unmanned underwater vehicles," *Control Engineering Practice*, vol. 7, no. 3, pp. 357–368, 1999.
- [3] M. Blanke, M. Staroswiecki, and N. E. Wu, "Concepts and methods in fault-tolerant control," in *Proceedings of the American Control Conference (ACC '01)*, pp. 2606–2620, Arlington, Va, USA, June 2001.
- [4] R. Patton, "Fault-tolerant control systems: the 1997 situation," in *Proceedings of the IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes*, pp. 1033–1054, Brussels, Belgium, 1997.
- [5] N. Sarkar, T. K. Podder, and G. Antonelli, "Fault-accommodating thruster force allocation of an AUV considering thruster redundancy and saturation," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 2, pp. 223–233, 2002.
- [6] T. Perez and A. Donaire, "Constrained control design for dynamic positioning of marine vehicles with control allocation," *Modeling, Identification and Control*, vol. 30, no. 2, pp. 57–70, 2009.
- [7] E. Omerdic and G. Roberts, "Thruster fault diagnosis and accommodation for open-frame underwater vehicles," *Control Engineering Practice*, vol. 12, no. 12, pp. 1575–1598, 2004.
- [8] B. Sun, D. Zhu, and L. Sun, "A tracking control method with thruster fault tolerant control for unmanned underwater vehicles," in *Proceedings of the 25th Chinese Control and Decision Conference (CCDC '13)*, pp. 4915–4920, Guiyang, China, May 2013.
- [9] X. Chen and W. W. Tan, "Tracking control of surface vessels via fault-tolerant adaptive backstepping interval type-2 fuzzy control," *Ocean Engineering*, vol. 70, pp. 97–109, 2013.
- [10] G. J. Toussaint, T. Basar, and F. Bullo, "Tracking for nonlinear underactuated surface vessels with generalized forces," in *Proceedings of the IEEE International Conference on Control Applications*, pp. 355–360, Anchorage, Alaska, USA, April 2000.
- [11] C. Kwan, D. M. Dawson, and F. L. Lewis, "Robust adaptive control of robots using neural network: global stability," *Asian Journal of Control*, vol. 3, no. 2, pp. 111–121, 2001.
- [12] S. N. Huang, K. K. Tan, and T. H. Lee, "Automated fault detection and diagnosis in mechanical systems," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1360–1364, 2007.
- [13] S. N. Huang and K. K. Tan, "Fault detection, isolation, and accommodation control in robotic systems," *IEEE Transactions on Automation Science and Engineering*, vol. 5, no. 3, pp. 480–489, 2008.
- [14] F. Caccavale, P. Cilibrizzi, F. Pierri, and L. Villani, "Actuators fault diagnosis for robot manipulators with uncertain model," *Control Engineering Practice*, vol. 17, no. 1, pp. 146–157, 2009.
- [15] R. D. Nussbaum, "Some remarks on a conjecture in parameter adaptive control," *Systems & Control Letters*, vol. 3, no. 5, pp. 243–246, 1983.
- [16] T. I. Fossen, *Marine Control Systems*, Marine Cybernetics, Trondheim, Norway, 2002.

- [17] K. D. Do and J. Pan, "Global tracking control of underactuated ships with nonzero off-diagonal terms in their system matrices," *Automatica*, vol. 41, no. 1, pp. 87–95, 2005.
- [18] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning based on FSCABC," *Information*, vol. 14, no. 3, pp. 687–692, 2011.
- [19] 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.
- [20] F. L. Lewis, S. Jagannathan, and A. Yesilderek, *Neural Network Control of Robot Manipulator and Nonlinear Systems*, Taylor & Francis, London, UK, 1999.
- [21] S. S. Ge and J. Wang, "Robust adaptive tracking for time-varying uncertain nonlinear systems with unknown control coefficients," *IEEE Transactions on Automatic Control*, vol. 48, no. 8, pp. 1463–1469, 2003.
- [22] R. Skjetne, T. I. Fossen, and P. Kokotovic, "Adaptive maneuvering, with experiments, for a model ship in a marine control laboratory," *Automatica*, vol. 41, no. 2, pp. 289–298, 2005.

Research Article

Multiagent Cooperative Learning Strategies for Pursuit-Evasion Games

Jong Yih Kuo,¹ Hsiang-Fu Yu,² Kevin Fong-Rey Liu,³ and Fang-Wen Lee¹

¹ Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan

² Department of Computer Science, National Taipei University of Education, Taipei, Taiwan

³ Department of Safety, Health and Environmental Engineering, Ming Chi University of Technology, New Taipei City, Taiwan

Correspondence should be addressed to Jong Yih Kuo; jykuo@ntut.edu.tw

Received 20 June 2014; Revised 17 September 2014; Accepted 12 October 2014

Academic Editor: Saeed Balochian

Copyright © 2015 Jong Yih Kuo 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 study examines the pursuit-evasion problem for coordinating multiple robotic pursuers to locate and track a nonadversarial mobile evader in a dynamic environment. Two kinds of pursuit strategies are proposed, one for agents that cooperate with each other and the other for agents that operate independently. This work further employs the probabilistic theory to analyze the uncertain state information about the pursuers and the evaders and uses case-based reasoning to equip agents with memories and learning abilities. According to the concepts of assimilation and accommodation, both positive-angle and bevel-angle strategies are developed to assist agents in adapting to their environment effectively. The case study analysis uses the Recursive Porous Agent Simulation Toolkit (REPAST) to implement a multiagent system and demonstrates superior performance of the proposed approaches to the pursuit-evasion game.

1. Introduction

A multiagent system (MAS) comprises a set of agents that interact with each other. These agents may either share a common goal or have contradictory objectives [1, 2]. This work deals with cooperative agents trying to achieve a common goal. When coordinating multiple agents as a team for the same task, the agents must have the ability to handle unknown and uncertain situations and take the success of the whole team into account.

A multiagent pursuit-evasion game involves guiding one group of agents (pursuers) to cooperate with each other to catch another group of agents (evaders). However, this game varies with the type of environment in which it is played (e.g., plane, grid, and graph), the knowledge of the players (e.g., the evaders' position and strategy), the controllability of their motions (is there a limit on their speed? and can they make turns whenever they want?), and the meaning of a capture (are the evaders to be intercepted, seen, or surrounded?). Being complex and dynamic, pursuit-evasion problems are difficult to solve [3]. To address such problem, the robotics community proposed several models [4, 5]. In

these models, the motion of the evader is usually modeled by a stochastic process. There has been growing interest in modeling the game, in which the evader is intelligent and has certain sensing capabilities [6]. Solutions to the problem proposed in other studies [7–9] included competitive coevolution, multiagent strategies, and multiagent communication algorithms.

The hybrid learning approach to RoboCup [10] includes a coach agent and multiple moving agents. Using case-based reasoning and genetic algorithms (CBR-GA), the coach agent decides on a strategy goal and assigns tasks to the moving agents. Every moving agent then executes its respective task to achieve the strategy goal. The proposed method also includes two kinds of agents (pursuer and evader). Unlike the hybrid learning approach to RoboCup, the proposed method does not use the coach agent to guide pursuers in catching evaders. The pursuers must search for the evaders themselves. If other evaders/pursuers are not in the sensing area of a pursuer, the pursuer does not know their positions. The pursuer thus uses the case-based reasoning with assimilation and accommodation to catch evaders.

Owing to the uncertain environment and the dynamic information of agents at each moment, previous research has mainly examined how to improve the efficiency of communication and accuracy in the learning process. In a different manner, this study focuses on agents with a mental state and the ability to plan evolution by using strategic modules and combining learning methods. Armed with such capability, the agents can adapt to the dynamic environment.

The rest of the paper is organized as follows. Section 2 reviews the background and related works of pursuit-evasion problems. Section 3 introduces the agent's adaptive learning process. Section 4 describes the case study of the pursuit-evasion game. The conclusions and directions for future work are presented in the last section.

2. Background and Related Work

Pursuit-evasion problems have long been studied from the perspective of differential game theories. Nishimura and Ikegami observed random swarming and other collective predator motions in a prey-predator game model [11]. Jim and Giles [12] proposed a simple effective predator strategy, which could enable predators to move to the closest capture position. This strategy does not prove to be very successful because the predators may block each other when they try to move to the same capture position. Haynes and Sen [13] used genetic programming to evolve strategies for both predators and preys. Hládek et al. [14] developed a multiagent control system using fuzzy inference for a group of two-wheeled mobile robots to execute a common task. They defined a pursuit-evasion task using fuzzy sets to establish a framework for inserting and updating expert knowledge in the form of rules by an inference system. Antoniadis et al. [3] proposed several pursuit algorithms for solving complex multiplayer pursuit-evasion games. Their work revealed that reducing sensing overlap between pursuers and avoiding overassignment of pursuers to target locations could improve the capture time. According to the Contract Net Protocol (CNP) [15], Zhou et al. [16] proposed a multiagent cooperative pursuit algorithm and extended the CNP by improving alliance decision and forming dynamic alliance. They used the support decision matrix that includes credits of agents, degrees of hookup, and degrees of self-confidence to help agents make decisions during the negotiation process.

Rao and Georgeff [17] proposed the belief-desire-intention (BDI) model for the agents in a distributed artificial intelligence environment. As its name implies, the BDI agent model involves belief, desire, and intention. In addition, the assimilation and accommodation approaches to the pursuit-evasion game were proposed by Piaget [18]. These two approaches equip an agent with a mental state and the ability to plan an evolution process. Using different learning methods, the agent can also adapt to the dynamic environment effectively.

2.1. BDI Agent Model. On the foundation of BDI, Kuo et al. [19, 20] proposed an agent, which can be completely specified to fulfill its intentions by events it perceives, actions

it performs, beliefs it holds, goals it adopts, and plans it has. A goal module describes the goals that an agent may adopt, as well as the events to which it can respond. A belief module includes the information about the internal state that an agent of a certain class holds, the strategies it may perform, and the environment it is in. A plan module generates the plans that an agent may employ to achieve its goals. A plan is a sequence of actions or strategies derived through a reasoning mechanism.

In the proposed approach, *belief* denotes the agent's knowledge of the environment, including the orientations and locations of evaders; *desire* represents the agent's wish to catch all evaders, and *intention* stands for the agent's plan of actions.

2.2. Assimilation and Accommodation. According to Piaget's cognitive theory [18], each recognition system aims at equilibration of the inconsistent information about the world. Thus, an organism seeks not only adaptation (harmony of organism and world) but also organization (harmony within itself). Assimilation and accommodation represent forms of the maintenance and modification of these cognitive schemata [21]. Takamuku and Arkin [22] applied assimilation on domestic robots for social learning, enabling the robots to perform well under various situations. This study proposes both bevel-angle and positive-angle accommodation strategies for modifying the agent's cognitive structure.

2.3. Pursuit-Evasion Game. In a pursuit-evasion game, pursuers try to capture evaders by besieging them from all directions in a grid world. The game focuses mainly on the effectiveness of structures, with varying degrees of pursuers' cooperation and control, to entrap evaders efficiently [13]. The first mathematical formulation of graph searching was proposed by Parsons [23] in 1978. The formulation was inspired by an earlier research by Breisch [24] who put forward an approach to finding an explorer lost in a complicated system of dark caves.

Furthermore, the pursuit-evasion game varies with different approaches to searching evaders as described in the following.

2.3.1. Node Search and Mixed Search. Ferrari [25] proposed a new family for path planning algorithms. One algorithm is for searching the goal, and the other is for searching the obstacles. These algorithms can be utilized to parameterize easily the potential scale length and strength, thus providing better control over the moving object path.

To address some unique demands from the game domain, Walsh and Banerjee [26] presented a new algorithm, called "VRA*" algorithm, for path finding on game maps, and proposed also the extension of a postsmoothing technique. Wong et al. presented a Bee Colony Optimization (BCO) algorithm for the symmetrical Traveling Salesman Problem (TSP) [27]. The BCO model is constructed algorithmically according to the collective intelligence observed from the foraging behavior of bees. The algorithm is integrated with

a fixed-radius near-neighbor 2-opt (FRNN 2-opt) heuristic to further improve prior solutions generated by the BCO model.

Raboin et al. [28] considered multiagent pursuit scenarios in which there is a team of tracker agents and a moving target agent. The trackers observe continuously the target until the target is at least one tracker's observation range at the end of the game. Their study described formalism and algorithms for game-tree search in partially observable Euclidean space.

Gerkey et al. [29] introduced a new class of searchers, the φ -searchers, which can be readily instantiated as a physical mobile robot, and proposed the first complete search algorithm for a single φ -searcher. They also showed how this algorithm can be extended to handle multiple searchers and gave various examples of computed trajectories. Their work aimed at coordinating teams of robots to execute tasks in application domains, such as clearing a building, for reasons of security or safety.

2.3.2. Game Theoretic. Game theoretic approaches to patrolling have increasingly become an interesting topic in recent years. Basilio et al. [30] presented a game theoretic scheme to determine the optimal patrolling strategy for a mobile robot that operates in environments with arbitrary topologies.

Prior research [31] proposed a game theory-based approach, which uses a multirobot system to perform multitarget search in a dynamic environment. A dynamic programming equation is employed to estimate the utility function, which considered the a priori probability map, travel costs, and current decisions of other robots. According to this utility function, a utility matrix can be calculated for an N -robot nonzero-sum game. Thus, pure Nash equilibrium and mixed-strategy equilibrium can be utilized to guide the robots in making their decisions.

3. Adaptive Agent Learning

This section introduces the adaptive agent model, the refinement of previously developed agent models [19, 20]. This adaptive agent model contains a hybrid approach for a multi-agent learning method. In addition, this model enables agents to learn and accumulate their experience and knowledge from other agents or the environment.

3.1. Adaptive Agent Model. The agent model, shown in Figure 1, is a cooperative learning model responsible for controlling the learning process. A goal module returns the goals that an agent may possibly adopt and the events to which it can respond. A belief module describes the information about the environment, the internal state that a certain class of agents may hold, and the strategies or tactics that the agent may perform. A plan module returns plans that are possibly employed to achieve the goals of the agent. A plan is a sequence of actions or strategies derived through reasoning. There are two types of ontology that provide the domain-specific and issue-specific knowledge, respectively.

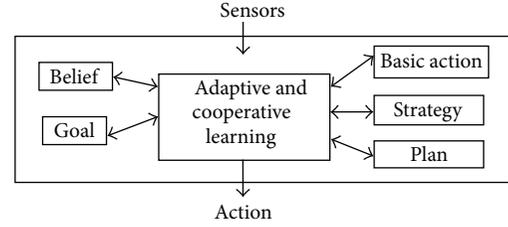


FIGURE 1: Agent model.

The mental state of an agent is represented by first-order language as follows:

$$\begin{aligned}
 FOTerm & ::= \langle Const \rangle | \langle FOV \rangle | \\
 & \quad \text{functor } \langle FuncSym \times seq FOTerms \rangle, \quad (1) \\
 Atom & ::= \langle Predicate \times seq FOTerms \rangle.
 \end{aligned}$$

A first-order term ($FOTerm$) is defined by a set of constants ($\langle Const \rangle$), first-order variables ($\langle FOV \rangle$), and functions ($\text{functor } \langle FuncSym \times seq FOTerms \rangle$). An atom consists of a predicate and a set of $FOTerms$.

3.1.1. Belief. The beliefs of an agent describe the situations where the agent is. The beliefs are specified by a belief base, which contains the information the agent believes about the world, as well as the information that is internal to the agent. The agent is assumed to have beliefs about its task and its environment. A belief is represented by a set of well-formed formulas such as

$$Belief ::= \langle Atom \rangle | \neg \langle Atom \rangle | \langle Atom \rangle [\wedge \langle Atom \rangle]. \quad (2)$$

For instance, a belief that there is an obstacle ($obstacle_id$) at a certain location ($X1, Y1$) can be represented as ($obstacle_id, X1, Y1$).

3.1.2. Goal. A goal is an agent's desire and describes the states of affairs that an agent would like to realize. For instance, $CatchAllEvader(True)$ stands for a pursuer's goal to catch all evaders. A goal is represented by a set of well-formed formulas such as

$$Goal ::= \text{achieved } \langle Atom \rangle. \quad (3)$$

3.1.3. Basic Actions. Basic actions are generally used by an agent to manipulate its environment. Before performing the basic actions, certain beliefs should be held. The beliefs of the agent must be updated after the execution of actions. Basic actions are the only entities that can change the beliefs of an agent. Let α be an action with parameters x , and let $\varphi, \psi \in \text{belief}$. The programming constructs for basic actions are then expressed as

$$\{\varphi\} \alpha(x) \{\psi\}, \quad (4)$$

where φ and ψ are preconditional and postconditional, respectively. The precondition indicates that an action cannot

be performed only if certain beliefs are held. For example, a pursuer is assumed to catch an evader from location $(X0, Y0)$ to location $(X1, Y1)$ with a basic action $MoveTo(X1, Y1)$. Before this action is executed, the pursuer must be at position $(X0, Y0)$, denoted by $pursuer(self, X0, Y0)$. The condition that the pursuer senses the evader at position $(X1, Y1)$ is denoted by $evaderAt(X1, Y1)$. This action is then defined in terms of the following beliefs:

$$\{pursuer(self, X0, Y0), evaderAt(X1, Y1)\}. \quad (5)$$

After executing an action, an agent must update its beliefs to make its postcondition true. According to the above example, after performing the catch action, the agent updates its beliefs and obtains

$$\{pursuer(self, X1, Y1), evaderAt(X2, Y2)\}, \quad (6)$$

where $(X2, Y2)$ is an evader's new location.

3.1.4. Strategies. The strategy of a player refers to one of the options that can be chosen in a setting. The choice depends not only on the agent's own actions but also on other agents' actions. A prior study [28] proposed a pure strategy in terms of a function. Given an agent's current information set, the function returns the agent's next move (i.e., its change in location between time t_j and t_{j+1}). For example, if an evader has a pure strategy σ_0 , then $\theta_0(t_{j+1}) = \theta_0(t_j) + \sigma_0(I_0(t_j))$, where $\theta_0(t_j)$ is the location of tracker agent 0 at time t_j and $I_0(t_j)$ is the tracker's information set at time t_j . Suppose that a pursuer has a pure strategy set $\sigma = (\sigma_1, \dots, \sigma_k)$ and the locations of the tracker agents $\theta(t) = \{\theta_1(t), \dots, \theta_k(t)\}$. Then, $\theta(t_{j+1}) = \theta(t_j) + \sigma(I_0(t_j))$ for each time t_j . The strategy is further refined by adding an algorithm for selecting a plan according to different conditions. The algorithm can be represented by a function that generates a plan. The condition includes the beliefs of agents or the current situation of the environment. The proposed strategy is denoted by

$$Strategy ::= \langle conditions \rangle function. \quad (7)$$

3.1.5. Plans. A plan is a method for an agent to approach its goal and is generated according to the agent's strategy. A plan consists of beliefs, actions, and rewards, as expressed in the following equation:

$$Plan ::= \langle belief \rangle \langle basic\ action \rangle [\langle basic\ action \rangle] \langle reward \rangle. \quad (8)$$

A plan can be one or a set of actions for an agent, and the value of the plan is determined by the reward.

3.2. Case-Based Reasoning. According to previous studies [32, 33], a general case-based reasoning (CBR) cycle may include retrieving the most similar cases for a problem, reusing the information and knowledge in those cases to solve the problem, revising the proposed solution, and retaining the parts of this experience likely to be useful for future problem solving.

In order to solve a new problem, the cases which contain the most useful knowledge have to be identified first. Since the utility of a case cannot be evaluated directly a priori, the similarity between problem descriptions is used in heuristics to estimate the expected utility of the cases. Therefore, the quality of this measure is crucial for the success of CBR applications. There are numerous similarity measures in use today. They are used not only in CBR but also in other fields, including data mining, pattern recognition, genetic algorithm, and machine learning. Euclidean distance is a typical similarity measure for objects in a Euclidean space [34]. According to Euclidean distance, this work devises a simple distance function to evaluate the similarity between two cases.

3.2.1. Case Description. A case stands for an agent's mental state and the result of its past output. Each case consists of a goal, beliefs, an action, a plan, and a reward. The goal is what the agent wants to achieve or realize. The beliefs describe the situation the agent is in. The action is what the agent does under that situation. The plan is the approach the agent takes to achieve the goal. The reward serves to evaluate the result of the plan.

3.2.2. Similarity Relation. Consider the following:

$$Sim_{ab} = \sum_{i=1}^n w_i \times (|a_i - b_i|). \quad (9)$$

This study proposes an evaluation function (Sim), shown as (9), for measuring the similarity between cases a and b . In this equation, a and b represent a new problem and a case in a case base, respectively. The variable n is the number of features in case a . The weight variable w_i stands for the importance of the i th feature. The weight vector is freely defined by a user. a_i and b_i are the i th features of cases a and b , respectively. In the case-retrieving stage of the CBR approach, the case with the smallest (Sim) value is always retrieved for reuse.

3.3. Probabilistic Framework. Suppose that a finite two-dimensional environment EX with n_c square cells contains an unknown number of fixed obstacles. We also assume that the environment has n_p pursuers and n_e evaders. Let x_{pk} and x_{ei} be the cells occupied by pursuer k and evader i , where $1 \leq i \leq n_p$ and $1 \leq j \leq n_e$. The pursuers and evaders are restricted to move to cells not occupied by obstacles. Each pursuer collects information about EX at discrete time instances $t \in T = \{1, 2, \dots, t_{end}\}$.

Define $x_{pk}(t) \subset EX$ as the cell of pursuer k at time t and $x_{ei}(t) \subset EX$ as the cell of evader i at time t . $o(t)$ is a set of cells where obstacles are detected. Let $U(x_{ei}(t))$ denote the one-step reachable set for the evader when the evader is in cell x

at time t . Then the probability of the evader being in cell x at time $t + 1$ is given by the following (10):

$$p(x, t + 1 | x_{e_i}(t)) = \begin{cases} \frac{1}{|U(x_{e_i}(t)) - o(t)|} & x \in U(x_{e_i}(t)) \\ 0 & x \in o(t), \end{cases} \quad (10)$$

where $p(x, t + 1 | x_{e_i}(t))$ represents the probability of the evader being in cell x at time $t + 1$ according to the location where the pursuer detects the evader at time t .

3.4. Strategies for Plan Generation. When the distance between the retrieved case and the new case is smaller than a threshold, the above approach cannot generate a useful plan for the new case. To overcome this problem, this study proposes two plan-generation strategies, local-max strategy and local-cooperative strategy.

3.4.1. Local-Max Strategy. This strategy is used only when there is a single pursuer for evaders. Let $S_k(y)$ be the set of all cells that lie within the sensing area of a pursuer k at cell y . The total evasion probability of the evaders at time t associated with the pursuer is then obtained by the following (11):

$$P(y, t) = \sum_{z \in S_k(y)} [p(z, t + 1 | x_{e_i}(t))]. \quad (11)$$

By computing the evasion possibilities of all sensing cells, pursuer k moves to cell $x_{p_k}(t + 1)$ that has the highest total evasion probability expressed as follows:

$$x_{p_k}(t + 1) = \arg \max_{y \in U(x_{p_k}(t))} [P(y, t)]. \quad (12)$$

3.4.2. Local-Cooperative Strategy. This strategy is for pursuers that cooperate with each other to catch the same evader. Initially, pursuers need to decide whether they can cooperate with each other or not. Let $d(x, y)$ be the Manhattan distance from cell x to cell y ; that is, if cells x and y lie in a two-dimensional space with coordinates (x_1, x_2) and (y_1, y_2) , the Manhattan distance is calculated using the following equation:

$$d(x, y) = \max(|x_1 - y_1|, |x_2 - y_2|). \quad (13)$$

Let S_k be the farthest sensing cell of a pursuer k . For example, if the sensing area of a pursuer k is a $5 * 5$ square and the pursuer stands at the center of the area, the farthest distance that the pursuer can detect is two squares. If there exists two or more pursuers and one evader corresponding with $d(x_{p_k}(t), x_{e_i}(t)) \leq s_k$ and $d(x_{p_k}(t), x_{e_i}(t)) \leq s_k$, the pursuers can cooperate with each other.

Suppose that two or more pursuers execute the local-cooperative strategy. Let $\text{Overlap}(x, y)$ be the number of overlap cells within the one-step reachable set (a $3 * 3$ region) around cells x and y . The overlap function is illustrated in Figure 2. The yellow grid represents the one-step reachable

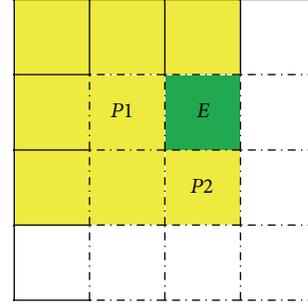


FIGURE 2: Number of overlap cells.

set for a pursuer $P1$, the dashed grid depicts the one-step reachable set for a pursuer $P2$, and the green grid represents the position of an evader E . The number of overlap cells thus equals four (i.e., the return value of $\text{Overlap}(x, y)$).

Once the pursuers can cooperate with each other, the pursuers randomly select one of them as a leader. Other pursuers can determine their locations at time $t + 1$ by the following (14):

$$x_{p_k}(t + 1) = \arg \min_{y \in U(x_{p_k}(t))} [\text{Overlap}(x_{p_k}(t + 1), y)], \quad (14)$$

where $x_{p_k}(t + 1)$ is the location of the leader. To avoid pursuers moving to the same location, each pursuer finds the minimum overlap area.

3.5. Reward Calculation. Each case has its own reward for evaluating its result. The higher the reward of a case is, the more likely the goal will be achieved. In this study, the reward is calculated using the following equation:

$$R = |S_k(y) \cap U(x_{e_i}(t))|. \quad (15)$$

Reward R represents the number of one-step reachable sets for an evader in a pursuer's sensing area. The reward value is between 0 and 10. If an evader is caught, then the reward of the case will be 10.

3.6. Assimilation and Accommodation. Piaget's cognitive theory [18] reveals that assimilation refers to the tendency to interpret experience as much as possible through existing cognitive structure of knowing. When an agent faces a new situation, it generates a new plan with the current cognitive structure, also called its strategy, to direct itself what to do next. The entire process is called assimilation. In other words, an agent assimilates a new situation with its current cognitive structure. However, if the current cognitive structure cannot explain the environment (i.e., the current cognitive structure cannot get the equilibration of inconsistent information about the world), an agent has to use the strategy of accommodation. Accommodation refers to the realization that the current structure is insufficient for adequate understanding of the world and that the current structure must be changed until it can assimilate the new situation.

This study proposes two accommodation methods, bevel-angle and positive-angle strategies, for modifying a cognitive structure. After performing accommodation, an agent employs the modified strategy to assimilate the new situation. By adjusting constantly its cognitive structure, the agent can adapt more effectively to the environment.

3.6.1. Bevel-Angle Strategy. Suppose that an evader's location is in the one-step reachable set for a pursuer, and the evader is at a bevel angle of the pursuer. The pursuer randomly chooses one of the following strategies when the evader is at 45° ($\pi/4$), 135° ($3\pi/4$), 225° ($5\pi/4$), and 315° ($7\pi/4$) of the pursuer. Let $\deg(x_{p_k}(t), x_{e_i}(t))$ be the angle between the pursuer k and evader i , where $x_{e_i}(t)$ is the position of the evader and $x_{p_k}(t)$ is the position of the pursuer at time t .

Strategy 1. According to (16), the pursuer moves to the evader's position at time $t + 1$:

$$\begin{aligned} x_{p_k}(t+1) &= x_{e_i}(t), \\ \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\ \deg(x_{p_k}(t), x_{e_i}(t)) &\in \left\{ \frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4} \right\}. \end{aligned} \quad (16)$$

Strategy 2. According to (17), the pursuer moves to 45° of the evader's position at time $t + 1$. Let x_j be an element of the one-step reachable set for the pursuer k when it is in cell x at time t :

$$\begin{aligned} x_{p_k}(t+1) &= x_j, \\ \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_{e_i}(t)) &\in \left\{ \frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4} \right\} \\ \cap x_j &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_j) &= \deg(x_{p_k}(t), x_{e_i}(t)) + 45. \end{aligned} \quad (17)$$

Strategy 3. According to (18), the pursuer moves to -45° of the evader's position at time $t + 1$:

$$\begin{aligned} x_{p_k}(t+1) &= x_j, \\ \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_{e_i}(t)) &\in \left\{ \frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4} \right\} \\ \cap x_j &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_j) &= \deg(x_{p_k}(t), x_{e_i}(t)) - 45. \end{aligned} \quad (18)$$

Strategy 4. According to (19), the pursuer stays in the same place at time $t + 1$:

$$\begin{aligned} x_{p_k}(t+1) &= x_{p_k}(t), \\ \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\ \deg(x_{p_k}(t), x_{e_i}(t)) &\in \left\{ \frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4} \right\}. \end{aligned} \quad (19)$$

3.6.2. Positive-Angle Strategy. Suppose that an evader's location is in the one-step reachable set for a pursuer, and the evader is at a positive angle of the pursuer, where the angle includes 0° , 90° ($\pi/2$), 180° (π), and 270° ($3\pi/2$).

Strategy 1. According to (20), the pursuer moves to the evader's position at time $t + 1$:

$$\begin{aligned} x_{p_k}(t+1) &= x_{e_i}(t), \\ \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\ \deg(x_{p_k}(t), x_{e_i}(t)) &\in \left\{ 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2} \right\}. \end{aligned} \quad (20)$$

Strategy 2. According to (21), the pursuer moves to 45° of the evader's position at time $t + 1$:

$$\begin{aligned} x_{p_k}(t+1) &= x_j, \\ \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_{e_i}(t)) &\in \left\{ 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2} \right\} \\ \cap x_j &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_j) &= \deg(x_{p_k}(t), x_{e_i}(t)) + 45. \end{aligned} \quad (21)$$

Strategy 3. According to (22), the pursuer moves to -45° of the evader's position at time $t + 1$:

$$\begin{aligned} x_{p_k}(t+1) &= x_j, \\ \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_{e_i}(t)) &\in \left\{ 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2} \right\} \\ \cap x_j &\in U(x_{p_k}(t)) \\ \cap \deg(x_{p_k}(t), x_j) &= \deg(x_{p_k}(t), x_{e_i}(t)) - 45. \end{aligned} \quad (22)$$

Strategy 4. According to (23), the pursuer moves to 90° of the evader's position at time $t + 1$.

$$\begin{aligned}
 x_{p_k}(t+1) &= x_j, \\
 \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\
 \cap \deg(x_{p_k}(t), x_{e_i}(t)) &\in 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2} \\
 \cap x_j &\in U(x_{p_k}(t)) \\
 \cap \deg(x_{p_k}(t), x_j) &= \deg(x_{p_k}(t), x_{e_i}(t)) + 90.
 \end{aligned} \tag{23}$$

Strategy 5. According to (24), the pursuer moves to -90° of the evader's position at time $t + 1$:

$$\begin{aligned}
 x_{p_k}(t+1) &= x_j, \\
 \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\
 \cap \deg(x_{p_k}(t), x_{e_i}(t)) &\in 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2} \\
 \cap x_j &\in U(x_{p_k}(t)) \\
 \cap \deg(x_{p_k}(t), x_j) &= \deg(x_{p_k}(t), x_{e_i}(t)) - 90.
 \end{aligned} \tag{24}$$

Strategy 6. According to (25), the pursuer stays in the same place at time $t + 1$:

$$\begin{aligned}
 x_{p_k}(t+1) &= x_{p_k}(t), \\
 \text{if } x_{e_i}(t) &\in U(x_{p_k}(t)) \\
 \deg(x_{p_k}(t), x_{e_i}(t)) &\in 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}.
 \end{aligned} \tag{25}$$

Combining both the bevel-angle and positive-angle strategies, the pursuer obtains 24 different plans to adapt to the environment effectively.

3.7. Plan Evolution Process. Plan evolution plays an important role in an agent's life cycle. When a pursuer finds an evader in the sensing area, it generates a pursuit plan. Initially, the pursuer searches its case base for similar cases. If similar cases exist, the pursuer chooses one of them as the plan. Otherwise, the pursuer uses one of the abovementioned strategies to generate a plan. When the pursuer faces a new state, the pursuer will adjust the plan accordingly. Figure 3 shows the agent's plan evolution process.

(1) *Problem Analysis.* Before determining the location in the next step, a pursuer analyzes the environmental state and updates its goal with information from the environment. The pursuer chooses a plan according to its mental state.

(2) *Case Retrieval.* Comparing the similarity between the result of the analysis in *Step 1* and the cases in the case base,

the pursuer retrieves similar cases by the (Sim) function and sorts them according to their similarity levels.

(3) *Similarity Calculation.* Similarity is classified into low and high levels. It is possible that the pursuer retrieves a case with high-level similarity but obtains unsatisfactory results, and thus a reward R is added to prevent such. If the case is with high-level similarity and the reward R exceeds a predefined constant ε , the pursuer executes *Step 4*. If the case is with low-level similarity or the reward R is smaller than the predefined constant ε , the pursuer executes *Step 5*.

(4) *Case Reuse.* If the retrieved case is with high-level similarity and its result is acceptable, the pursuer reuses the case with revisions and then goes to *Step 10*.

(5) *Strategy Assimilation.* If no similar case exists in the case base or the result of the similar case is unsatisfactory, the pursuer uses strategy assimilation; that is, the pursuer generates a suitable plan with the current strategy (*Steps 6–8*) according to the situation.

(6) *Cooperation.* Before using the strategy module, the pursuer decides whether to cooperate with the others or not. If the decision is to cooperate with the others in the situation, then the local-cooperative strategy (i.e., *Step 8*) is executed. Otherwise, the pursuer executes the local-max strategy (i.e., *Step 7*).

(7) *Application of the Local-Max Strategy.* If the pursuer decides not to cooperate with the others, it executes the local-max strategy and moves to the cell in the one-step reachable set with the highest probability of containing an evader. Then, go to *Step 9*.

(8) *Application of the Local-Cooperative Strategy.* If the cooperative mode is taken, the pursuer executes the local-cooperative strategy and moves to the cell which has the minimum overlap sensing area.

(9) *Reward Calculation.* The pursuer evaluates the plan and calculates a reward accordingly.

(10) *Plan Transformation.* The pursuer represents the plan by beliefs, actions, and rewards.

(11) *Strategy Assessment.* The pursuer evaluates whether the current strategy is suitable or not. The pursuer counts the number C of consecutive failures in catching an evader by using the strategy. If C is smaller than a constant σ , the strategy is effective. Then, go to *Step 13*. Otherwise, the strategy is accommodated by *Step 12*.

(12) *Strategy Accommodation.* The pursuer modifies the strategy to generate a new plan for the current environment.

(13) *Case Revision.* The actions and reward of the case are revised after the problem is solved.

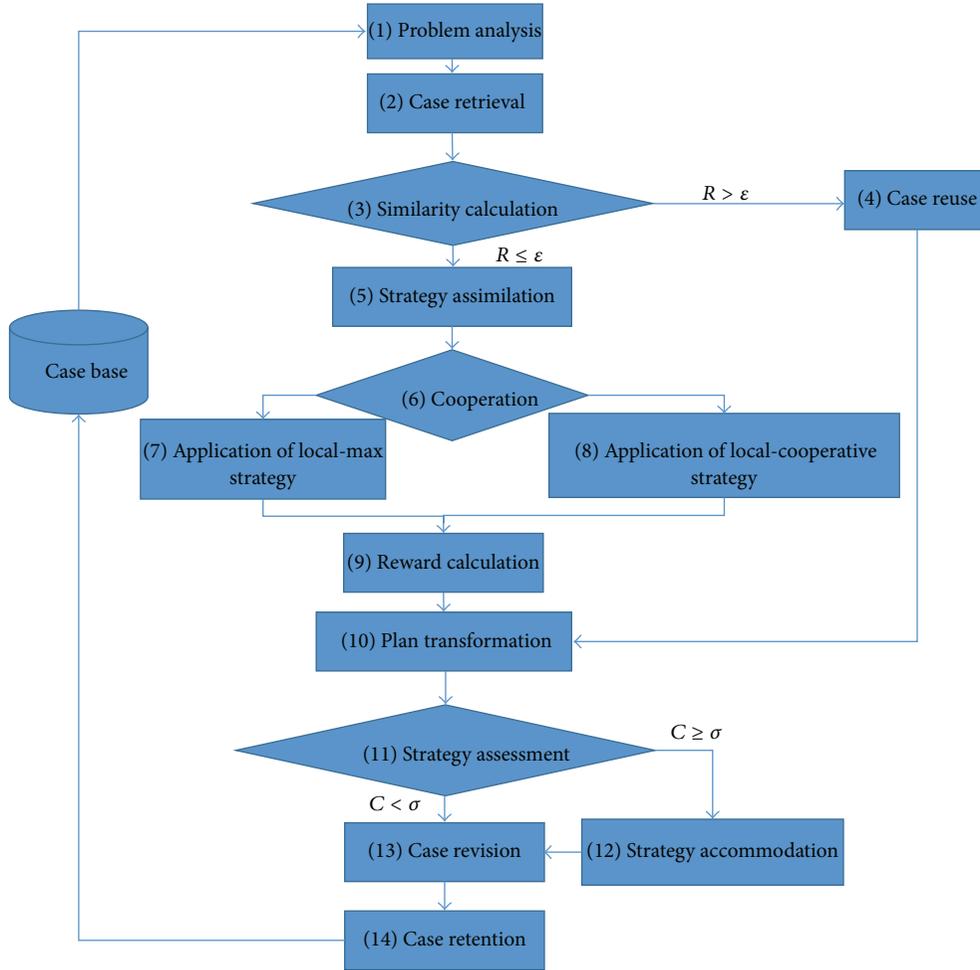


FIGURE 3: Plan evolution process.

(14) *Case Retention*. Parts of the experience that are likely to be useful for future problem solving are retained.

4. Case Study

This study uses a pursuit-evasion game to demonstrate the proposed approach and implements a multiagent system on the Recursive Porous Agent Simulation Toolkit (REPAST) [35]. Figure 4 shows the GUI of the system, where the blue, red, and black circles represent pursuers, evaders, and obstacles, respectively. The yellow region depicts the sensing area of the pursuers.

4.1. Problem Formulation. The system contains multiple pursuing robots, which search for multiple evading robots in a two-dimensional environment. The physical world is abstracted as a grid world, and the game is played according to the following settings and rules.

- (i) The number of pursuers is n . The number of evaders is unknown to the pursuers.

- (ii) Each pursuer has a sensing area comprising $5 * 5$ cells, and the pursuer is at the center of the area, as shown in Figure 4.

- (iii) During each moving step, the agents are allowed to either move one cell away from their current cells or stay in their current cells.

- (iv) Each pursuer has no information about the environment other than that of its own 5-by-5 sensing area.

- (v) All pursuers use the same case base.

- (vi) An evader is captured when it occupies the same cell as a pursuer, and it will be removed from the environment.

- (vii) The game ends when all evaders have been captured.

4.2. Evasion Policy. When an evader is caught, the system first checks whether the sum of the caught and uncaught evaders is smaller than the total number of the predefined evaders. If so, the system generates a new evader and chooses its type of movement. Otherwise, the system does nothing. Clearly, when the sum of the caught and uncaught evaders

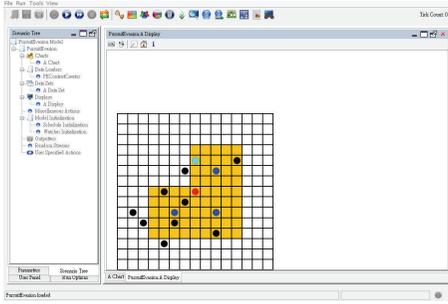


FIGURE 4: Implementation of multiagent system.

equals the number of the predefined evaders, the system stops generating new evaders and the uncaught evaders do not increase. The uncaught evaders even decrease with the successful evader captures. The game thus ends finally. In addition, when the system selects a movement type for a new evader, the choice varies with different experiments, which in turn affects the outcome of the experience. This study investigates four types of movement, namely, random, clockwise, counterclockwise, and smart movements.

Random Movement. An evader randomly selects a location that can be reached in one step or just stays in the same location. The details of the algorithm are shown in Algorithm 1.

Clockwise Movement. An evader moves clockwise in a square path. The evader initially selects the direction of its first step and sets up a distance randomly. If the evader runs into an obstacle on the way, it changes its direction automatically. Algorithm 2 shows the algorithm.

Counterclockwise Movement. This movement type is similar to the clockwise movement except that an evader moves counterclockwise. The algorithm is shown in Algorithm 3.

Smart Moving. Evaders try actively to avoid pursuers. Each evader has a square sensing area. Each side of the square is three cells long. If the evader finds any pursuer in the area, it attempts to move to the location that has fewer pursuers in the one-step reachable set of the area. Algorithm 4 presents the algorithm.

4.3. Results and Discussion. This study designs several experiments to compare the performance of the proposed approach, Hysteretic Q-Learning [36], and Reinforcement-Learning fusing (RL-Fusing) [37]. All these approaches are the learning techniques for multiagent systems. Each experiment has three pursuers and one evader. In addition, the experiments run in “time step” unit. One round is considered finished when the evader has been caught. The performance results are the average values obtained by running 30 rounds with the current learning policy after 100 training episodes for each algorithm.

4.3.1. Comparison of Approaches. Maignon et al. presented a decentralized reinforcement learning algorithm for

independent learning robots called Hysteretic Q-Learning [36]. This algorithm computes a better policy in a cooperative multiagent system without additional information or communication between agents. Hysteretic Q-Learning updates the equation for an agent i executing the action a_i from states ending up in state s' as follows:

$$\delta \leftarrow r + \gamma \max_{a'} Q_i(s', a') - Q_i(s, a_i),$$

$$Q_i(s, a_i) \leftarrow \begin{cases} Q_i(s, a_i) + \text{inc}\delta & \text{if } \delta > 0 \\ Q_i(s, a_i) + \text{dec}\delta & \text{else,} \end{cases} \quad (26)$$

where $Q_i(s, a_i)$ is the action value function of state s and action a_i , r is the reward it receives, $\gamma \in [0; 1]$ is the discount factor, and inc and dec are the rates of increase and decrease in Q -values, respectively. Partalas et al. proposed the RL-Fusing approach that uses coordinated actions and a fusing process to guide the agents [37]. The fusion process combines the decisions of all agents and then outputs a global decision, which the agents must follow as a team. Algorithm 5 shows the RL-Fusing algorithm. The set of strategies means the plans.

4.3.2. Obstacles. The first experiment compares the performance of the proposed approach, Hysteretic Q-Learning, and RL-Fusing in environments with/without obstacles. The evader is assumed to move randomly in every round. In the environment with obstacles, 10 obstacles are randomly set up initially, and no evader is trapped in the obstacles.

Table 1 shows the performance results. As can be seen, all three methods require more capture steps in the environment with obstacles, because pursuers need extra time to detect and avoid obstacles. Hysteretic Q-Learning takes the greatest number of capture steps in both environments because it does not support pursuer cooperation. Furthermore, RL-Fusing also takes more capture steps than the proposed scheme. Although RL-Fusing involves cooperation between pursuers, the generated strategies sometimes cause a pursuer to miss evaders in its sensing area. The pursuer thus needs to search again and requires more capture steps. In contrast, the proposed approach utilizes not only the cooperative strategy but also assimilation and accommodation which enable a pursuer to learn from either success or failure experiences. Thus, the pursuer can keep in step with an evader and catch it in a smaller number of capture steps.

4.3.3. Movement Types. The second experiment measures the average number of capture steps required by the proposed approach, Hysteretic Q-Learning, and RL-Fusing under two types of evader movement. One is clockwise movement, and the other is random clockwise and counterclockwise movement.

The average number of capture steps required by the three approaches for different movement types is shown in Table 2. Similar to the results obtained in the first experiment, the proposed scheme outperforms Hysteretic Q-Learning and RL-Fusing. Among the three approaches, Hysteretic Q-Learning requires the highest number of capture steps for

```

(1) while true do
(2)   next location  $\leftarrow$  generates randomly a location nearby the evader
(3)   if next location has an obstacle then
(4)     next location  $\leftarrow$  generates randomly a new location nearby the evader
(5)   end if
(6) end while

```

ALGORITHM 1: Random movement algorithm.

```

(1) direction  $\leftarrow$  generates randomly a direction
(2) distance  $\leftarrow$  generates randomly a distance shorter than the environment
(3) move  $\leftarrow$  0
(4) while true do
(5)   get the next location according to the direction
(6)   move add 1
(7)   if move equals to distance or next location has an obstacle then
(8)     evader turns right
(9)     changes direction to evader
(10)    move  $\leftarrow$  0
(11)  end if
(12) end while

```

ALGORITHM 2: Clockwise movement algorithm.

```

(1) direction  $\leftarrow$  generates randomly a direction
(2) distance  $\leftarrow$  generates randomly a distance shorter than environment
(3) move  $\leftarrow$  0
(4) while true do
(7)   if move equals to distance or next location has obstacle then
(8)     evader turns left
(9)     change direction to evader
(10)    move  $\leftarrow$  0
(11)  end if
(12) end while

```

ALGORITHM 3: Counterclockwise movement algorithm.

```

(1) while true do
(2)   if evader's sensor area has pursuer then
(3)     next location  $\leftarrow$  generate a location with least pursuers detected
(4)   else
(5)     next location  $\leftarrow$  randomly generate a location nearby the evader
(6)   end if
(7) end while

```

ALGORITHM 4: Smart movement algorithm.

the same reason that it does not support agent cooperation. According to the RL-Fusing algorithm presented in Algorithm 5, when a state is not the coordinated state, the action of an agent chooses a random or predefined policy and a single pursuer thus has no learning ability. Therefore, the

possibility of catching an evader is low, and RL-Fusing also takes more capture steps. In contrast, when the pursuer of the proposed scheme faces an evader, the pursuer observes the evader continuously. If the evader changes its moving type, the pursuer uses accommodation to change its catching

```

Require: A set of strategies, an initial state  $s_0$ 
(1)  $s \leftarrow s_0$ 
(2) while true do
(3)   if  $s$  is coordinated state then
(4)      $\rho \leftarrow \text{RandomReal}(0, 1)$ 
(5)     if  $\rho < \varepsilon$  then
(6)       select a strategy randomly //the same for all agent
(7)     else
(8)       rank strategies
(9)       communicate ranks
(10)      receive ranks from other agents
(11)      average ranks
(12)      select corresponding strategy
(13)    end if
(14)    execute selected strategy
(15)    receive reward
(16)    transit to a new state  $s'$ 
(17)    update Q-value
(18)  else
(19)    act with a predefined or random policy
(20)  end if
(21) end while

```

ALGORITHM 5: RL-Fusing algorithm.

TABLE 1: Average number of capture steps in different environments.

	Grid size	Without obstacles (steps)	With obstacles (steps)
Hysteretic Q-Learning	12 * 12	68	73
RL-Fusing	12 * 12	61	65
The proposed approach	12 * 12	24	27

TABLE 2: Average number of capture steps for different movement types.

	Grid size	Clockwise movement (steps)	Random clockwise and counterclockwise movement (steps)
Hysteretic Q-Learning	12 * 12	56	69
RL-Fusing	12 * 12	48	56
The proposed approach	12 * 12	26	32

strategy. If the moving type remains fixed, the pursuer keeps the same strategy by assimilation. As seen in Table 2, all three methods require more steps to capture evaders with random clockwise and counterclockwise movement because pursuers need extra time to detect the moving directions of the evaders.

TABLE 3: Average number of capture steps for evaders with smart movement in different grid sizes.

	Grid size: 12 * 12 (steps)	Grid size: 24 * 24 (steps)
Hysteretic Q-Learning	74	212
RL-Fusing	63	300
The proposed approach	63	119

TABLE 4: Average number of capture steps with/without assimilation and accommodation.

	Evader movement type Clockwise	Evader movement type Smart
With assimilation and accommodation	18.6	27.8
Without assimilation and accommodation	24.2	36.4

4.3.4. Smart Movement and Size of Grid. The third experiment involves evaders with smart movement. The evaders try actively to avoid pursuers when they sense any pursuer. The environment is of two grid sizes, 12 * 12 and 24 * 24.

Table 3 shows that the proposed scheme requires smaller capture steps in different grid sizes, when compared with Hysteretic Q-Learning and RL-Fusing. When the grid size is smaller, RL-Fusing outperforms Hysteretic Q-Learning. However, as the grid size becomes larger, RL-Fusing requires more capture steps than Hysteretic Q-Learning. This is

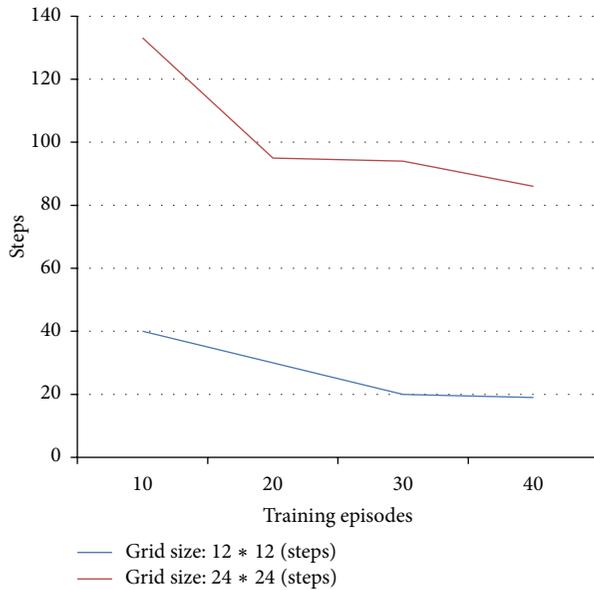


FIGURE 5: Average number of capture steps under different training episodes and grid sizes.

because RL-Fusing depends mainly on its cooperative strategies to catch evaders, but not all strategies can guide pursuers to move close to the evaders. For example, strategy 3 in RL-Fusing [37] is that once all the pursuers go at the distance of 3 from an evader. Because the sensing area of a pursuer is a $5 * 5$ square, the pursuers easily lose the evader in a larger environment. In contrast, the proposed approach assists the pursuers to move as close to the evader as possible by the avoidance of sensing overlapping. Thus, the approach has the highest possibility of catching the evader.

4.3.5. Smart Movement, Training Episodes, and Size of Grid. The fourth experiment investigates the relationship between training episodes and grid sizes when using the proposed approach.

Figure 5 shows that regardless of grid sizes ($12 * 12$ or $24 * 24$), the greater the number of training episodes is, the smaller the average number of capture steps is required. This is because more training enriches the case base, thus enabling pursuers to make better strategies and decisions.

4.3.6. Assimilation and Accommodation. The last experiment evaluates the proposed approach with/without assimilation and accommodation. There are two types of evader movement, clockwise and smart. The results are listed in Table 4. Whether the approach is with or without assimilation and accommodation, the average number of capture steps for smart movement is greater than that for clockwise movement. This is because smart movement can enable an evader to dynamically change its movement and position. In contrast, movement of an evader with clockwise movement can be easily predicted; hence, the pursuers can take fewer steps for capture. Furthermore, the average number of capture steps

required by the approach with assimilation and accommodation is smaller than that required by the approach without assimilation and accommodation for both movement types. The approach without assimilation and accommodation generates a fixed plan, which cannot be changed. In contrast, the approach with assimilation and accommodation enables the first pursuer to use the local-max strategy to find a better position, and the remaining pursuers can determine the cooperative positions according to the first pursuer's position. These pursuers can adjust their positions according to different catching situations and strategies. Accordingly, the approach with assimilation and accommodation requires fewer capture steps, when compared with the approach without assimilation and accommodation.

5. Conclusion

This work proposes an effective strategy learning approach to the pursuit-evasion game in a dynamic environment. This method can facilitate a pursuer in learning and implementing effective plans from either success or failure experiences under the condition of no knowledge of environment available. The evolved plans fare extremely well when compared with some of the best constructed strategies. This study also demonstrates the superior performance of the proposed approach for the pursuit-evasion game by implementing a multiagent system on REPAST.

Conflict of Interests

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

Acknowledgment

This work was supported by the Ministry of Science and Technology under Grant no. 102-2218-E-027-007.

References

- [1] K. P. Sycara, "Multiagent systems," *AI Magazine*, vol. 19, no. 2, pp. 79–92, 1998.
- [2] N. Vlassis, *A Concise Introduction to Multiagent Systems and Distributed Artificial Intelligence*, Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool, 2007.
- [3] A. Antoniadis, H. J. Kim, and S. Sastry, "Pursuit-evasion strategies for teams of multiple agents with incomplete information," in *Proceedings of the 42nd IEEE Conference on Decision and Control*, pp. 756–761, December 2003.
- [4] L. J. Guibas, J.-C. Latombe, S. M. Lavalley, D. Lin, and R. Motwani, "A visibility-based pursuit-evasion problem," *International Journal of Computational Geometry & Applications*, vol. 9, no. 4-5, pp. 471–493, 1999.
- [5] J. P. Hespanha, G. J. Pappas, and M. Prandini, "Greedy control for hybrid pursuit-evasion games," in *Proceedings of the European Control Conference*, pp. 2621–2626, 2001.
- [6] R. Vidal, O. Shakernia, H. J. Kim, D. H. Shim, and S. Sastry, "Probabilistic pursuit-evasion games: theory, implementation,

- and experimental evaluation,” *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 662–669, 2002.
- [7] R. E. Korf, “A simple solution to pursuit games,” in *Proceedings of the 2nd International Workshop on Distributed Artificial Intelligence*, pp. 183–194, 1992.
- [8] S. Nolfi and D. Floreano, “Coevolving predator and prey robots: do “arms races” arise in artificial evolution?” *Artificial Life*, vol. 4, no. 4, pp. 311–335, 1998.
- [9] N. J. Savill and P. Hogeweg, “Evolutionary stagnation due to pattern-pattern interactions in a coevolutionary predator-prey model,” *Artificial Life*, vol. 3, no. 2, pp. 81–100, 1997.
- [10] J. Y. Kuo, F.-C. Huang, S.-P. Ma, and Y.-Y. Fanjiang, “Applying hybrid learning approach to RoboCup’s strategy,” *Journal of Systems and Software*, vol. 86, no. 7, pp. 1933–1944, 2013.
- [11] S. I. Nishimura and T. Ikegami, “Emergence of collective strategies in a prey-predator game model,” *Artificial Life*, vol. 3, no. 4, pp. 243–260, 1998.
- [12] K.-C. Jim and C. L. Giles, “Talking helps: evolving communicating agents for the predator-prey pursuit problem,” *Artificial Life*, vol. 6, no. 3, pp. 237–254, 2000.
- [13] T. Haynes and S. Sen, “Evolving behavioral strategies in predators and prey,” in *Adaptation and Learning in Multi-Agent Systems*, vol. 1042 of *Lecture Notes in Computer Science*, pp. 113–126, Springer, Berlin, Germany, 1995.
- [14] D. Hládek, J. Vaščák, and P. Sinčák, “Hierarchical fuzzy inference system for robotic pursuit evasion task,” in *Proceedings of the 6th International Symposium on Applied Machine Intelligence and Informatics (SAMII ’08)*, pp. 273–277, Herl’any, Slovakia, January 2008.
- [15] R. G. Smith, “The contract net protocol: high-level communication and control in a distributed problem solver,” *IEEE Transactions on Computers*, vol. 19, no. 12, pp. 1104–1113, 1980.
- [16] P.-C. Zhou, B.-R. Hong, Y.-H. Wang, and T. Zhou, “Multi-agent cooperative pursuit based on extended contract net protocol,” in *Proceedings of the International Conference on Machine Learning and Cybernetics*, vol. 1, pp. 169–173, August 2004.
- [17] A. Rao and M. Georgeff, “BDI agents: from theory to practice,” in *Proceedings of the International Conference on Multi-Agent Systems*, pp. 312–319, 1995.
- [18] J. Piaget, *The Equilibration of Cognitive Structures: The Central Problem of Intellectual Development*, University of Chicago Press, 1985.
- [19] J. Y. Kuo, S. J. Lee, C. L. Wu, N. L. Hsueh, and J. Lee, “Evolutionary agents for intelligent transport systems,” *International Journal of Fuzzy Systems*, vol. 7, no. 2, pp. 85–93, 2005.
- [20] J. Y. Kuo and F. C. Huang, “A hybrid approach for multi-agent learning systems,” *Intelligent Automation and Soft Computing*, vol. 17, no. 3, pp. 385–399, 2011.
- [21] S. Gebhardt, P. Grant, R. von Georgi, and M. T. Huber, “Aspects of Piaget’s cognitive developmental psychology and neurobiology of psychotic disorders—an integrative model,” *Medical Hypotheses*, vol. 71, no. 3, pp. 426–433, 2008.
- [22] S. Takamuku and R. C. Arkin, “Multi-method learning and assimilation,” *Robotics and Autonomous Systems*, vol. 55, no. 8, pp. 618–627, 2007.
- [23] T. Parsons, “Pursuit-evasion in a graph,” in *Theory and Applications of Graphs*, vol. 642 of *Lecture Notes in Mathematics*, pp. 426–441, Springer, Heidelberg, Germany, 1978.
- [24] R. Breisch, “An intuitive approach to speleotopology,” *Southwestern Cavers*, vol. 6, no. 5, pp. 72–78, 1967.
- [25] F. Ferrari, “A new parameterized potential family for path planning algorithms,” *International Journal on Artificial Intelligence Tools*, vol. 18, no. 6, pp. 949–957, 2009.
- [26] K. Walsh and B. Banerjee, “Fast A* with iterative resolution for navigation,” *International Journal on Artificial Intelligence Tools*, vol. 19, no. 1, pp. 101–119, 2010.
- [27] L.-P. Wong, M. Y. H. Low, and C. S. Chong, “Bee colony optimization with local search for traveling salesman problem,” *International Journal on Artificial Intelligence Tools*, vol. 19, no. 3, pp. 305–334, 2010.
- [28] E. Raboin, D. S. Nau, U. Kuter, S. K. Gupta, and P. Svec, “Strategy generation in multi-agent imperfect-information pursuit games,” in *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, pp. 947–954, 2010.
- [29] B. P. Gerkey, S. Thrun, and G. Gordon, “Visibility-based pursuit-evasion with limited field of View,” *International Journal of Robotics Research*, vol. 25, no. 4, pp. 299–315, 2006.
- [30] N. Basilico, N. Gatti, and F. Amigoni, “Leader-follower strategies for robotic patrolling in environments with arbitrary topologies,” in *Proceedings of the 8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS ’09)*, pp. 57–64, May 2009.
- [31] M. Yan, “Multi-robot searching using game-theory based approach,” *International Journal of Advanced Robotic Systems*, vol. 5, no. 4, pp. 341–350, 2008.
- [32] K.-D. Althoff, “Case-based reasoning,” in *Handbook on Software Engineering and Knowledge Engineering*, pp. 549–587, 2001.
- [33] E. K. Burke, B. L. MacCarthy, S. Petrovic, and R. Qu, “Multiple-retrieval case-based reasoning for course timetabling problems,” *Journal of the Operational Research Society*, vol. 57, no. 2, pp. 148–162, 2006.
- [34] J. Long, D. G. Schwartz, S. Stoecklin, and M. K. Patel, “Application of loop reduction to learning program behaviors for anomaly detection,” in *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC ’05)*, vol. 1, pp. 691–696, April 2005.
- [35] The Repast Suite, The Repast Suite, 2014, <http://repast.sourceforge.net/>.
- [36] L. Maignon, G. J. Laurent, and N. L. Fort-Piat, “Hysteretic Q-Learning: an algorithm for decentralized reinforcement learning in cooperative multi-agent teams,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS ’07)*, pp. 64–69, San Diego, Calif, USA, October-November 2007.
- [37] I. Partalas, I. Feneris, and I. Vlahavas, “A hybrid multiagent reinforcement learning approach using strategies and fusion,” *International Journal on Artificial Intelligence Tools*, vol. 17, no. 5, pp. 945–962, 2008.

Research Article

Stereo Matching Based on Immune Neural Network in Abdomen Reconstruction

Huan Liu,^{1,2} Kuangrong Hao,³ Yongsheng Ding,³ and Chunjuan Ouyang^{1,2}

¹College of Electronic and Information Engineering, Jinggangshan University, Ji'an, Jiangxi 343009, China

²Laboratory of Watershed Ecology and Geographical Environment Monitoring, NASG, Ji'an, Jiangxi 343009, China

³College of Information Sciences and Technology, Donghua University, Shanghai 201620, China

Correspondence should be addressed to Huan Liu; liuhuan816618@163.com

Received 15 September 2014; Accepted 21 June 2015

Academic Editor: Marco Mussetta

Copyright © 2015 Huan 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.

Stereo feature matching is a technique that finds an optimal match in two images from the same entity in the three-dimensional world. The stereo correspondence problem is formulated as an optimization task where an energy function, which represents the constraints on the solution, is to be minimized. A novel intelligent biological network (Bio-Net), which involves the human B-T cells immune system into neural network, is proposed in this study in order to learn the robust relationship between the input feature points and the output matched points. A model from input-output data (left reference point-right target point) is established. In the experiments, the abdomen reconstructions for different-shape mannequins are then performed by means of the proposed method. The final results are compared and analyzed, which demonstrate that the proposed approach greatly outperforms the single neural network and the conventional matching algorithm in precise. Particularly, as far as time cost and efficiency, the proposed method exhibits its significant promising and potential for improvement. Hence, it is entirely considered as an effective and feasible alternative option for stereo matching.

1. Introduction

The morphology of human body, especially, abdomen shape, is a crucial factor to be considered in apparel customized services, body fitness, and so forth. It is widely accepted that obtaining the abdomen size accurately and effectively and building the 3D abdomen profile realistically are critical to provide a vital reference value for humane fashion design.

Image matching is a fundamental but critical step for its broad application, such as computer vision and noncontact measurement. The goal we pursuit is to realize an automatic, accurate, and efficient stereo matching algorithm. A previous feature matching approach based on a constraint condition can be found in [1]. Jeong and Moon [2] presented a further detailed constraint condition from the spatial coordinate, the orientation, and breadth of motion path. But the above methods require multitude computational time, and the larger error exists in the location for occluded point. Burie et al. [3] proposed exploring the features of the two stereo linear images sequentially from one end to the other. Performing

this scheme forward and backward and at different resolution levels made the matching in majority of candidate features without ambiguities. However, the false matches and miss-correspondences still exist in the final results. Recently, many researchers hope to explore faster and improved methods to solve the matching problem. The matching task is firstly formulated as an optimization problem where an energy function, defined to represent the constraints on the solution, is mapped into a two-dimensional Hopfield neural network for minimization in the literature [4]. Ruichek [5] put forward a feature matching based on the neural network. However, the disparity achieved is sparse due to the irregular distribution of features. It is necessary for additional surface fitting for a dense disparity. The method proposed by [6] is based on the using of ZNCC as matching cost, integrated within a neural network model. The results are satisfactory, but they are not suitable for real time applications on account of the high running time for standard image sets. Literature [7] executed fingerprint matching using a multidimensional artificial neural network. This method has some disadvantages,

such as qualitative computation and huge data storage. Neural network algorithm, similar to other classical combinatorial optimization algorithms, always converges at a stable state nearest to its initial state. When the initial condition is inappropriate, it is very possible to tap into the extremum. And it also depends on the correct matches and relationships among features. Once any connection error occurs, it is extremely easy to cause the mismatching. Moreover, since the match is innately an NP issue, its computation grows exponentially with the increase of the points' number. Consequently, the trade-off between execution time and quality of the matching is a difficult task and must be handled with care.

In this study, a creative intelligent biological matching algorithm based on the combination of B-T immune system and neural network is proposed. The details can be described as follows. From a viewpoint that the mathematical model is not clear in actual situation and difficult to complete the image feature matching, it combines the advantages of two-way adjustment flexibility of immune system and sound organizational configuration of neural network. In addition, in view of greater practicability, the extension of matching primitives from just pixel intensity to the combination of the intensity, entropy, and main orientation of gradient vector of pixel as well as the involvement of geometry constraint in T-cell layer is introduced for the sake of excluding the mismatching point. It not only improves the precision but also enhances the network convergence speed.

The outline of the rest of this paper is organized as follows: the basic principles concerning stereo matching, immune system, and neural network are introduced in Section 2. The details of the improving strategies of Bio-Net and DIE structure (main orientation of gradient, intensity, and entropy of pixel associate to a window) are expounded in Section 3. Qualitative experiments are performed and the results are analyzed and discussed in Section 4. Finally, conclusions are drawn in Section 5.

2. Overview

2.1. Stereo Matching Algorithm. Recently, literatures [8, 9] on feature matching proceeded from the idea of mimicking the principle of human visual system. Image matching is defined in a pair of stereo images for finding the correspondent pairs in order to estimate the depth information. The matching procedure can be described such that, for each pixel in right image, its matching point in the left image must be found or vice versa. In this paper, the pinhole for the cameras is adopted. It means that the projection of the 3D scene onto 2D images is described by a full perspective projection, and then the spatial information can be calculated by triangulation method [10].

Pursuing matching algorithms with high precision, high speed, and strong robustness is the goal to researchers. Generally speaking, the current matching methods are classified into two categories: area-based and feature-based. On the one hand, area-based algorithm takes gray intensity or color value into account to match the corresponding block which consists of the middle pixel and its surrounding neighbors. It can achieve high accuracy together with abundant details.

However, the proper size of block is difficult to select, and the matching cannot achieve good effects under the circumstance of shading and texture deficiency as well. On the other hand, feature matching mainly conducts by extracting the features like corners, lines, and edges. It does not directly depend on the intensities, so it is insensitive to noise and vulnerable to the light variation. Yet, it is always difficult to ensure the robustness when the data is incomplete, is not accurate enough, is redundant, or is even distortion. The above two kinds of methods all require a lot of computation, as well as a large amount of memory, making them unsuitable for most fast real-time applications.

2.2. Immunity Algorithm

2.2.1. Immune System and Artificial Immune System (AIS). All living organisms have the capacity of presenting some types of defense against strange attack. The evolution of species resulting in the emergence of the vertebrates also leads to the evolution of the immune system of species. The vertebrate immune system is particularly interesting due to its several computational capabilities.

Immune systems are naturally existing mechanisms which are responsible for detecting and coping with intruders in living organisms [11]. The main purpose of the immune system is to recognize all cells (or molecules) within the body, categorize those cells as self or nonself [12], and protect the organism against disease-causing cells called pathogens and eliminate malfunctioning cells [13]. To protect the body from invading pathogens (threats), there is a complex adaptive system that has embedded in vertebrates' biological immune system. The task of defeating against foreign attack is accomplished by special detectors called lymphocytes. These lymphocytes are created in random manner. After that, they are trained to remember infections so that the organism is protected from any future intrusions (threats) as well as past ones.

In immune system, immunity has two types of response: cellular and humoral responses, which are shown in Figure 1, where antigens (Ag) are pathogens that invade the body. The macrophages (MA) are large important phagocyte cells in innate immunity on the early nonadaptive phases of host defense. B cells (B) are activated by both the antigen itself and T cells. Antibodies (Ab) are produced by B cells to neutralize antigens.

Cellular response is involved in T cells. Macrophages recognize the antigen and present information about the antigen to helper T cells (T_H). This procedure is called antigen presenting. T_H cells accept the stimulation and produce a series of reactions, and then the information about the antigen is transmitted to the killer T cells (T_K). Thereby, T_K cells are activated. Then the germs infected by viruses are eliminated by T_K cells. Humoral response is involved in antibody, which is a core of the immune response. When an antigen invades the body, T_H cells stimulate those B cells, and the effector B cells (B_E) produce antibody. If the antigen is eliminated, the immune response stops. Then the concentration of immune cell and antibody begins to be adjusted, which induces the inhibition of suppressor T cells

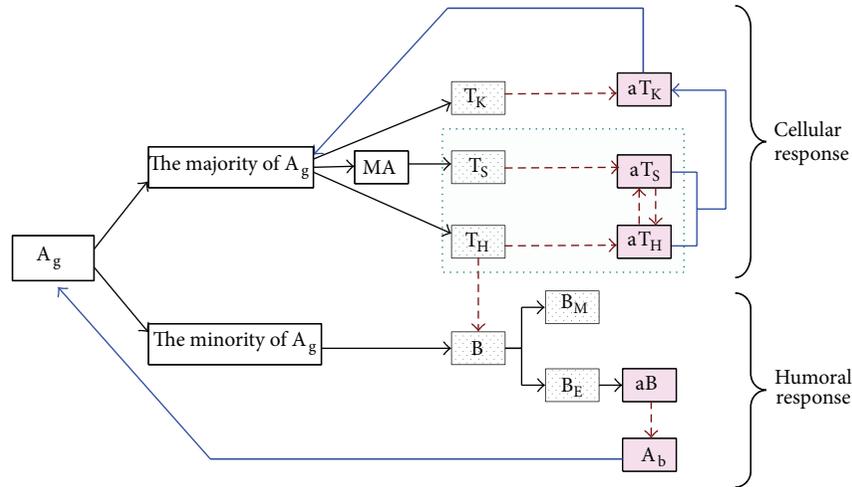


FIGURE 1: The procedure of immune response.

(T_S). Meanwhile, T_S cells secrete interleukin for suppressing the immune response. If the antibody is no longer produced, the immune response terminated.

Artificial immune system which emerged around the mid-1980s is a young field. Artificial immune system, a system of interconnected components, is inspired by biological/natural immune system. It simulates the identification between antigen and antibody and then combines the generation course of antibody and other particular immune mechanisms. The properties it contained are adaptability, diversity, learning, identification, memorability, and broad applicability. The field of AIS is becoming more popular and AIS-based work is spanning from theoretical modeling and simulation to wide variety of domains. For example, it has widely applied in machine-learning and pattern recognition, clustering classification [14], anomaly and intrusion detection [15, 16], autonomous navigation, optimization [17], and data mining [18].

2.2.2. Immune Network. Immune network theory [19] is first proposed by Jerne, which is defined as a network constituted of immune cells with mutual activation and coordination. The immune network algorithm can be categorized into PDP network, AINET, multivalued immune network, and dynamical identification immune network. Varela and Coutinho [20] proposed three characteristics: network structure, dynamics, and subdynamics. Hunt and Cooke [21] did the pioneer work of DNA recognition with immune network model and then came up with B-cell immune network algorithm. Tang et al. [22] designed a multivalued immune network on the basis of the relationship among T_H , T_S , and B cells and then successfully applied in character recognition. de Castro and von Zuben [23] introduced an immune network algorithm on morphological space, which is applied in the characteristic recognition. Although the existing model has effective applications, the theoretical research is difficult to recognize due to its complex network structure. Therefore it is necessary to explore continually.

2.3. Immune Neural Network. An artificial neural network (ANN) can be defined as information processing systems designed with inspiration taken from the nervous system. In most cases of the human brain, currently, most works on ANN place particular emphasis on the ability of solving practical problems. The important features of neural networks are the distributed information representation and the parallel processing [24]. Artificial neural networks have been recognized as powerful tools for learning and simulating systems in a great variety of fields, such as fault detection, automatic control, combinatorial optimization, information prediction, and other fields.

In the previous works, Gallo et al. [25] introduced neural networks in stereo matching. Lee et al. [26] presented Hopfield neural network in correspondent points. Sun et al. [27] proposed a scan lined-based asynchronous Hopfield neural network for eigenface matching. Zigh and Belbachir [28] applied successfully Hopfield neural network in extracting regions of buildings and matching them. Venkatesh et al. in the literature [29] explored the potential of self-organizing maps to solve the correspondent problem conceived as imitation of the stereo-perception ability of the human visual system. Kouskouridas et al. [30] proposed a new neural network-based solution to the 3D object pose estimation problem by establishing a novel input-output mapping with the learning process.

It is necessary to select proper neural network model, reasonable structure, and high-effective training algorithm in the course of solving practical problems. The aim of training, a process of adjusting the connection weights repeatedly, is to get a suitable network for concrete issue. In a variety of neural network models, multilayer feedforward neural network gets more attention and wider applications. However, confronting complicated problems, it is prone to plunge into local extreme and converge slowly.

Recently, there are some researchers contributing to the combination of immunity and neural network. For instance, a hybrid artificial intelligent approach based on the clonal

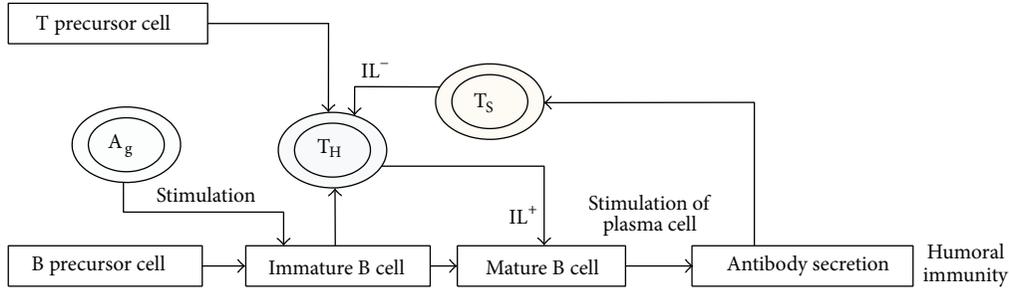


FIGURE 2: Activation procedure of B and T cell.

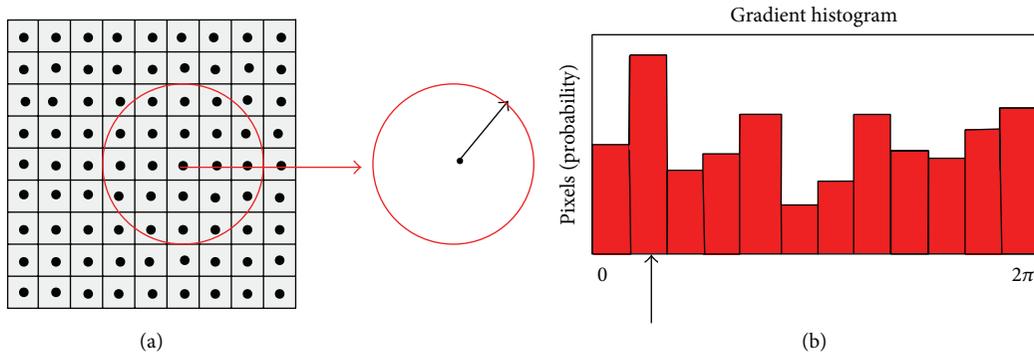


FIGURE 3: Main direction of circle.

selection principle of artificial immune system and neural networks is proposed to solve multiobjective programming problems in literature [31]. In addition, Mohammad and Zitar [32] put forward a novel idea that it utilizes the genetic optimization for artificial immune system and neural networks in spam detection application. In his paper, the immune system was an appealing system for spam detection because of the classification of self and nonself-messages. The classification needs to be categorized into the legitimate messages (the self) and spam (the nonself).

3. Feature Matching Based on B-T Immune System and Neural Network

3.1. The Mechanism of B-T Immune Regulation. B-T immune network takes effect through reciprocal activation and suppression between B cells and T cells. Figure 2 shows the activation procedure of B and T cells, where IL^+ and IL^- are interleukins secreted by T_H and T_S cells, respectively. As shown in Figure 2, when the antigens seep into body and are digested by surrounding cells, the news is delivered to T cells, that is, T_H cells and T_S cells. T_S cells can suppress the generation of T_H cells. Finally, they together activate B cells. After a period of time, the antibodies produced by B cells can remove the antigens. When the number of antigens becomes larger, T_H cells will increase, and T_S cells will decrease. Hence B cells will increase accordingly. With the decrease of antigen, T_S cells will increase, which inhibit the generation of T_H cells. As a result, B cells decrease. Ultimately, the immune feedback system tends to balance.

3.2. Feature Matching Based on B-T Immune Network. The left-right matching consistency of stereo matching is that a feature point in left image has a unique correspondent point in right image and vice versa. The biphasic regulation mechanism in B-T immune system inspires us to extract the similarity between B-T immunity mechanism and left-right consistency. A feature point from the left image is loaded on B cell, and then a search for correspondent point in right image can be considered as the activation from B cells to T cells. Conversely, a search for matching point from right image to left image can be considered as the suppression from T cells to B cells. Furthermore, geometry constraint property is involved into T-cell layer in order to ensure the uniqueness, so the mismatched points can be eliminated. Finally, the mechanism of involvement of B-T immune into BP neural network is adopted to improve the local search ability and enhance its global search capabilities for all the points.

3.2.1. A Combined Feature Description. The description is required after extracting features points. In order to improve the accuracy of identification, a higher distinctive description is demanded. It is to be noted that producing a description merely on the basis of the gray intensity is not enough to provide sufficient information for feature depiction. Consequently, taking the image structure into account, the unit entropy is proposed in this paper. Meanwhile, allowing for the engineering application background concerning the feature matching related to the human abdomen, the template is designed as circle window. As displayed in Figure 3. The main direction is determined by the distribution of the peak

gradient direction within the adjacent pixels around the key point

$$m(x, y) = \sqrt{(f(x+1, y) - f(x-1, y))^2 + (f(x, y+1) - f(x, y-1))^2}, \quad (1)$$

$$\theta(x, y) = \text{atan} \left(\frac{f(x, y+1) - f(x, y-1)}{f(x+1, y) - f(x-1, y)} \right),$$

where $m(x, y)$ and $\theta(x, y)$ are the gradient and direction of the point at coordinate (x, y) . The gradient histogram is calculated by dividing each unit at an angle (e.g., 30°) in a circle region (see Figure 3(b)), and then the peak of the gradient histogram is defined as the main direction. In the gradient histogram, the horizontal coordinate is the block of the direction angle and the vertical coordinate is the number that the gradient direction of a pixel belongs in its own angle block; that is, the probability of the gradient direction of each pixel belongs in the angle block on horizontal axis.

Once the main direction is set, the circle region is divided clockwise into six sectors at each 60° starting from the main direction, and each sector is divided into three subdivisions (see Figure 4). And then a total of 18 gate regions can be obtained (see Figure 4). The entropy can be calculated as

$$e_{ij} = - \sum_{m \in M} p_m \log_2^{p_m}, \quad (2)$$

where p_m denotes the probability in each unit gate of which intensity is equal to m .

Making use of CCA (canonical correlation analysis) principle, the intensity vector $\mathbf{F} = (f_1, f_2, \dots, f_N)$ and entropy vector $\mathbf{E} = (e_1, e_2, \dots, e_N)$ of each unit gate are composed as a new feature vector $\mathbf{g} = (g_1, g_2, \dots, g_N)$, where $N = 1, 2, \dots, 18$ (see Figure 5). It is great significance to explore feature description by means of synthesizing the above two individual properties.

3.2.2. Training of B-T Immune Network. First, input pattern vector \mathbf{g} which is a vector of pixel feature is inputted to the B cells' group (e.g., N is just the dimension of vector \mathbf{g} and also the number of B cells). Each dot of the input pattern is inputted to its corresponding B cell. The input pattern is weighted and then presented to all T_H cells (M is the number of T_H cells). This is the antigen presentation (Figure 6).

Their weight vector \mathbf{W} is defined as

$$\mathbf{W}_j = (w_{1j}, w_{2j}, \dots, w_{Nj}), \quad (3)$$

where $j = 1, 2, 3, \dots, M$ and the weight vector represents the stimulations of an input data to different T_H cell. For the purpose of avoiding activating a cell that has never been memorized, in the course of initialization, it is necessary to initialize the weights from B cells to T_H cells to be very small values. It can be achieved by

$$w_{ij} < \frac{L}{L-1+N}, \quad (4)$$

where L is a constant which is greater than 1 and N is the number of B cells, that is, the dimension of the input feature vector.

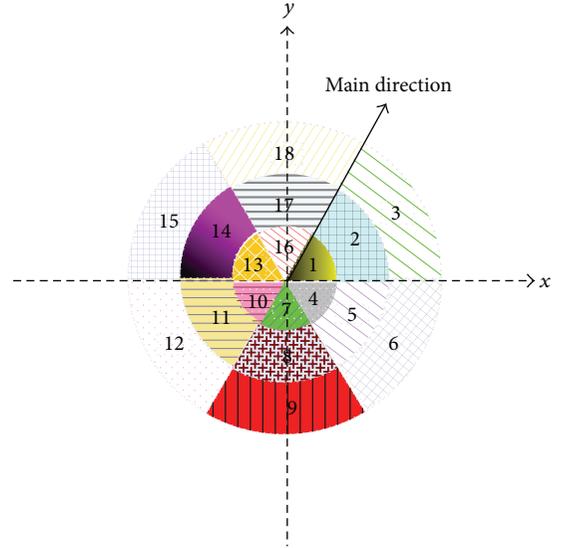


FIGURE 4: The distribution of 18 gate regions.

Then T_H cell takes the sum of the weighted input and only T_H cell that receives the strongest stimulus called antigen presentation secretes the interleukin (IL^+). Namely, T_H cells' group can be considered as a competition network. The interleukin (IL^+) is then weighted and set to B cells once again (see Figure 7). Thus, we have also M (the number of T_H cells) N -dimensional weighted vector \mathbf{T}_j

$$\mathbf{T}_{Sj} = (t_{1j}, t_{2j}, t_{ij}, \dots, t_{Nj}), \quad (5)$$

where suppression $t_{ij} = 0, 1, 2, \dots, M \times N - 1$, $i = 1, 2, \dots, N$; $j = 1, 2, \dots, M$, and M means dimension of vector \mathbf{T}_{Sj} .

We call the weight vector memory pattern. In B cells, the input feature vector \mathbf{g} and the memory pattern \mathbf{g}' are compared (see (6)), and the error of each pattern is computed and outputted to T_S cells' group. In T_S cells, antibodies, that is, the maximum errors b_{\max} , are compared to a predetermined parameter which is called tolerance ρ . If the maximum error is within the tolerance, the matching is viewed as a successful one and its coordinates are outputted. Otherwise it begins to enter the weights modification of neural network

$$b_i = |g_i - g'_i| \quad i = 1, 2, \dots, N. \quad (6)$$

From the explanation above, ρ is an important control parameter in the B-T immune network. Its value directly influences the evolution accuracy of immune network. Therefore, the choice of a proper ρ is essential. The correct feature mapping pairs obtained by TM in literature [33] are taken as samples. The feature vector \mathbf{g} of each feature point in the left image and \mathbf{g}' of each feature point in the right image are calculated, respectively. And then the distribution of b_i is necessary to be counted up and observed, which is computed by (6). The results show that b_i of about 87 percent of the matched pairs are in the range of 0.05 and 0.15. In view of this, ρ is taken as 0.10 in the paper. In addition, it is noted that the smaller ρ is, the longer the consuming time is.

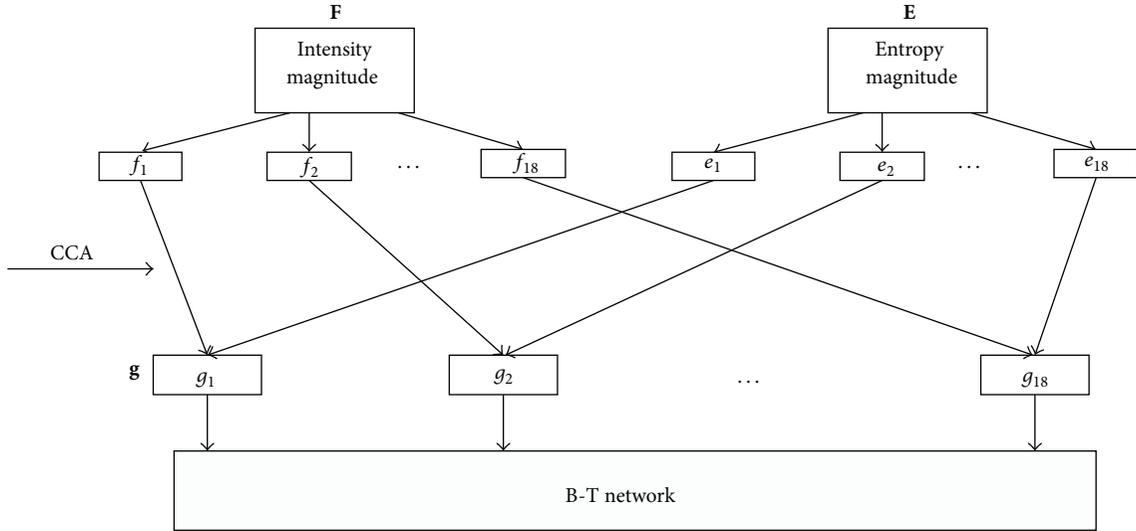
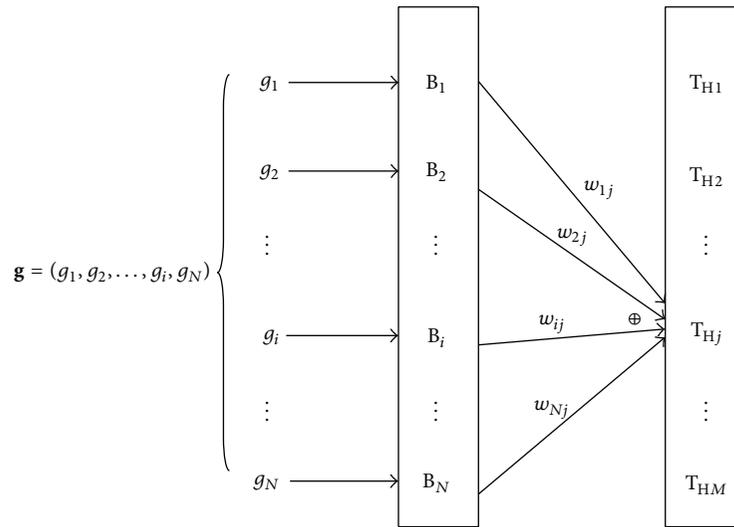


FIGURE 5: The combination of vectors F and E.

FIGURE 6: Weight connections from B cells to T_H cells.

3.2.3. Immune Network Evolvement. If maximum error b_{\max} is larger than the tolerant threshold ρ , the IL^- secreted from T_S cell is used to suppress T_H cell that has been activated. Thereby, it produces a competition once again and evolves the immune network.

The update of the k th B- T_H cell (T_{HK}) is thus

$$w_{ik}(\text{iter} + 1) = \frac{t_{ik}(\text{iter}) \cdot g_i}{\|T_{HK}\| \cdot \|\mathbf{g}\|}, \quad (7)$$

where $\|\mathbf{g}\| = \sqrt{\sum_{i=1}^N |g_i|^2}$, $\|T_{HK}\| = \sqrt{\sum_{i=1}^N |t_{ik}|^2}$, t_{ik} is the feedback weight from the k th T_H cell to the i th B cell, \mathbf{g} says an input vector, $\mathbf{g} = (g_1, g_2, g_i, \dots, g_N)$ ($i = 1, 2, \dots, N$), g_i denotes the i th input value from B cell, and iter is the iteration, respectively.

The update of the k th T_S -B cell (T_{SK}) is

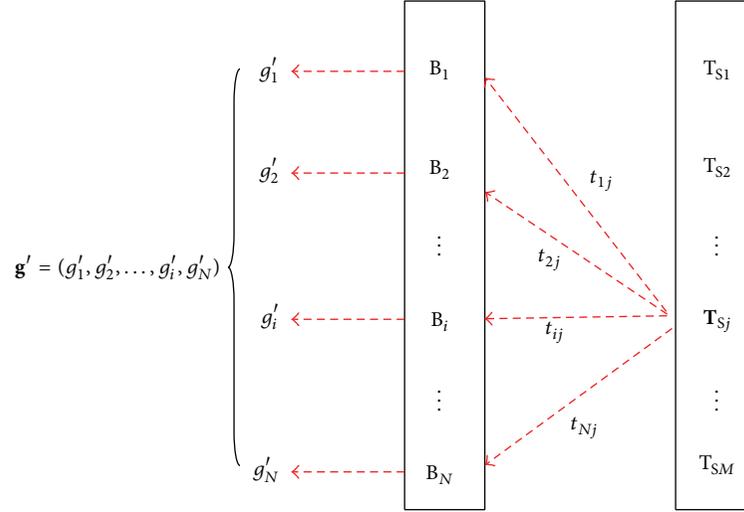
$$t_{ik}(\text{iter} + 1) = \frac{[t_{ik}(\text{iter}) + g_i]}{2}, \quad (8)$$

where iter is the iteration.

The procedure of B-T network is displayed in Figure 8.

3.2.4. Geometry Constraint. There may be several very similar larger values, but only one of them is correct matching point. The correct matching pairs have not only high similarity but also identical geometric distribution. For removing those error matches, the geometry consistency principle [33] is added to T cell, which benefits to obtain the unique maximum used to activate interleukin.

If a point L in the left image is matched with a point R in the right image, it is not possible for another point L' in


 FIGURE 7: Weight connections from T_S cells to B cells.

the left image, such that $X_{L'} < X_L$ (or $Y_{L'} < Y_L$), to be matched with another point R' in the right image with $X_{R'} > X_R$ (or $Y_{R'} > Y_R$). This constraint means that if a pair n_{LR} represents a correct match, then all pairs $n_{L'R'}$ such that $L' < L$ and $R' > R$ or such that $L' > L$ and $R' < R$ must correspond to invalid matches. In general, a feature detection algorithm enables us to extract a few hundreds to thousands features from an image, and therefore, there are a sufficient number of features on the same object that can prove their matching degrees reciprocally. In the paper, the distance and the slope of the connection between two matching points are employed to describe their geometric relationship. The images with same resolution from different perspectives were explored in the tests. Hence, it can be considered that the distance and the slope of the lines between the correct matching pairs are the same, which are, respectively, termed as letter D and K .

At the first place, the similarity S on the T cell is sorted by value in ascending order. At the second place, a certain percentage of number is selected according to the actual requirement. Finally, the values H (calculated in the light of (9)) are achieved in the selected sequence. Then sorting the new values H on T cell in descending order and taking the maximum as interleukin are used to activate B cells for reverse adjustment of t_{ij}

$$H = S \times D \times K. \quad (9)$$

3.3. The Neural Network Model Based on B-T Immune System (Bio-Net). The material basis of ANN comes from the model of neuron in biology. It is generally realized that an intact neuron is made up of several components, such as cell body, dendrite, axon, synapse, and neuron twin. A cell body is the principal part of a neuron. In existing ANN models, a neuron is regarded as a unit that sums up the all input signal and then outputs a signal after comparing with a threshold. Distinctive features of this kind of models denote that they have simple structures and good versatility. However,

the active and assistant functions of characteristics information are not considered in these features when dealing with a concrete problem. To be exact, there is no such interface in these existing models. Based on the consideration, the B-T immune system is designed in this paper for utilizing background information and prior knowledge of feature matching. This model is shown as Figure 9.

In the model shown above, the whole matching process is composed of two steps. Neural network plays a part in total optimization for the whole sample database. Moreover, each feature matching is realized by B-T immune network. Its architecture consists of a set of neurons, where the input layer is made up of two neurons and so is the output layer. Those neurons of input layer denote the coordinates of feature point in the left image $L(X_L, Y_L)$, and those of output layer indicate the coordinates of the matched point in the right image $R(X_R, Y_R)$. The middle hidden layer is designed as B-T immune network for the concrete feature matching. B-T immune network with bidirectional adjustment function has the similar mechanism with the consistency principle in feature matching. As for B-T immune network, N -dimensional vector \mathbf{g} derived from the combination of vectors \mathbf{F} and \mathbf{E} is taken as the input values of hidden layer (B cells). Then the weight values ($\sum_i w_{ij} \cdot g_i$) are received by T cells, and then N -dimensional weight vector ($\mathbf{T}(t_{ij})$) is generated. Meanwhile, the weight values are fed back to B cells. The corrections of the weights of neural network are shown in the following formulas:

$$u_{ij}(\text{iter} + 1) = u_{ij}(\text{iter}) + \delta_l \cdot y_j, \quad (10)$$

where $\delta_l = (p_l - o_l) \cdot o_l \cdot (1 - o_l)$, p_l and o_l are refer to the expected and calculated values of output, and $y_j = \sum_{i=1}^N w_{ij} g_i$, where y_j is the j th output value from T cell and g_i means the i th input value from B cell:

$$v_{ij}(\text{iter} + 1) = v_{ij}(\text{iter}) + \delta'_i \cdot x_j, \quad (11)$$

where $\delta'_i = y_i \cdot (1 - y_i) \sum_l \delta_l \cdot u_{li}$; x_j is the input value.

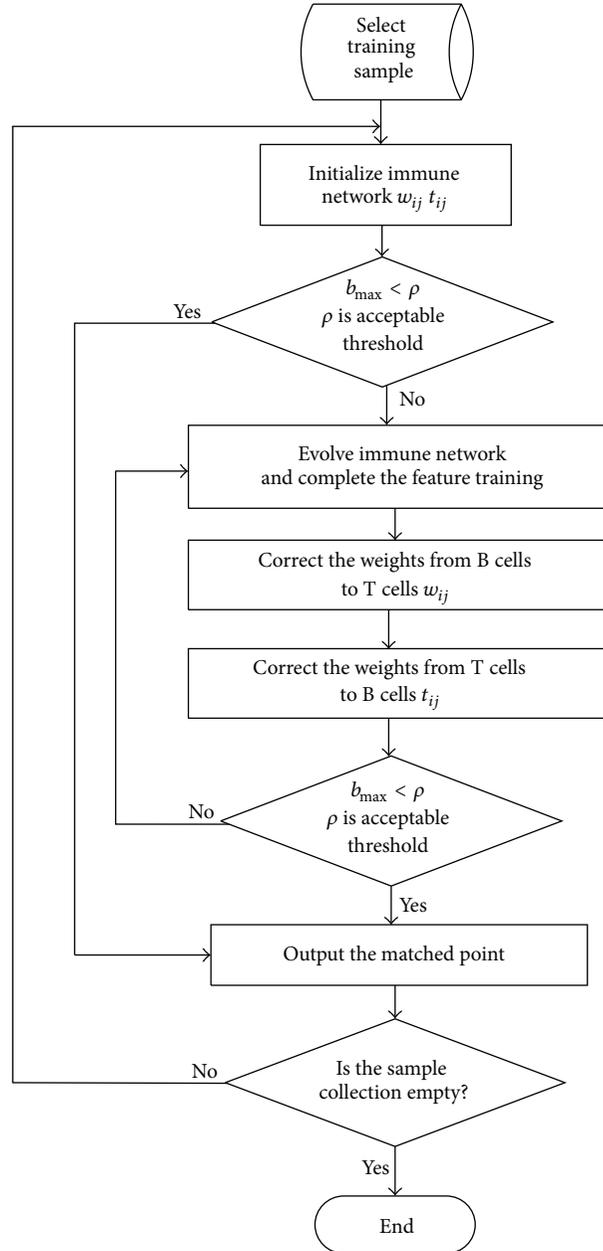


FIGURE 8: B-T network flow chart.

The intelligent stereo feature matching based on biological network algorithm was introduced above. Its specific steps are described as follows.

Step 1. Select training sample database of feature points for two stereo matching images.

Step 2. With a feature point $L(X_L, Y_L)$ in sample database as a center point, draw three different sizes of circulars, $r = 3$ pixels, 5 pixels, 7 pixels, and then determine the main direction. Finally, the circle area is divided clockwise into six sectors for every 60 degree starting with the main direction, and each sector is divided into three subdivisions, a total of 18 gate regions.

Step 3. Calculate the parameters intensity vector \mathbf{F} and entropy vector \mathbf{E} .

$$\mathbf{F} = (f_1, f_2, \dots, f_{18}), \quad (12)$$

$$\mathbf{E} = (e_1, e_2, \dots, e_{18}).$$

Step 4. Combine \mathbf{F} and \mathbf{E} as new $\mathbf{g} = (g_1, g_2, \dots, g_{18})$ with CCA.

Step 5. Complete intelligent matching computation by B-T immune network and obtain a matched point $R(X_R, Y_R)$.

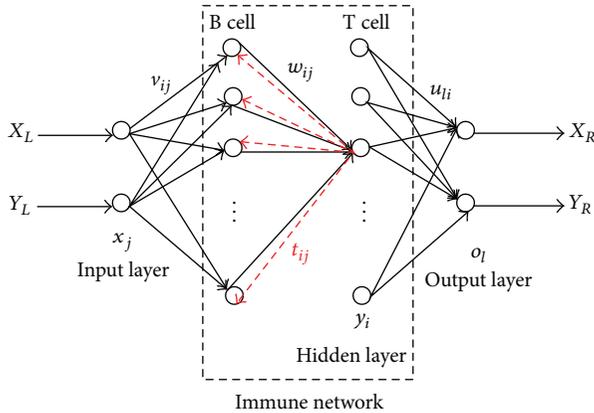


FIGURE 9: Immune ANN network model.

Step 6. Regulate weights of neural network by means of minimizing the error between expected value and calculated value achieved from Step 5.

Step 7. Return to Step 1 and get a next point until all the training feature points completed, and the algorithm ends up.

Step 8. Train all the feature points in sample database by neural network, and establish the model of reference feature points and matched correspondences.

Step 9. Retrieve the correct matching correspondences for all the testing feature points in left and right images by that corresponding model so as to obtain the three-dimensional cloud data.

4. Experiment and Analysis

4.1. Experimental Setup. In the section, experiments on the abdomen reconstruction and measurement for three different shape models are conducted by the proposed methods. The specific setup shown in Figure 10 consists of three main parts: a shape-flexible model installed airbag (see Figure 10(a)), an electric air pump (see Figure 10(b)), and a software platform (see Figure 10(c)). The electric air pump is applied to make shape variation by inflating the airbag on abdomen with different pressures. In this test, the airbag is inflated about 10 kpa and 20 kpa air, respectively. The two correspondent shapes in left view are shown in Figure 11. The concrete instructions are given as follows. The image shown in Figure 11(a) is on the initial condition with no pressure in abdomen airbag, which says model 1. Model 2 (shown in Figure 11(b)) is the condition with 10 kpa air and model 3 (shown in Figure 11(c)) says the condition with 20 kpa air. The three images with different pressures in right view are shown in Figure 12. With the increase of the pressure, the deformation of the abdomen shape gradually occurs.

Here given a comparison with three different matching methods, namely, traditional method (TM), BP neural network (ANN), and biological network, we proposed Im-ANN. TM, the previous study we have proposed in the literature

[33], mainly includes two steps: extracting features and matching. The correct matching pairs which have obtained in TM are taken not only as training samples for two methods (ANN and Im-ANN) but also as the standard values for error analysis in this paper. The BP neural network (ANN) consists of three layers, the architecture of which is similar to the Im-ANN. The only key difference is in the middle hidden layer without B-T immune network but rather some ordinary neurons.

4.2. Tests of Feature Matching. The following results associated with feature mapping are shown in Figure 13. Figure 13(a) displays the comparison of correct matching rate in three approaches, which denotes the ratio between the number of correct matching and the total matching pairs. Figure 13(b) denotes the contrast of runtime in three ways. As can be seen from the charts, the correct matching rate derived from TM and Im-ANN is most extremely similar and approximately at 90%, while the value from ANN is only at about 60%. The computational time taken in ANN and Im-ANN is very similar, and TM takes the longest time in three methods.

For the sake of general effect, the following error analyses are produced by the statistics in their own unmatched results. The contrast related to average error among the three mentioned methods is shown in Figure 14. Figure 14(a) reveals the error at coordinate X, and Figure 14(b) indicates the error at coordinate Y. As the figure shows, no matter whether X or Y coordinates, the errors calculated by ANN are all larger, up to more than 1.5 pixel, while the error derived from TM and Im-ANN is merely nearly to 0.5 pixel which indicates that Im-ANN enables us to achieve better matching effect. A further detailed error analyses are exhibited in Figure 15 including maximum, minimum, and average values. Figure 15(a) shows errors at coordinate X and Figure 15(b) shows errors at coordinate Y. It can be observed that both maximum and minimum errors from ANN are all greater than those from TM and Im-ANN. There are no distinct difference with TM and Im-ANN. The comparison about the rate is computed on the error greater than 3 pixels (see Figure 16). Figure 16(a) shows the errors at coordinate X and Figure 16(b) shows the errors at coordinate Y. It can be found that the results obtained from ANN are worst with two times greater than other two methods, and Im-ANN is slightly better than TM.

Overall, the proposed method (Im-ANN) proves its high correct matching rate and great superiority on runtime over TM and ANN methods, with exceedingly similar error to TM, whereas ANN is inferior in that without any consideration for both the innate principle of left-right associated consistency and the geometry constraint. There exists the same mechanism between regulation of B-T immune network and the correspondence problem, so it is believed to be reasonable in application of stereo matching. By means of combining the flexible regulation of B-T immune system and the rational organized structure, it is able to heighten the capability of dealing with concrete problem and lessening the computational workload in training network with the improvement of mapping accuracy and convergence velocity.

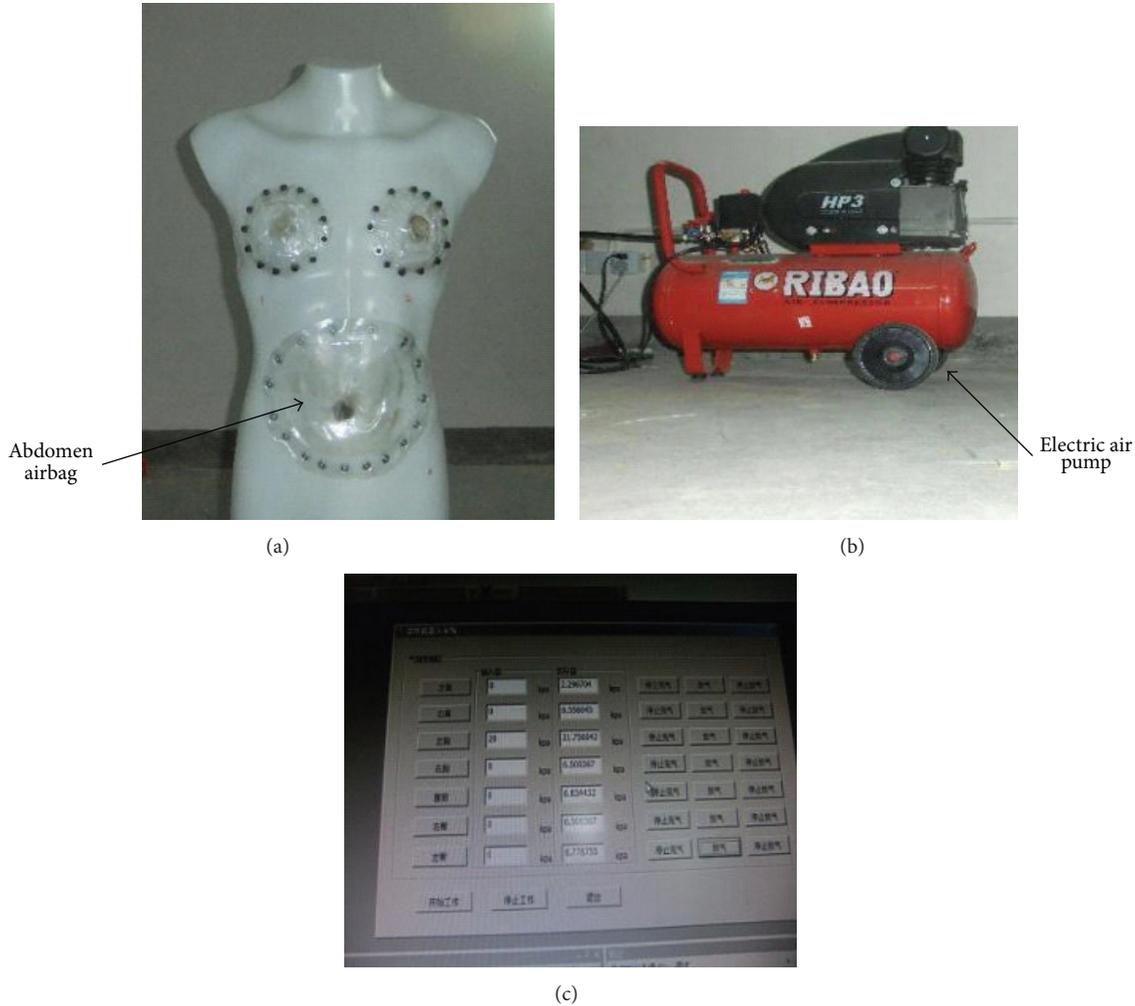


FIGURE 10: Experimental setup.

4.3. Analysis of Shape Size. In order to evaluate the overall accuracy of our proposed method, it is tested on a shape-flexible model. We complete the measurement tests for the abdomen circumference through three independent models, that is, on model 1 (the initial state), model 2 (with the pressure of 10 kpa), and model 3 (20 kpa), respectively. The three-dimensional cloud data of human abdomen from two different views are shown in Figure 17. (a) and (a1) are results of model 1; (b) and (b1) are results of model 2; (c) and (c1) are results of model 3.

Manual measurement is a traditional contact method. The 3D scanner is considered as the comparative highest accuracy of noncontact measurement. Comparisons of parameters are made among the above three schemes (Im-ANN, 3D scanner, manual measurement), and the values from manual measurement are taken as the standard values for comparisons in Table 1. CIR means the circumference, DEP means the abdomen depth, BRE means the abdomen breadth, $Error_M$ & Im is the error between manual and Im-ANN in three different models, $Error_M$ & 3D is the error between manual and 3D scanner in three different models.

Similarly, $Error_3D$ & Im is the same meaning parameter between 3D scanner and Im-ANN. In comparison with the manual measurement, although the differences exist, the differences from Im-ANN are merely slight, smaller than the differences from 3D scanner, and completely conform to the acceptable criteria in the light of the apparel design and engineering. Meanwhile, the precisions of belly parameters obtained from Im-ANN are highly similar to those from the 3D scanner.

5. Conclusion

In this paper, the stereo correspondence problem is formulated as an optimization task. For the purpose of solving this problem effectively, we proposed a novel intelligent matching algorithm based on the combination of immune system and neural network, as well as DIE data structure. The proposed intelligent network combined neural network with B-T immune system, which is comprised of three layers and is involved the well-known biphasic regulation B-T network into intermediate layer. The first step deals with computing

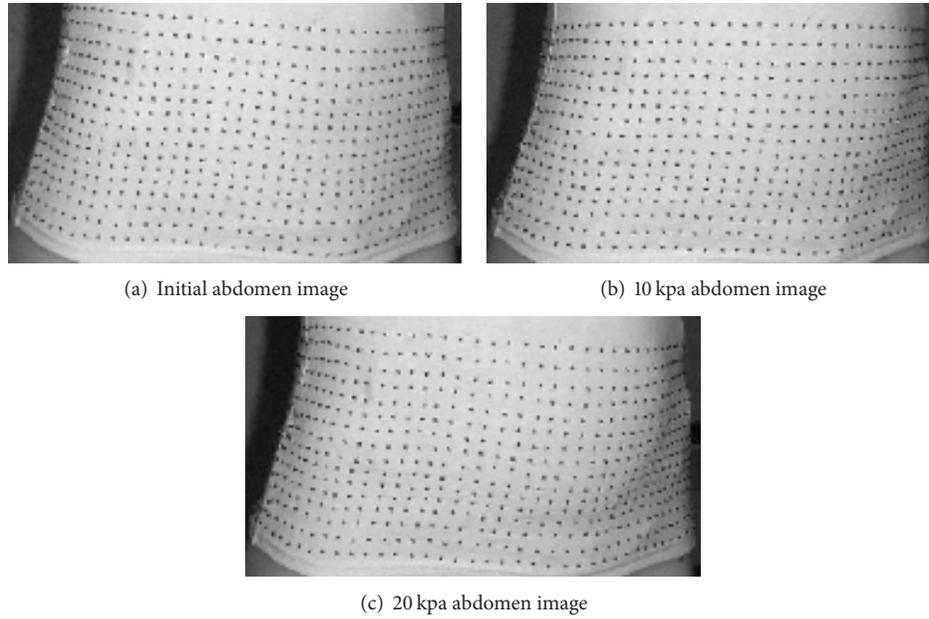


FIGURE 11: Abdomen images from left camera under different conditions.

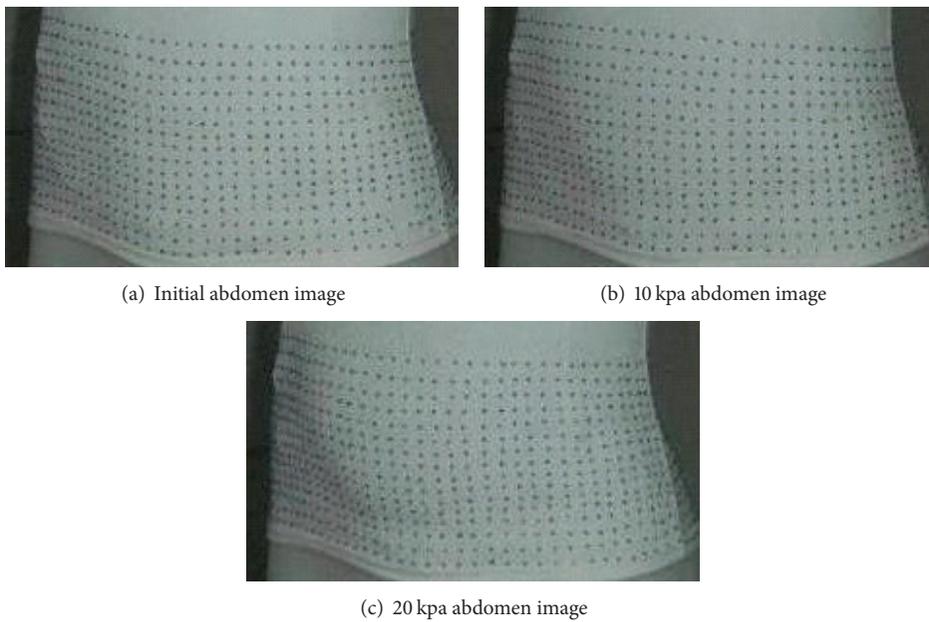


FIGURE 12: Abdomen images from right camera under different conditions.

TABLE 1: Error inspection of abdomen size between measured and calculated value (cm).

Model	Model 1			Model 2			Model 3		
	CIR	DEP	BRE	CIR	DEP	BRE	CIR	DEP	BRE
Manual	69.8	16.3	23.8	78.2	18.2	25.2	82.6	21.8	29.2
3D scanner	69.65	16.26	23.75	77.79	18.13	25.04	82.09	21.66	28.97
Im-ANN	69.62	16.27	23.74	77.81	18.11	25.06	82.13	21.69	28.94
<i>Error_M & Im</i>	0.18	<i>0.03</i>	0.06	0.39	<i>0.09</i>	0.14	0.47	<i>0.11</i>	0.26
<i>Error_M & 3D</i>	0.15	<i>0.04</i>	0.05	0.41	<i>0.07</i>	0.16	0.51	<i>0.14</i>	0.23
<i>Error_3D & Im</i>	0.03	<i>0.01</i>	0.01	0.02	<i>0.02</i>	0.02	0.04	<i>0.03</i>	0.03

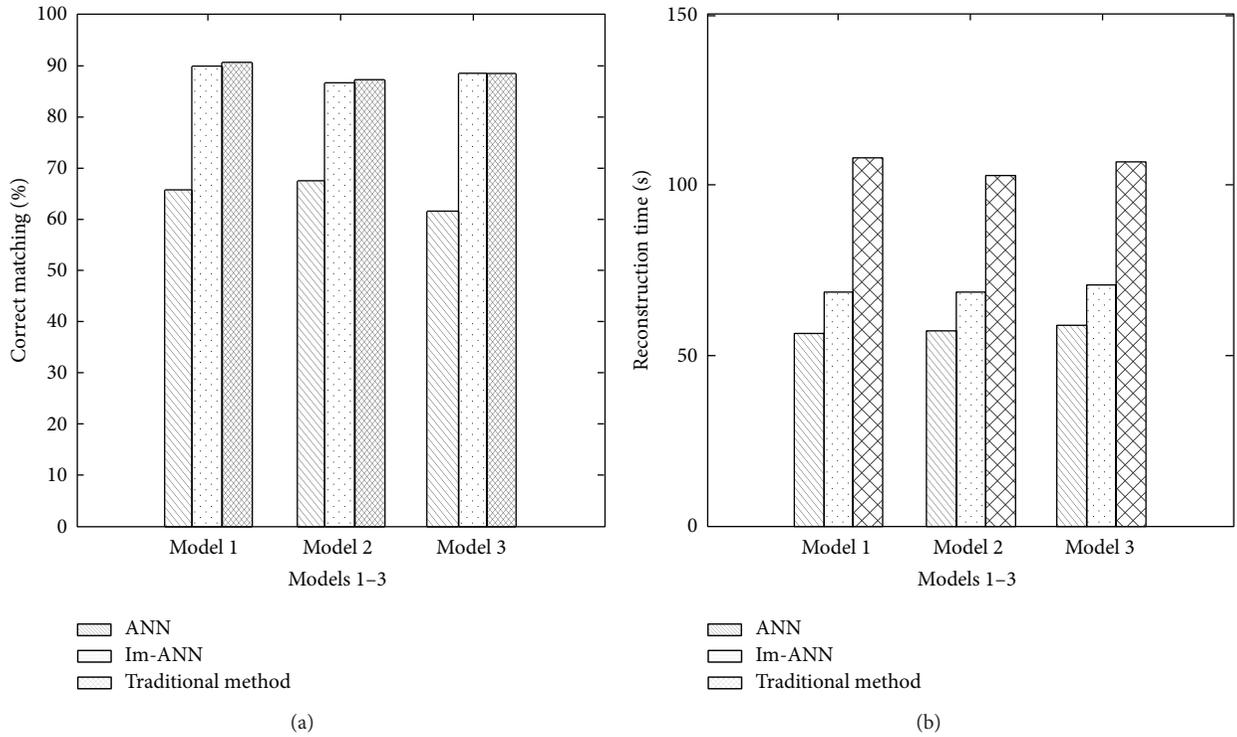


FIGURE 13: Comparison of correct matching ratio and runtime in three methods.

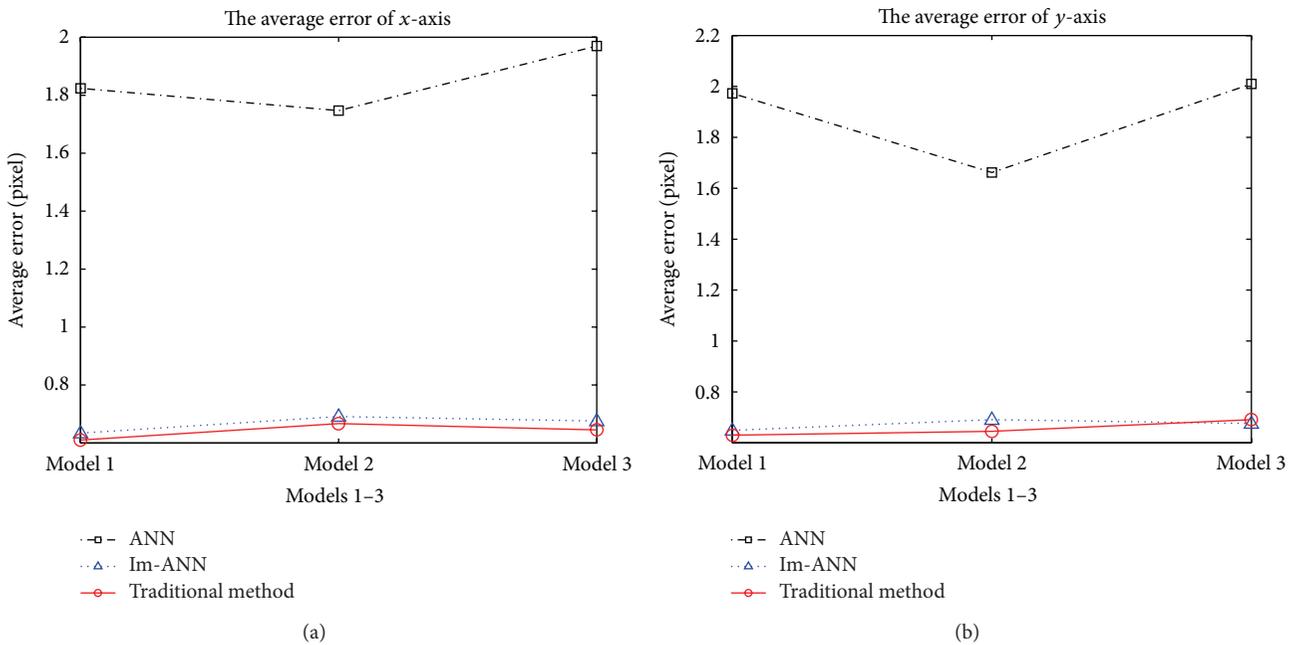


FIGURE 14: Analysis of average error in comparison in three methods.

the matched feature point using B-T immunity and DIE structure for each feature point in sample database. The second step conducts the regulation of weights connected neurons by means of neural network in order to minimize the propagation error in correspondence process and an accurate result can be achieved. Finally, the corresponding relation

between feature points in left view image and matched points in right view image is established for the general pixels matching and is used to complete the three-dimensional dense cloud data for abdomen reconstruction.

Three matching algorithms, traditional method (TM), BP neural network (ANN), and immune neural network

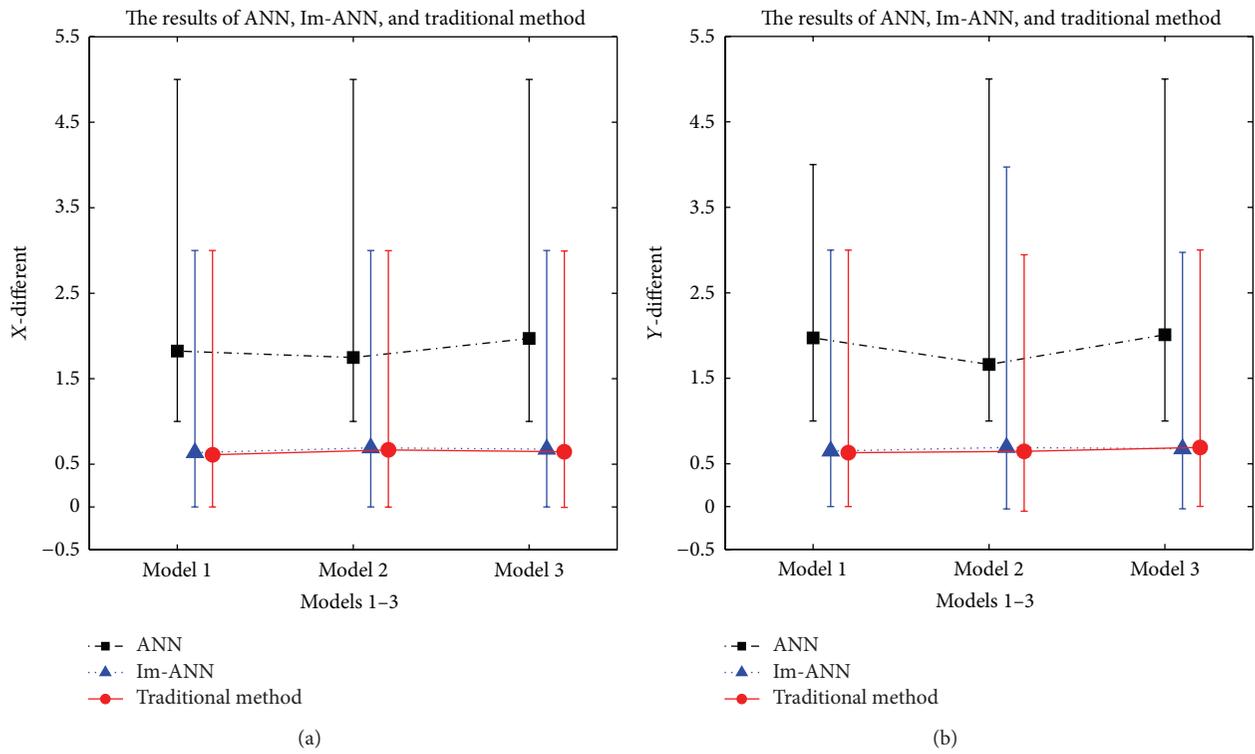


FIGURE 15: Detailed error analyses in comparison in three methods.

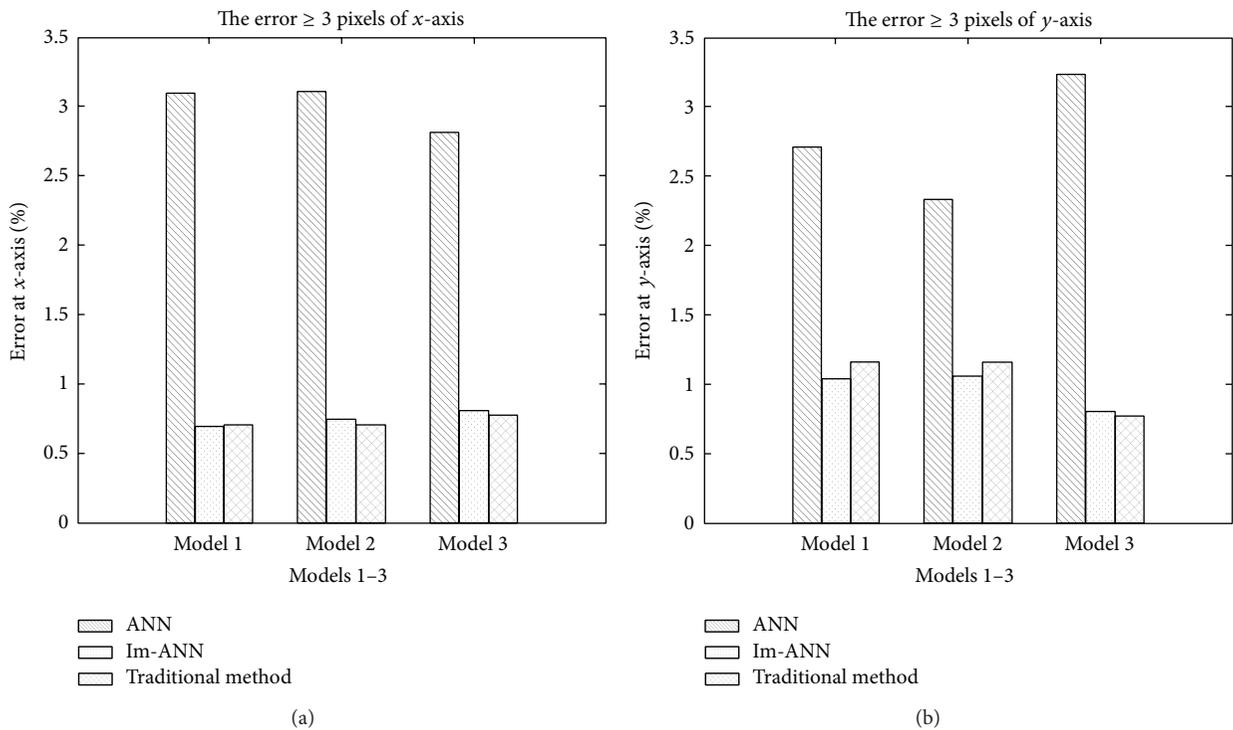


FIGURE 16: Analysis of ratio about error ≥ 3 pixels in comparison in three methods.

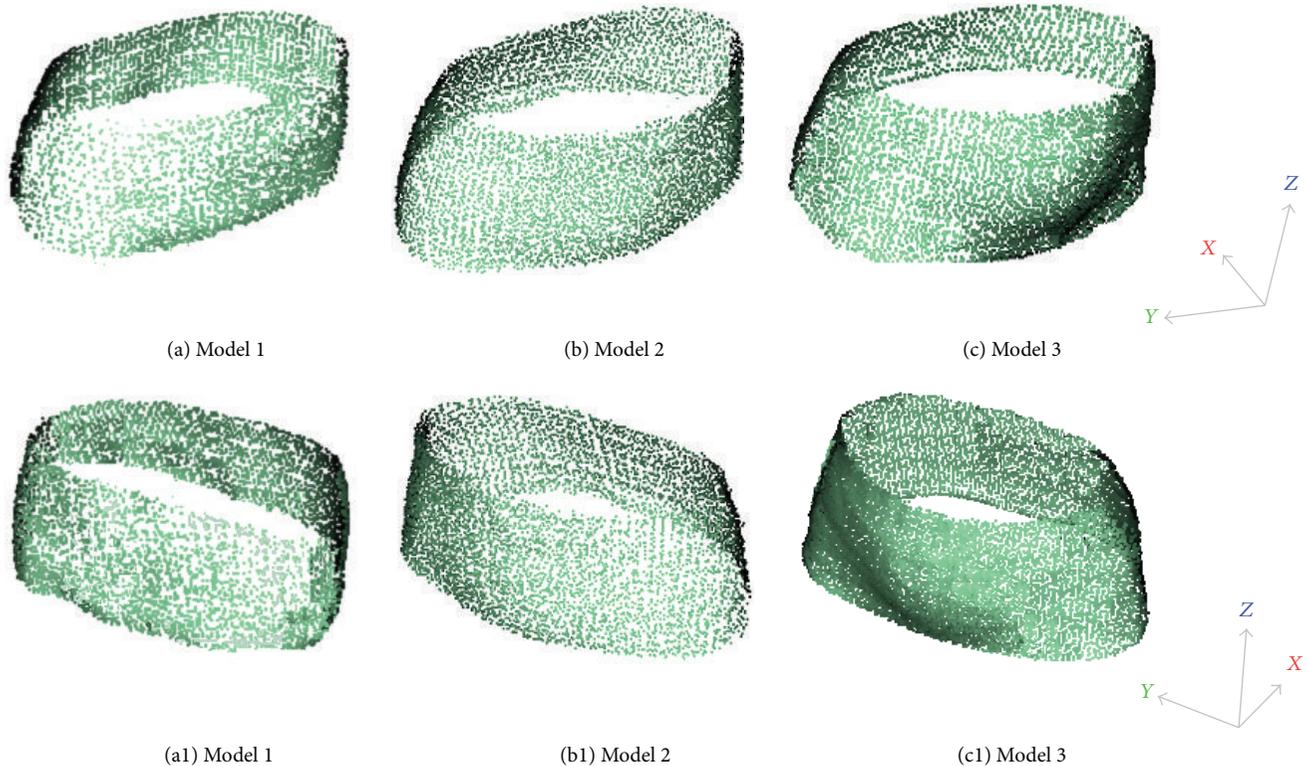


FIGURE 17: Three-dimensional cloud data of three different models from two different views.

(Im-ANN), are proposed to perform the task of correspondence for abdomen stereo images. Comparing the three algorithms, the experimental results in this paper show that the proposed algorithms (Im-ANN) can obtain the same accuracy as the traditional method (TM) but a less time consumption. The advantages of this method over other methods include massive parallelism and possibility of hardware implementation. It was concluded that the use of intelligent network in stereo correspondence is necessary, useful, and helpful to improve the performance of classical BP neural network algorithm. All its merits make it promising for application in a large scale anthropometric survey to collect body dimension for personalized apparel design and other related to human engineering and so forth.

Notwithstanding, there exist a number of deficiencies, such as massive sample data memory and computation, and some parameters rely on experience and so forth. Future work in this field could be in optimizing the trade-off between the speed of training time and the size of the database can be supported by the network and proposition of a common method for setting parameters about immune network, trying to apply to other types of neural networks for wider fields, such as information forecasts and machine control.

Conflict of Interests

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

Acknowledgments

This work was supported in part by the Key Project of the National Nature Science Foundation of China (no. 61134009), the National Nature Science Foundation of China (nos. 61473077, 61473078, and 61462046), cooperative research funds of the National Natural Science Funds Overseas and Hong Kong and Macao scholars (no. 61428302), Program for Changjiang Scholars and Innovation Research Team in University from the Ministry of Education (no. IRT1220), Specialized Research Fund for Shanghai Leading Talents, Project of the Shanghai Committee of Science and Technology (nos. 13JC1407500, 11JC1400200), Innovation Program of Shanghai Municipal Education Commission (no. 14ZZ067), the Fundamental Research Funds for the Central Universities (no. 2232012A3-04), the Key laboratory of watershed ecology and geographical environment monitoring, NASG, the Natural Science Foundation of Jiangxi Province (no. 20151BAB207026), the Natural Science Fund for Young Scholars of Jiangxi Province (no. 20151BAB217012), and the Science and Technology Research Projects of Jiangxi Province Education Department (no. GJJ14559).

References

- [1] V. Salari and I. K. Sethi, "Feature point correspondence in the presence of occlusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 87–91, 1990.
- [2] J. M. Jeong and Y. S. Moon, "A feature tracking algorithm using adaptive weight adjustment," *Journal of the Institute of*

- Electronics Engineers of Korea S*, vol. 36, no. 11, pp. 68–78, 1999.
- [3] J.-C. Burie, J.-L. Bruyelle, and J.-G. Postaire, "Detecting and localising obstacles in front of a moving vehicle using linear stereo vision," *Mathematical and Computer Modelling*, vol. 22, no. 4–7, pp. 235–246, 1995.
 - [4] J. J. Hopfield and D. W. Tank, "Neural computation of decisions in optimization problems," *Biological Cybernetics*, vol. 52, no. 3, pp. 141–152, 1985.
 - [5] Y. Ruichek, "Multilevel- and neural-network-based stereo-matching method for real-time obstacle detection using linear cameras," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 54–62, 2005.
 - [6] E. Binaghi, I. Gallo, G. Marino, and M. Raspanti, "Neural adaptive stereo matching," *Pattern Recognition Letters*, vol. 25, no. 15, pp. 1743–1758, 2004.
 - [7] R. Kumar and B. R. D. Vikram, "Fingerprint matching using multi-dimensional ANN," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 2, pp. 222–228, 2010.
 - [8] S. Gutiérrez and J. L. Marroquín, "Robust approach for disparity estimation in stereo vision," *Image and Vision Computing*, vol. 22, no. 3, pp. 183–195, 2004.
 - [9] Q. Luo, J. Zhou, S. Yu, and D. Xiao, "Stereo matching and occlusion detection with integrity and illusion sensitivity," *Pattern Recognition Letters*, vol. 24, no. 9–10, pp. 1143–1149, 2003.
 - [10] P. Zhao and N.-H. Wang, "Precise perimeter measurement for 3D object with a binocular stereo vision measurement system," *Optik*, vol. 121, no. 10, pp. 953–957, 2010.
 - [11] F. Seredynski and P. Bouvry, "Anomaly detection in TCP/IP networks using immune systems paradigm," *Computer Communications*, vol. 30, no. 4, pp. 740–749, 2007.
 - [12] A. Kalinli and N. Karaboga, "Artificial immune algorithm for IIR filter design," *Engineering Applications of Artificial Intelligence*, vol. 18, no. 8, pp. 919–929, 2005.
 - [13] P. Musilek, A. Lau, M. Reformat, and L. Wyard-Scott, "Immune programming," *Information Sciences*, vol. 176, no. 8, pp. 972–1002, 2006.
 - [14] Y. Zhong, L. Zhang, J. Gong, and P. Li, "A supervised artificial immune classifier for remote-sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 12, pp. 3957–3966, 2007.
 - [15] F. Esponda, S. Forrest, and P. Helman, "A formal framework for positive and negative detection schemes," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 1, pp. 357–373, 2004.
 - [16] A. O. Tarakanov, "Immunocomputing for intelligent intrusion detection," *IEEE Computational Intelligence Magazine*, vol. 3, no. 2, pp. 22–30, 2008.
 - [17] K. M. Woldemariam and G. G. Yen, "Vaccine-enhanced artificial immune system for multimodal function optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 40, no. 1, pp. 218–228, 2010.
 - [18] A. A. Freitas and J. Timmis, "Revisiting the foundations of artificial immune systems for data mining," *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 4, pp. 521–540, 2007.
 - [19] N. K. Jerne, "Towards a network theory of the immune system," *Annales de l'Institut Pasteur/Immunologie C*, vol. 125, no. 1–2, pp. 373–389, 1974.
 - [20] F. J. Varela and A. Coutinho, "Second generation immune networks," *Immunology Today*, vol. 12, no. 5, pp. 159–166, 1991.
 - [21] J. E. Hunt and D. E. Cooke, "Learning using an artificial immune system," *Journal of Network and Computer Applications*, vol. 19, no. 2, pp. 189–212, 1996.
 - [22] Z. Tang, T. Yamaguchi, K. Tashima, O. Ishizuka, and K. Tanno, "Multiple-valued immune network model and its simulations," in *Proceedings of the 27th International Symposium on Multiple-Valued Logic*, pp. 233–238, May 1997.
 - [23] L. N. de Castro and F. J. von Zuben, "Learning and optimization using the clonal selection principle," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 3, pp. 239–251, 2002.
 - [24] L. N. de Castro, "Fundamentals of natural computing: an overview," *Physics of Life Reviews*, vol. 4, no. 1, pp. 1–36, 2007.
 - [25] I. Gallo, E. Binaghi, and M. Raspanti, "Neural disparity computation for dense two-frame stereo correspondence," *Pattern Recognition Letters*, vol. 29, no. 5, pp. 673–687, 2008.
 - [26] J. J. Lee, J. C. Shim, and Y. H. Ha, "Stereo correspondence using the Hopfield neural network of a new energy function," *Pattern Recognition*, vol. 27, no. 11, pp. 1513–1522, 1994.
 - [27] T.-H. Sun, M. Chen, S. Lo, and F.-C. Tien, "Face recognition using 2D and disparity eigenface," *Expert Systems with Applications*, vol. 33, no. 2, pp. 265–273, 2007.
 - [28] E. Zigh and M. F. Belbachir, "Soft computing strategy for stereo matching of multi spectral urban very high resolution IKONOS images," *Applied Soft Computing Journal*, vol. 12, no. 8, pp. 2156–2167, 2012.
 - [29] Y. V. Venkatesh, S. K. Raja, and A. J. Kumar, "On the application of a modified self-organizing neural network to estimate stereo disparity," *IEEE Transactions on Image Processing*, vol. 16, no. 11, pp. 2822–2829, 2007.
 - [30] R. Kouskouridas, A. Gasteratos, and C. Emmanouilidis, "Efficient representation and feature extraction for neural network-based 3D object pose estimation," *Neurocomputing*, vol. 120, no. 23, pp. 90–100, 2013.
 - [31] W. F. Abd El-Wahed, E. M. Zaki, and A. M. El-Refaey, "Artificial immune system based neural networks for solving multi-objective programming problems," *Egyptian Informatics Journal*, vol. 11, no. 2, pp. 59–65, 2010.
 - [32] A. H. Mohammad and R. A. Zitar, "Application of genetic optimized artificial immune system and neural networks in spam detection," *Applied Soft Computing*, vol. 11, no. 4, pp. 3827–3845, 2011.
 - [33] H. Liu, K. R. Hao, and Y. S. Ding, "New anti-blur and illumination-robust combined invariant for stereo vision in human belly reconstruction," *Imaging Science Journal*, vol. 62, no. 5, pp. 251–264, 2014.

Research Article

Landslide Occurrence Prediction Using Trainable Cascade Forward Network and Multilayer Perceptron

Mohammad Subhi Al-batah,¹ Mutasem Sh. Alkhasawneh,² Lea Tien Tay,²
Umi Kalthum Ngah,² Habibah Hj Lateh,³ and Nor Ashidi Mat Isa²

¹Department of Software Engineering, Faculty of Science and Information Technology, Jadara University, Irbid 2001, Jordan

²School of Electrical & Electronic Engineering, Universiti Sains Malaysia, Engineering Campus, 14300 Nibong Tebal, Penang, Malaysia

³School of Distance Education, Universiti Sains Malaysia, 11600 Penang, Malaysia

Correspondence should be addressed to Mutasem Sh. Alkhasawneh; m_sh_kal@yahoo.com

Received 28 September 2014; Revised 18 December 2014; Accepted 25 December 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Mohammad Subhi Al-batah 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.

Landslides are one of the dangerous natural phenomena that hinder the development in Penang Island, Malaysia. Therefore, finding the reliable method to predict the occurrence of landslides is still the research of interest. In this paper, two models of artificial neural network, namely, Multilayer Perceptron (MLP) and Cascade Forward Neural Network (CFNN), are introduced to predict the landslide hazard map of Penang Island. These two models were tested and compared using eleven machine learning algorithms, that is, Levenberg Marquardt, Broyden Fletcher Goldfarb, Resilient Back Propagation, Scaled Conjugate Gradient, Conjugate Gradient with Beale, Conjugate Gradient with Fletcher Reeves updates, Conjugate Gradient with Polakribiere updates, One Step Secant, Gradient Descent, Gradient Descent with Momentum and Adaptive Learning Rate, and Gradient Descent with Momentum algorithm. Often, the performance of the landslide prediction depends on the input factors beside the prediction method. In this research work, 14 input factors were used. The prediction accuracies of networks were verified using the Area under the Curve method for the Receiver Operating Characteristics. The results indicated that the best prediction accuracy of 82.89% was achieved using the CFNN network with the Levenberg Marquardt learning algorithm for the training data set and 81.62% for the testing data set.

1. Introduction

Landslide hazard is a particular case of natural hazard which is defined as the probability of occurrence within a specified period of time and within a given area of a potentially damaging phenomenon [1, 2]. Numerous occurrences of landslides have caused lives to perish and incurred losses in terms of financial stakes, across the entire world annually. However, the main causes behind the occurrence of the landslides are still unspecified. Different factors such as geological, topographic, physical, and human causes (disregard for sustainable form of developments) contribute to landslide occurrences [3]. Therefore, many studies have been conducted and different techniques have been applied to predict the occurrence of landslides. These techniques involve variations and mixtures in approaches, from logical, experience-based

analyses, extending to complex mathematical and computer based system.

Over the last two decades, a keen interest has been shown in the application of artificial neural networks (ANNs). It has been widely applied in forecasting, decision making, food industry, agriculture sector, and many other different applications [4–7]. The popularity of ANNs is due in part to their computational simplicity, finite parameterization, and stability. Different ANNs architecture such as the MLP, radial basis function (RBF), and recurrent neural networks (RNN) have been proposed in the literature [8]. Amongst all these models, the most commonly and widely used model for landslide is the MLP model [9, 10].

ANNs are one of the techniques which produce good accuracy when used to predict the occurrences of landslides [11]. However, an ideal method for predicting landslide

occurrence has not been agreed upon yet [12]. Therefore, an intelligent computer system is proposed to enable automatic prediction of landslide using MLP and CFNN in ANNs.

Penang Island is being subject of interest for many studies. Pradhan (2010) produced a landslide hazard map for Penang Island using MLP neural network. Five training sites from Penang island and nine different factors involved in their analysis include slope angle, slope aspect, curvature, distance from drainage, distance from lineament, geology, land cover, soil, and rain precipitation [13].

Pradhan et al. (2010) also investigated the possible application of an artificial neural network model and its cross-application of weights at three study areas in Malaysia, namely, Penang Island, Cameron Highlands, and Selangor. The weight of each factor was calculated. The factors are, namely, slope angle, slope aspect, plan curvature, altitude, stream power index, wetness index, distance from drainage, distance from road, distance from faults, geology, land use, soil texture, soil material, vegetation index, and topography. The results show that case of the weight using the same test area showed slightly higher accuracy than the weight used for the cross-applied area [10].

Lim et al. (2011) used probabilistic methods such as frequency ratio, statistical index, certainty factor and landslide susceptibility analysis, and logistic regression to produce landslide hazard maps for Penang Island. In their study, twelve factors including four topographic factors were used. The importance of the input factors was not estimated in their study [14].

Oh and Pradhan (2011) applied adaptive neurofuzzy inference system (ANFIS) with seven factors: altitude, slope angle, plan curvature, distance from drainage, distance from road, soil texture, and stream power index on an area of Penang Island covering only 8.064 km² of Penang Island [15].

Pang et al. (2012) used Decision Tree (DT) to produce landslide hazard mapping for Penang Island with the same twelve factors, used by [14]. DT model was calculated and constructed using the DT algorithm. The use of DT method improves the landslide hazard map where the percentage of past landslide event increases at three risk levels, that is, most hazardous, hazardous, and moderate, while the percentage is reduced in the nonhazardous level [16].

Digital Elevation Model has been generally used as the basic source for extracting the topographic factors such as slope aspect and curvatures. It also is one of the core database sources for several GIS applications [17]. For this study the DEM with 5 meter/pixel resolution was used to extract the slope angle, slope aspect, profile curvature, plan curvature, and general curvature.

Fourteen factors were used as the input features for the MLP and CFNN. These factors are slope angles, slope aspect, profile curvature, plan curvature, general curvature, distance from the road, distance from the fault lines, elevation, distance from the drainages, soil texture, land cover, vegetation cover, geology, and the rain precipitation as a triggering factor. The MLP and CFNN were trained with eleven learning algorithms to produce the most accurate prediction results. The 11 learning algorithms used were Levenberg Marquardt (LM), Broyden Fletcher Goldfarb (BFG), Resilient

Back Propagation (Rp), Scaled Conjugate Gradient (SCG), Conjugate Gradient with Beale (CGB), Conjugate Gradient with Fletcher Reeves updates (CGF), Conjugate Gradient with Polakribiere updates (CGP), One Step Secant (OSS), Gradient Descent (GD), Gradient descent with momentum and adaptive learning rate (GDX) and Gradient Descent with Momentum (GDM) algorithms.

The organization of this paper is as follows. The CFNN and MLP are explained as landslide prediction methods in Section 2. Section 3 introduces the study area and provides descriptions on the data collection and factor extraction. Results of the prediction performance are presented in Section 4. Conclusion is drawn in Section 5. Figure 1 describes the methodology for this work.

2. MLP and CFNN

The popularity of the MLP and CFNN comes from their stability, simplicity of application, and smaller structure size for a particular problem, as compared to the other structures [18]. The network learns the relationship between pairs of factors (inputs) and output (responses) vectors by altering the weight and bias values [19]. Figure 2 shows an example of a standard MLP and CFNN. It consists of three layers in the order of input, hidden, and output layer. Each layer consists of independent processing units called neurons [20].

These neurons receive inputs; each input value is multiplied by the weight (the strength of the input). The input is computed using a mathematical function that determines the activation values of the neuron and is then passed to the next layer. The output from the hidden layer is given by

$$h_j = f \left(\sum_{i=1}^I w_{ij} x_i + b_j \right); \quad \text{for } 1 \leq j \leq J, \quad (1)$$

where h_j is the output from hidden layer and x_i and w_{ij} denote the inputs and the weight from input i to hidden unit j in the first layer, respectively. b_j is the bias for hidden unit j and $f(\cdot)$ is the transfer function. For the hidden layer, the tan sigmoid function was used.

The predicted output of the k th node in the output layer is denoted as y_k which can be expressed as in

$$y_k = \sum_{j=1}^J w_{jk} h_j; \quad \text{for } 1 \leq k \leq K, \quad (2)$$

where w_{jk} denotes the weights from the hidden layer to the output layer. k denotes the number of outputs neurons. Combining (1) and (2), the complete representation of the output for the MLP network is obtained as in

$$y_k = \sum_{j=1}^J w_{jk} f \left(\sum_{i=1}^I w_{ij} x_i + b_j \right); \quad (3)$$

for $1 \leq j \leq J, \quad 1 \leq k \leq K.$

CFNN network shares the same structure and the work methodology with MLP network. However, CFNN includes

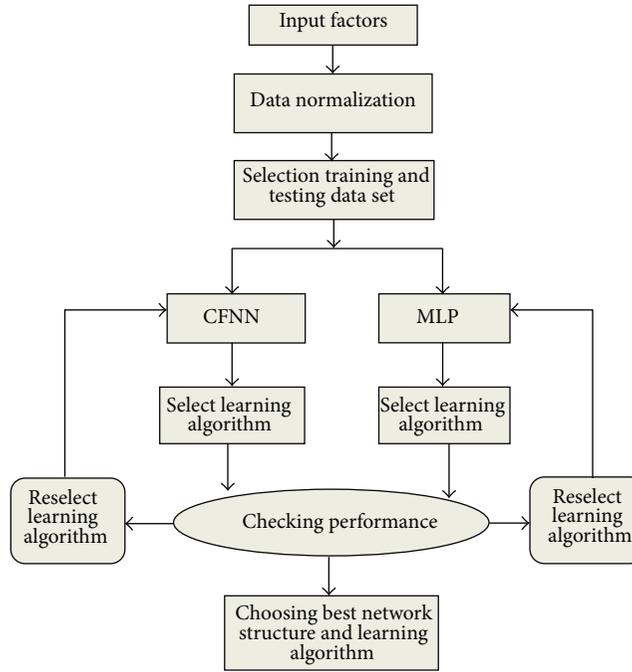


FIGURE 1: Flow chart of the work methodology.

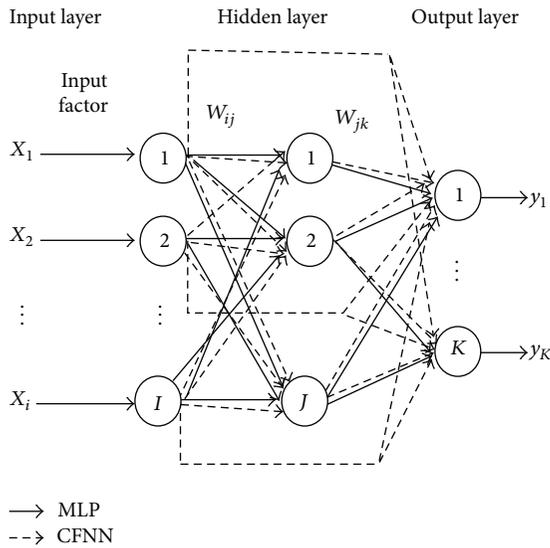


FIGURE 2: MLP and CFNN.

a weight connection from the input to the output layer and from each layer to the successive layers [21]. As shown in Figure 2, for the CFNN network with K output nodes, J hidden nodes, and I input nodes, the output of the k th neuron, y_k in the output layer is given by

$$y_k = \sum_{j=1}^J w_{jk} f \left(\sum_{i=1}^I w_{ij} x_i + b_j \right) + \left(\sum_{i=1}^I w_{ik} x_i + b_k \right); \quad (4)$$

for $1 \leq j \leq J, \quad 1 \leq k \leq K.$

Because of the fact that neural networks have numerous numbers of neurons, adjusting of the neural weights without a learning algorithm may be quite difficult. For that, various learning algorithms have been developed and established for two reasons: minimizing the error rate between the actual output and the output results and building up the weights, w_{ij} and w_{jk} , for the inputs factors [22, 23]. In this paper, MLP and CFNN were trained with eleven learning algorithms. Detailed descriptions on the learning algorithms can be found in [24]. Both MLP and CFNN with different learning algorithms are assessed based on their performance in producing landslide hazard map of Penang Island.

3. Data Collection and Preparation for the Neural Network

For this work, 14 factors were investigated and analyzed. The data for 14 factors were collected and extracted for the study area. This study is focused on Penang Island which is shown in Figure 3. Study area lies between $5^{\circ}15'$ and $5^{\circ}30'N$ latitude and $100^{\circ}10'$ and $100^{\circ}20'E$ longitude. It occupies an area of 285 km^2 and is one of the thirteen states of Malaysia. The island is bounded to the north and east by the state of Kedah, to the south by the state of Perak, and to the west by the Straits of Malacca and Sumatra (Indonesia). It consists of both the island of Penang and a coastal strip on the mainland which is known as Province Wellesley. The island of Penang is the study area in this research work. Penang Island experiences frequent landslides, which occur quite frequently during the rainy seasons [14, 15, 25]. Penang Island has a tropical climate with high temperatures of $29^{\circ}C$ to $32^{\circ}C$ and humidity ranging from 65% to 96%. Topographic elevations



FIGURE 3: Penang Island (source: Google maps).

vary between 0 m and 820 m above sea level, and the slope angle ranges from 0° to 87° . Flat lands make up 43.28% of the island. Geological data from the Department of Mineral and Geosciences show that Feringgi granite, Batu Maung granite, clay, and sand granite represent more than 72% of the study area's geology. Vegetation cover consists mainly of forests and fruit plantations.

Data collection on the geographical database of Penang Island was obtained through various government agencies. Factors such as geology, road, fault lines, elevation, drainage, soil texture, land cover, vegetation cover, and rain precipitation maps were obtained from Department of Survey and Mapping Malaysia (JUPEM), Department of Agriculture Malaysia (DOA), Department of Minerals and geosciences Malaysia (JMG), Geographic information System Center of Penang (Pusat PeGIS), Malaysian Meteorological Department (MMD), and Department of Irrigation and Drainage Malaysia (JPS). Topographic factors which include slope angle, slope aspect, profile curvature, plan curvature, and general curvature were extracted from the elevation data [16]. Landslide occurrence locations were also collected and determined. The range of each factor and the ratio of occurrence on the study area are shown in Table 1.

The data is prepared for the neural network, including training data set and testing data set. The data were normalized to range between 0 and 1, for each of the factors individually based on

$$\begin{aligned} & \text{Normalised sample } (i) \\ &= \frac{\text{sample } (i) - \text{minimum sample } (I)}{\text{maximum sample } (I) - \text{minimum sample } (I)}, \end{aligned} \quad (5)$$

where the sample (i) is the sample to be normalized and (I) is the minimum or the maximum sample value for every single input factor. The neural network outputs are represented by an output of 1 for landslide and 0 for no landslide. An effective neural network requires a comprehensive trained data set. Therefore 137572 data samples were selected from each factor in this analysis, where 68786 samples represent landslides and 68786 samples represent no landslides. Two-thirds of the data

(91715) were used for training and the remaining one-third (45857) was used for testing. The two neural networks, MLP and CFNN, were trained using the Matlab software.

To determine the network parameters, the experiments were carried out by varying the number of hidden neurons from 1 to 100. For each number of hidden neuron, the network was trained by varying the number of epochs from 1 to 1000. The purpose was to find the number of epoch that produced the best generalization for each number of hidden neuron. The optimum epoch and hidden neuron, which produced the minimum value of mean squared error for the testing set, was noted and its prediction accuracy was determined.

4. Results and Discussions

Two characteristics were used for the neural network performance analysis, which are the accuracy and the MSE. Model validation using MSE is tested by calculating the mean squared errors after each epoch. The MSE is defined as the average squared error between the actual output and the predicted output. The MSE at every epoch is given by

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (y(i) - \hat{y}(i))^2, \quad (6)$$

where $y(i)$ and $\hat{y}(i)$ are the actual output and the predicted output for a given set of estimated parameters after t epochs, respectively, and N is the number of data that were used to calculate the MSE.

To verify the accuracy of each model, the Receiver Operating Characteristics (ROC) method was used and the Area Under the Curve method (AUC) was calculated for all the models. AUC is one of the popular accepted methods for models prediction in natural hazard and the extracted AUC becomes the value of the accuracy.

The ROC plots the false positive rate on the X -axis and the false negative rate on the Y -axis. The plot shows the tradeoff between the two rates, where AUC is one of the indicators computed based on ROC. In addition to that, the AUC explains the accuracy of the model in predicting landslides. In general, the lowest value of AUC is 0.5, which means that the model does not predict any better than a random approach.

Table 2 shows the testing performance for the training data sets that were achieved from the standard MLP and CFN, using the eleven different learning algorithms based on the 14 input factors. Based on the results in Table 2, it can be clearly seen that the performance values vary considerably across the model of the neural network and the learning algorithms. The best performance achieved was obtained through the CFNN model with LM learning algorithm. The accuracy is 82.89% with MSE of 0.0620. The worst accuracy of 71.15% and MSE of 0.1839 was obtained through MLP model with GDM learning algorithm.

The best accuracy rate obtained using MLP was achieved by the LM algorithm, that is, accuracy of 81.57% with MSE of 0.0910. On the contrary, the worst accuracy was 71.15% with MSE of 0.1839 achieved by the GDM algorithm. For CFNN, the worst performance was achieved using the GD algorithm with 71.24% with MSE of 0.1607. Meanwhile,

TABLE 1: Landslide causative factor's ranges and ratio.

Factors	Class	Area (pixels)		Ratio	
		Total area	Landslide occurrence area	Total area	Landslide occurrence
Elevation (meter)	0–79	6641099	10902	55.2%	15.85%
	80–159	1390848	13026	11.56%	18.94%
	160–239	1186046	12181	9.86%	17.71%
	240–319	882426	9687	7.33%	14.08%
	320–399	608281	8491	5.06%	12.34%
	400–479	449927	6900	3.74%	10.03%
	480–559	391076	4461	3.25%	6.49%
	>560	481490	3138	4%	4.56%
Slope angle (degree)	0–7	6318295	6100	52.52%	8.87%
	8–15	1515746	12512	12.6%	18.19%
	16–23	2003138	19769	16.65%	28.74%
	24–31	1490172	19504	12.39%	28.35%
	32–39	554603	8537	4.61%	12.41%
	40–47	121293	1825	1.01%	2.65%
	>48	539	539	0.23%	0.78%
Slope aspect (degree)	North	811040	7819	6.74%	11.37%
	North-East	980909	11490	8.15%	16.70%
	East	996431	9081	8.28%	13.20%
	South-East	723126	7515	6.01%	10.93%
	South	796410	7164	6.62%	10.41%
	South-West	903353	9771	7.51%	14.20%
	West	958631	8657	7.97%	12.59%
	North-west	654463	5217	5.44%	7.58%
Flat	52066830	2072	43.28%	3.01%	
General curvature	Convex	2992360	35717	24.86%	51.92%
	Concave	3148149	26894	26.17%	39.10%
	Flat	5890684	6175	48.97%	8.98%
Profile curvature	Convex	3098090	26030	25.75%	37.84%
	Concave	2886983	36014	23.99%	52.35%
	Flat	6046120	6742	50.99%	9.80%
Plan curvature	Convex	361329	31779	30.18%	46.19%
	Concave	2594977	31281	21.56%	45.47%
	Flat	5804887	5726	48.26%	8.30%
Land cover	Forest, bush, swam	6112837	52195	50.81%	75.88%
	Vegetation	1617410	10951	13.44%	15.92%
	Transport	894789	2288	7.44%	3.33%
	Settlement	1476907	404	12.28%	0.59%
	Cemetery	138389	1442	1.15%	2.1%
	Mining	31931	0	0.27%	0.0%
	Industry	192485	13	1.6%	0.019%
	Government institution	156625	177	1.3%	0.26%
	Public facility	217492	155	1.18%	0.23%
	Plains, hills	303211	501	2.51%	0.73%
	Buildings	24327	0	0.2%	0%
	Religious area	41945	159	0.35%	0.23%
	Business	205595	0	1.71%	0%
	Sea, lake, river	267665	430	2.22%	0.63%
	Public utility	56358	71	0.47%	0.1%
	Livestock	81141	0	0.67%	0%
Education	212085	0	1.76%	0%	

TABLE 1: Continued.

Factors	Class	Area (pixels)		Ratio	
		Total area	Landslide occurrence area	Total area	Landslide occurrence
Vegetation cover	Forest, plant, Bush	5441433	51148	45.23%	74.36%
	Swamp	412778	960	3.43%	1.4%
	Mixed farms	258984	80	2.15%	0.12%
	Fruit farm	145514	1683	1.21%	2.45%
	Oil farm	968301	9023	8.05%	13.12%
	Sugarcane	176678	0	1.47%	0%
	Vegetable	156	0	0.0013%	0%
	Farm	55599	87	0.46%	0.13%
	Coconut	79156	0	0.66%	0.0%
	Pineapple	81	0	0.00067%	0.0%
	Paddy	126263	0	1.05%	0.0%
	Rubber	2241	0	0.019%	0.0%
	Others	65338	208	0.54%	0.3%
	None	4298671	5597	35.73%	8.14%
Distance from fault line	0–100	641379	7164	5.33%	10.41%
	101–200	651750	3427	5.42%	4.98%
	201–300	643860	1939	5.35%	2.82%
	301–400	637145	4272	5.30%	6.21%
	401–500	630562	5715	5.24%	8.31%
	501–600	611528	4756	5.08%	6.92%
	601–700	588891	5671	4.89%	8.24%
	701–800	549700	4855	4.57%	7.06%
	801–900	511179	6090	4.25%	8.85%
	9001–1000	457995	3593	3.81%	5.22%
>1000	6107204	21298	50.76%	30.96%	
Distance from road	0–49	3807202	5863	31.64%	8.52%
	50–99	1044665	2056	8.68%	2.99%
	100–149	748374	3324	6.22%	4.83%
	150–199	598005	3053	4.97%	4.44%
	200–249	504546	3267	4.19%	4.75%
	250–299	440547	3368	3.66%	4.9%
	300–349	389709	3787	3.24%	5.51%
	350–399	354222	4152	2.94%	6.04%
	400–449	328068	2805	2.73%	4.08%
	>450	3815855	37111	31.72%	53.95%
Distance from drainage	0–49	4229076	41931	35.15%	60.96%
	50–99	2214851	11711	18.41%	17.03%
	100–149	1497866	6864	12.45%	9.98%
	150–199	980562	3980	8.15%	5.79%
	200–249	643367	1943	5.35%	2.82%
	250–299	455199	1003	3.78%	1.46%
	300–349	344950	586	2.87%	0.85%
	350–399	274824	517	2.28%	0.75%
	400–449	224731	205	1.87%	0.30%
	>450	1165767	46	9.69%	0.07%
Geology	Muka head microgranite	174024	1431	1.45%	2.08%
	Feringgi, granite	2343321	22931	19.48%	33.34%
	Tanjung Bunga granite	2751458	19292	22.87%	28.05%
	Sungai area granite	318909	500	2.65%	0.73%
	Batu Maung granite	3359896	24365	27.93%	35.42%
	Clay, sand, granite	3083585	267	25.63	0.39%

TABLE 1: Continued.

Factors	Class	Area (pixels)		Ratio	
		Total area	Landslide occurrence area	Total area	Landslide occurrence
Soil texture	Sand	133506	0	1.11%	0.00%
	Sandy clay	443859	505	3.69%	0.74%
	Loam	6535957	67126	54.33%	97.59%
	Sandy loam	1606303	790	13.35%	1.15%
	Silty clay	332707	55	2.77%	0.08%
	Urban land	2978861	309	24.76%	0.45%
Rain precipitation	2254–2319.8	85398	849	0.71%	1.23%
	2319.9–2379.3	549900	1571	4.57%	2.28%
	2379.4–2433.2	1489192	10850	12.38%	15.77%
	2433.3–2481.9	2815573	16632	23.4%	24.18%
	2481.0–2535.8	3202204	6815	26.62%	9.91%
	2535.9–2595.8	3003842	22832	24.97%	33.19%
	2595.9–2661.1	559762	6617	4.65%	9.61%
	2661.2–2722.8	227570	1755	1.89%	2.55%
2733.9–2903	97752	865	0.81%	1.26%	

TABLE 2: The training accuracy (%) and MSE for MLP and CFNN using the eleven learning algorithms with data set.

Network type	MLP				CFNN				
	Learning algorithm	Accuracy	MSE	Hidden nodes	Epoch	Accuracy	MSE	Hidden nodes	Epoch
LM		81.57	0.0910	82	22	82.89	0.0620	72	33
BFG		81.35	0.1150	60	191	79.68	0.1326	68	152
Rpro		77.62	0.1535	52	274	77.73	0.1290	89	190
SCG		77.22	0.1548	81	178	77.19	0.1280	142	64
CGB		77.28	0.1534	80	147	78.14	0.1258	72	231
CGF		75.75	0.1625	31	134	76.76	0.1334	23	127
CGP		76.54	0.1578	55	144	76.28	0.1370	82	148
OSS		76.13	0.1624	49	168	74.98	0.1422	18	147
GD		71.23	0.1836	80	989	71.24	0.1607	82	968
GDX		72.34	0.1791	19	140	72.27	0.1590	25	122
GDM		71.15	0.1839	87	999	71.24	0.1606	89	977

the LM algorithm showed the best accuracy regardless of the neural network model. The LM learning algorithm achieved the best accuracy of 82.89%, with MSE of 0.0620 and 81.57% with MSE of 0.0910 for CFNN and MLP, respectively. On the other hand, GDM algorithm has the worst results in MLP neural network, whereas GD algorithm has the worst results in CFNN.

Overall, CFNN model achieved better accuracy and MSE as compared to MLP model, using six learning algorithms, that is, LM, Rprop, CGB, CGF, GD, and GDM, while for learning algorithms including BFG, SCG, OSS, CGP, and GDX, the MLP achieved better accuracy compared to CFNN.

In Table 3, the testing data sample was tested by using the same networks parameters. As expected the test accuracy result followed the training accuracy result where the CFNN with LM training algorithm achieved the best accuracy. Figures 4, 5, 6, and 7 show the ROC of CFNN and MLP with 11 learning algorithms applied on the testing data set.

TABLE 3: Accuracy obtained using testing data set.

Network type	MLP	CFNN	
	Learning algorithm	Testing data accuracy	Testing data accuracy
LM		81.11	81.62
BFG		79.69	75.80
Rprop		75.61	77.93
SCG		75.58	76.34
CGB		75.52	75.81
CGF		74.47	75.16
CGP		74.96	74.82
OSS		74.37	73.87
GD		71.04	71.17
GDX		71.87	71.84
GDM		70.90	70.91

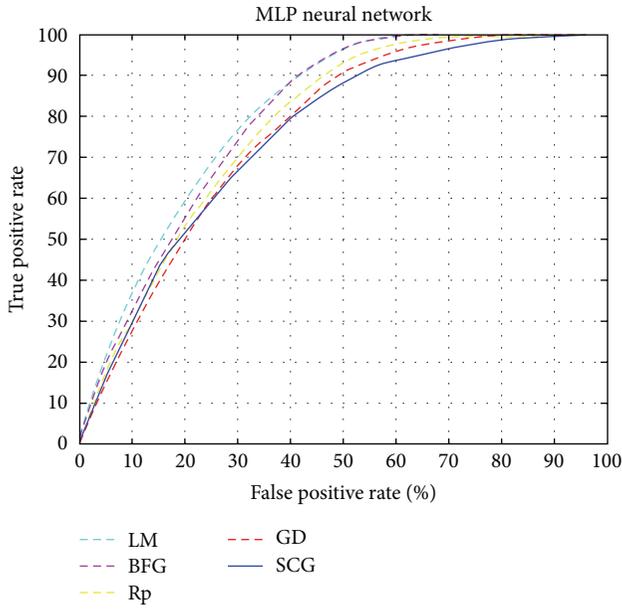


FIGURE 4: ROC curve for MLP neural network trained with LM, BFG, Rp, GD, and SCG.

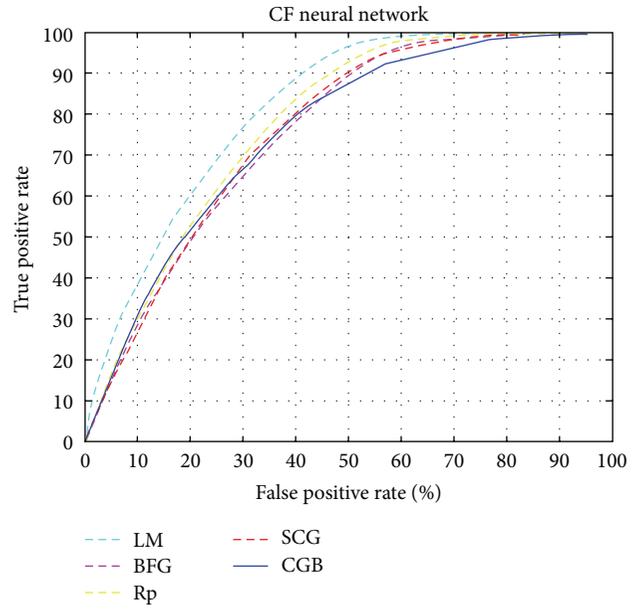


FIGURE 6: ROC curve for CF neural network trained with LM, BFG, Rp, SCG, and CGB.

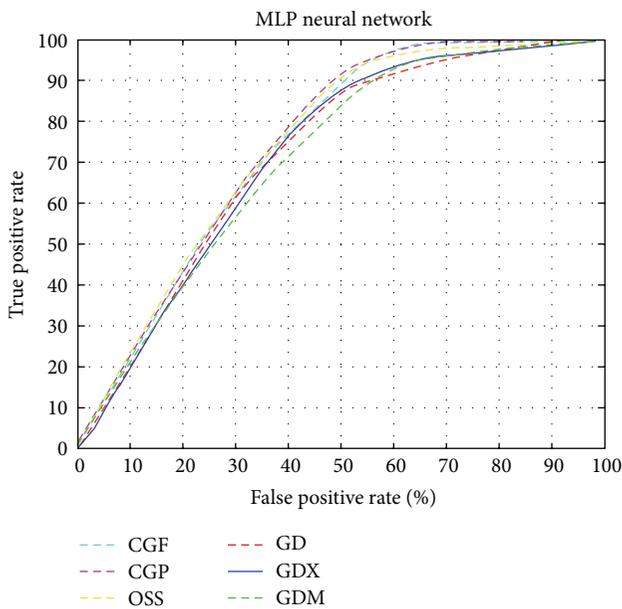


FIGURE 5: ROC curve for MLP neural network trained with CGF, CGP, OSS, GD, GDX, and SCG.

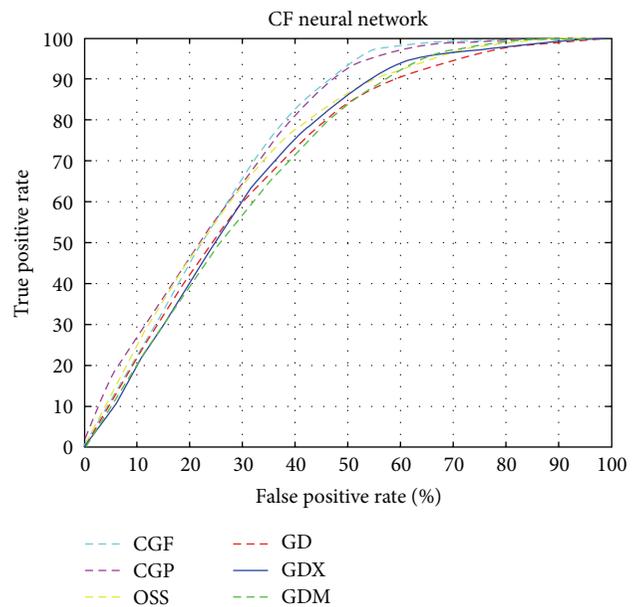


FIGURE 7: ROC curve for CF neural network trained with CGF, CGP, OSS, GD, GDX, and SCG.

5. Conclusion

In this paper, fourteen suitable factors were collected and applied as input factors for ANN models. Two efficient neural network models, MLP and CFNN, are proposed and compared using eleven learning algorithms. The 14 factors show a good performance in predicting the landslide occurrence of Penang Island with accuracy up to 81.62%. The comparison results show that the CFNN network trained with LM can successfully be adopted for prediction of the landslide

with significantly high performance. Moreover, applying the CFNN for prediction of the landslide on different study areas could be subject of interest in the future work.

Conflict of Interests

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

Acknowledgments

This research was supported by Ministry of Education, Malaysia (Grant no. 203/PJJAUH/6711279), Japan Science and Technology Agency (JST)/Japan International Cooperation Agency (JICA), and Science and Technology Research Partnership for Sustainable Development (SATREPS). The authors would also like to thank the JKR, JPS, JMG, and PEGIS, Malaysia, for the data used in this research.

References

- [1] D. J. Varnes, *Landslide Hazard Zonation: Preview of Principals and Practice*, vol. 3 of *Natural Hazards*, UNESCO, International Association of Engineering Geologists, Commission on Landslides and Other Mass Movements on Slopes, Paris, France, 1984.
- [2] D. J. Varnes, *Landslide Hazard Zonation: Preview of Principals and Practice. Paris, UNESCO, Int Assciation of Engineering Geologists, Commission on Landslides and Other Mass Movements on Slopes, Natural Hazards V-3 Pp176*, 1984.
- [3] J. N. Hutchinson, "Keynote paper: landslide hazard assessment," in *Landslides, Proc. Sixth Int. Symp. on Landslides, February, Christchurch, New Zealand*, D. H. Bell, Ed., vol. 3, pp. 1805–1841, A.A. Balkema, Rotterdam, The Netherlands, 1995.
- [4] G. Zhang, B. E. Patuwo, and M. Y. Hu, "Forecasting with artificial neural networks: the state of the art," *International Journal of Forecasting*, vol. 14, no. 1, pp. 35–62, 1998.
- [5] L. Sanzogni and D. Kerr, "Milk production estimates using feed forward artificial neural networks," *Computers and Electronics in Agriculture*, vol. 32, no. 1, pp. 21–30, 2001.
- [6] M. Kumar, N. S. Raghuwanshi, R. Singh, W. W. Wallender, and W. O. Pruitt, "Estimating evapotranspiration using artificial neural network," *Journal of Irrigation and Drainage Engineering*, vol. 128, no. 4, pp. 224–233, 2002.
- [7] M. Lashkarbolooki, Z. S. Shafipour, and A. Z. Hezave, "Trainable cascade-forward back-propagation network modeling of spearmint oil extraction in a packed bed using SC-CO₂," *The Journal of Supercritical Fluids*, vol. 73, pp. 108–115, 2013.
- [8] M. S. Alkhasawneh, U. K. Bt Nghah, T. L. Tien, and N. A. B. Mat Isa, "Landslide susceptibility hazard mapping techniques review," *Journal of Applied Sciences*, vol. 12, no. 9, pp. 802–808, 2012.
- [9] S. Lee, J. Ryu, K. Min, and J. Won, "Development of two artificial neural network methods for landslide susceptibility analysis," in *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS '01)*, vol. 5, pp. 2364–2366, 2001.
- [10] B. Pradhan, L. Saro, and M. F. Buchroithner, "A GIS-based back-propagation neural network model and its cross-application and validation for landslide susceptibility analyses," *Computers, Environment and Urban Systems*, vol. 34, no. 3, pp. 216–235, 2010.
- [11] S. Lee, J. Choi, and K. Min, "Probabilistic landslide hazard mapping using GIS and remote sensing data at Boun, Korea," *International Journal of Remote Sensing*, vol. 25, no. 11, pp. 2037–2052, 2004.
- [12] F. Guzzetti, A. Carrara, M. Cardinali, and P. Reichenbach, "Landslide hazard evaluation: a review of current techniques and their application in a multi-scale study, Central Italy," *Geomorphology*, vol. 31, no. 1–4, pp. 181–216, 1999.
- [13] B. Pradhan and S. Lee, "Delineation of landslide hazard areas on Penang Island, Malaysia, by using frequency ratio, logistic regression, and artificial neural network models," *Environmental Earth Sciences*, vol. 60, no. 5, pp. 1037–1054, 2010.
- [14] K.-W. Lim, L. T. Tay, and H. Lateh, "Landslide hazard mapping of Penang island using probabilistic methods and logistic regression," in *Proceedings of the IEEE International Conference on Imaging Systems and Techniques (IST '11)*, pp. 273–278, May 2011.
- [15] H.-J. Oh and B. Pradhan, "Application of a neuro-fuzzy model to landslide-susceptibility mapping for shallow landslides in a tropical hilly area," *Computers & Geosciences*, vol. 37, no. 9, pp. 1264–1276, 2011.
- [16] P. K. Pang, L. T. Tay, and H. Lateh, "Landslide hazard mapping of Penang Island using decision tree model," in *Proceedings of the International Conference on Systems and Electronic Engineering (ICSEE '12)*, December 2012.
- [17] Q. Zhou and X. Liu, "Analysis of errors of derived slope and aspect related to DEM data properties," *Computers and Geosciences*, vol. 30, no. 4, pp. 369–378, 2004.
- [18] M. Barletta and A. Gisario, "An application of neural network solutions to laser assisted paint stripping process of hybrid epoxy-polyester coating on aluminum substrates," *Surface and Coatings Technology*, vol. 200, no. 24, pp. 6678–6689, 2006.
- [19] M. S. Al-Batah, N. A. Mat Isa, K. Z. Zamli, Z. M. Sani, and K. A. Azizli, "A novel aggregate classification technique using moment invariants and cascaded multilayered perceptron network," *International Journal of Mineral Processing*, vol. 92, no. 1–2, pp. 92–102, 2009.
- [20] S. Haykin, *Neural Networks a Comprehensive Foundation*, 2nd edition, 1999.
- [21] M. Kurban and U. B. Filik, "A new approach for next day load forecasting integrating artificial neural network model with weighted frequency bin blocks," in *Neural Information Processing*, M. Ishikawa, K. Doya, H. Miyamoto, and T. Yamakawa, Eds., vol. 4985, pp. 703–712, Springer, Berlin, Germany, 2008.
- [22] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK, 1995.
- [23] G. Hinton, D. E. Rumelhart, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, D. E. Rumelhart and J. L. McClelland, Eds., vol. 1, pp. 318–362, MIT Press, Cambridge, UK, 1986.
- [24] K. K. Aggarwal, Y. Singh, P. Chandra, and M. Puri, "Bayesian regularization in a neural network model to estimate lines of code using function points," *Journal of Computer Sciences*, vol. 1, no. 4, pp. 505–509, 2005.
- [25] New Starits Time, *Don't Over-Develop Penang to Avoid Landslides*, 2012, <http://www.nst.com.my/latest/don-t-over-develop-penang-to-avoid-landslides-1.193992>.

Research Article

System Optimization for Temporal Correlated Cognitive Radar with EBPSK-Based MCPC Signal

Peng Chen and Lenan Wu

School of Information Science and Engineering, Southeast University, Nanjing 210096, China

Correspondence should be addressed to Peng Chen; chenpengdsp@seu.edu.cn

Received 19 August 2014; Accepted 21 September 2014

Academic Editor: Yudong Zhang

Copyright © 2015 P. Chen and L. Wu. 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 system optimization is considered in cognitive radar system (CRS) with extended binary phase shift keying- (EBPSK-) based multicarrier phase-coded (MCPC) signal. A novel radar working scheme is proposed to consider both target detection and estimation. At the detection stage, the generalized likelihood ratio test (GLRT) threshold is deduced, and the GLRT detection probability is given. At the estimation stage, an approach based on Kalman filtering (KF) is proposed to estimate target scattering coefficients (TSC), and the estimation performance is improved significantly by exploiting the TSC temporal correlation. Additionally, the optimal waveform is obtained to minimize the mean square error (MSE) of KF estimation. For the practical consideration, iteration algorithms are proposed to optimize the EBPSK-based MCPC signal in terms of power allocation and coding matrix. Simulation results demonstrate that the KF estimation approach can improve the estimation performance by 25% compared with maximum a posteriori MAP (MAP) method, and the KF estimation performance can be further improved by 90% by optimizing the transmitted waveform spectrum. Moreover, by optimizing the power allocation and coding matrix of the EBPSK-based MCPC signal, the KF estimation performances are, respectively, improved by 7% and 8%.

1. Introduction

The cognitive radar system (CRS) as the future trend of the radar systems, compared with the traditional ones, mainly includes three different aspects [1]: (1) CRS can sense the targets and environment; (2) the transmitted waveform is adaptively optimized to improve the detection and estimation performance; (3) the transmitter, environment, and receiver form a closed loop feedback system. Therefore, optimizing the transmitted waveform based on the working environment becomes a popular research direction in the CRS [2–8].

For the modeling of CRS, a point is utilized to model the target [9]. The echo waveform from the point target is only the delay or Doppler frequency shift of the original one. However, when the size of target is large enough to occupy many resolution cells, the echo signals from different scattering cells are superimposed, and this type of target is modeled as an extended target (ET) [10]. In addition, the target impulse response (TIR), which is often referred to as the high resolution range profile (HRRP) in the automatic target recognition (ATR) problem [11], can be used to describe ET

based on the assumption of the linear time-invariant target [12, 13]. However, the view angle between the target and radar changes, and the ET does not satisfy this assumption. Hence, an exponential correlation model is proposed to describe this time dynamic characteristic, where the TIR is stationary with time and uncorrelated among different resolution cells, namely, wide sense stationary-uncorrelated scattering (WSSUS). Therefore, the TIR of this type target can be estimated by Kalman filtering (KF) [5, 14], where only the estimation is taken into consideration and under the present assumption of the target. However, the practical radar systems should detect the target before estimating.

The performance of the target estimation and detection can be significantly improved by optimizing the transmitted waveform in the CRS. During the stage to estimate the target scattering coefficients (TSC), which is essential the Fourier transform of TIR, the performance can be improved by optimizing the transmitted waveform to maximize the mutual information (MI) between the echo signal and TSC [12]. Furthermore, when the precise priori knowledge of

the target cannot be obtained, an eigensubspace projection-based method is proposed to enlarge the separation between the echo signals from different targets in [15, 16]. During the detection stage, in order to maximize the probability of target detection, more power is concentrated on frequencies with relatively large TSC in the additive white Gaussian noise (AWGN) systems or relatively high ratio between the target and clutter in the clutter interference systems [17–20]. However, these optimized waveforms have arbitrary nature and are not conveniently generated in the practical radar systems.

The multicarrier phase-coded (MCPC) signal is first applied in the wideband radar systems by Levanon [21–23]. Furthermore, the MCPC waveform can achieve the optimal target detection performance with higher spectrum efficiency than the linear frequency modulation (LFM) signal [24, 25]. Unlike the P3 or P4 signal, the MCPC signal has a thumbtack-shaped ambiguity function and is easily generated in the practical radar systems. However, the nonconstant amplitude of MCPC waveform cannot take full advantage of the nonlinear amplifier [26]. On the modulation of MCPC waveform, an extended binary phase shift keying (EBPSK) modulation proposed by Wu et al. is adopted in this work [27, 28]. EBPSK is more flexible than BPSK, where a small angle phase and jump time are utilized to distinguish the modulation waveform of code 1 from that of code 0, which tightens the spectrum of the transmitted waveform [25, 29, 30]. In the wireless communication system, reference [31] has confirmed that EBPSK can achieve the same theoretical bit error rate (BER) performance as BPSK and achieve higher spectral efficiency with the same bit rate by tuning modulation parameters. Therefore, we first utilize the EBPSK-based MCPC signal in the CRS, since the tight spectrum is easy to optimize.

In this work, the problem of system optimization is considered in the CRS with EBPSK-based MCPC signal, and we propose a new radar working scheme, where both the target detection and estimation are taken into account. During the initial stage of the target detection, the generalized likelihood ratio test (GLRT) algorithm based on constant false alarm rate (CFAR) is utilized. In the presence of ET, the approach based on KF is proposed to estimate TSC by exploiting the temporal correlation, and the optimal spectrum is obtained to minimize the trace of the mean square error (MSE) matrix of KF estimation. In addition, the iteration algorithms are proposed to optimize the amplitudes and coding matrix of EBPSK-based MCPC signal, respectively.

The remainder of this work is organized as follows. In Section 2, the system model of the CRS with EBPSK-based MCPC signal and the new radar working scheme are given. In Section 3, we optimize this radar system, including the GLRT target detection based on CFAR, the maximum a posteriori probability (MAP) receive filter, the TSC estimation based on KF, the optimal power allocation, and coding matrix. Section 4 gives the simulation results. Finally, Section 5 concludes this work.

The notations used in this work are defined as follows. Symbols for matrices (upper case) and vectors (lower case)

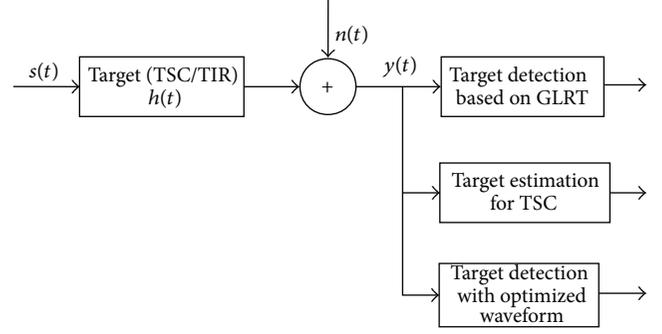


FIGURE 1: The system model of cognitive radar with EBPSK-based MCPC signal.

are in boldface. $(\cdot)^H$, $\text{diag}\{\cdot\}$, \mathbf{I}_L , $\mathcal{N}(\mathbf{0}, \mathbf{R})$, $|\cdot|$, $\|\cdot\|_2$, $\det(\cdot)$, $\mathcal{E}\{\cdot\}$ and $\text{Tr}\{\cdot\}$ denote the conjugate transpose (Hermitian), the diagonal matrix, the identity matrix of size L , the Gaussian distribution with zero mean and covariance being \mathbf{R} , the absolute value, the ℓ_2 norm, the determinant of a matrix, the expectation, and the trace of a matrix, respectively.

2. The System Model of Cognitive Radar with EBPSK-Based MCPC Signal

The system model of cognitive radar considered in this work to detect and estimate the ET is shown in Figure 1, where the transmitted waveform is EBPSK-based MCPC signal $s_k(t)$ and the TIR of ET is $h_k(t)$ during the k th pulse. Then the echo signal $y_k(t)$ can be described as

$$y_k(t) = s_k(t) * h_k(t) + n_k(t), \quad k = 1, 2, 3, \dots, \quad (1)$$

where $*$ denotes the convolution operation, and $n_k(t)$ denotes the AWGN. The EBPSK-based MCPC signal can be expressed as

$$s_k(t) = \sum_{l=1}^L a_{k,l} \sum_{r=1}^R r_{k,l,r}(t), \quad (2)$$

where $r_{k,l,r}(t) \triangleq C_{k,l,r} s_1(t - T_s r, f_l) + (1 - C_{k,l,r}) s_0(t - T_s r, f_l)$, $C_{k,l,r} \in \{0, 1\}$ denotes the entry at l th row and r th column of the coding matrix \mathbf{C}_k , $a_{k,l}$ denotes the amplitude of the l th subcarrier, and $f_l \triangleq f_c + (l - 1)\Delta f$ denotes the carrier frequency of the l th subcarrier, f_c denotes the carrier frequency of the first subcarrier, and Δf denotes the frequency interval between subcarriers. $s_1(t, f)$ and $s_0(t, f)$ are the waveform of code 0 and 1 with EBPSK modulation, respectively,

$$s_0(t, f) = \cos(2\pi f t), \quad 0 \leq t \leq T_s$$

$$s_1(t, f) = \begin{cases} -\cos(2\pi f t), & 0 \leq t \leq \tau \\ \cos(2\pi f t), & \tau \leq t \leq T_s, \end{cases} \quad (3)$$

where $T_s \triangleq N/f_c$ ($N = 1, 2, 3, \dots$) denotes the time interval of the waveform for one code and $\tau \triangleq K/f_c$ ($K = 1, 2, 3, \dots, K \leq N$) denotes the jump interval of the EBPSK

modulation. Then we can get the spectrum of transmitted waveform by the Fourier transform

$$S_k(f) = \sum_{l=1}^L a_{k,l} \sum_{r=1}^R R_{k,lr}(f) \exp(-j2\pi f T_s r), \quad (4)$$

where $R_{k,lr}(f) \triangleq C_{k,lr} S_1(f, f_l) + (1 - C_{k,lr}) S_0(f, f_l)$ and $S_i(f, f_l)$ ($i \in \{0, 1\}$) is the Fourier transform of $s_i(t, f_l)$; then the spectrum of echo signal $Y(f)$ is

$$Y_k(f) = H_k(f) S_k(f) + N_k(f). \quad (5)$$

The discrete form in the frequency domain can be obtained as

$$\mathbf{y}_k = \mathbf{S}_k \mathbf{g}_k + \mathbf{n}_k, \quad (6)$$

where the length of \mathbf{y}_k is M , $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I}_M)$ denotes the discrete Gaussian noise, σ_n^2 denotes the variance of noise, \mathbf{y}_k denotes the discrete form of $Y_k(f)$, \mathbf{g}_k denotes the TSC, that is, the discrete form of $H_k(f)$, and the diagonal matrix $\mathbf{S}_k \triangleq \text{diag}\{\mathbf{s}_k\}$, where \mathbf{s}_k is the discrete form of $S_k(f)$. According to [14], the exponential correlation model of the TSC is

$$\mathbf{g}_k = e^{-T/\tau} \mathbf{g}_{k-1} + \mathbf{u}_{k-1}, \quad (7)$$

where \mathbf{g}_k denotes the TSC during the k th pulse, T denotes the pulse repetition interval (PRI) in the radar system, τ denotes the temporal decay constant, and $\mathbf{u}_{k-1} \sim \mathcal{N}(\mathbf{0}, (1 - e^{-T/\tau}) \boldsymbol{\Sigma}_g)$ follows the Gaussian distribution. When $\mathbf{g}_0 = \mathbf{0}$,

$$\mathbf{g}_k = \sum_{m=1}^{k-1} e^{-mT/\tau} \mathbf{u}_{k-1-m}, \quad (8)$$

and $\mathbf{g}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k \triangleq (1 - e^{-2T/\tau}) \boldsymbol{\Sigma}_g)$. To simplify the analysis in this work, the correlation between the individual scatters is small enough to assume that $\boldsymbol{\Sigma}_g$ is a diagonal matrix; then $\sigma_{k,Gm}^2$ denotes the variance of $g_{k,m}$, where $g_{k,m}$ is the m th entry of \mathbf{g}_k .

This work proposes a new radar working scheme for the CRS with EBPSK-based MCPC signal. As shown in Figure 2, the target detection and estimation stages in this radar system are described as follows.

- (1) Initially, the nonoptimized EBPSK-based MCPC signal with power evenly distributed across all subcarriers is transmitted, and the coding matrix is a random ± 1 matrix.
- (2) The CRS detects the target via the method based on GLRT, and the detection threshold is obtained by the theory of CFAR. If the power of echo signal is greater than threshold, then the CRS detects the presence of target, otherwise not.
- (3) When the GLRT detects the presence of target, we propose an approach to estimate the TSC from the echo signal based on KF.
- (4) In order to improve the performance of KF estimation, this work optimizes the EBPSK-based MCPC signal in terms of power allocation among subcarriers and coding matrix;

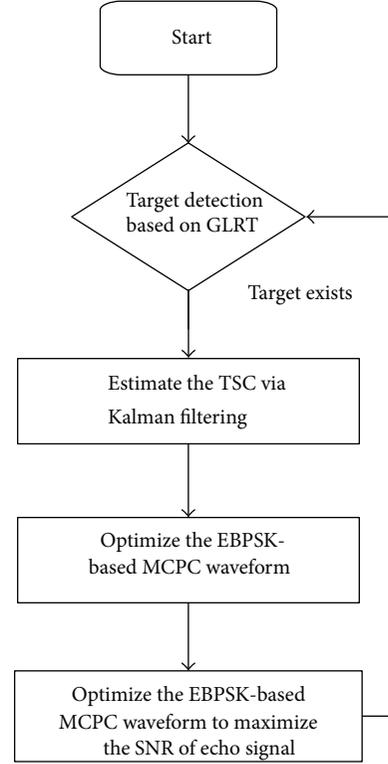


FIGURE 2: The proposed working scheme for the CRS with EBPSK-based MCPC signal.

- (5) With the knowledge of the estimated TSC, the signal-to-noise ratio (SNR) of the echo signal can be further improved, which attains a better detection probability for the particular target.

More details to describe this working scheme will be given in the following sections.

3. The Optimization of CRS with EBPSK-Based MCPC Signal

3.1. The Initial Target Detection Based on GLRT. Assuming that the target is present H_1 and absent H_0 , the echo waveform is

$$H_0: \mathbf{y}_k = \mathbf{n}_k, \quad (9)$$

$$H_1: \mathbf{y}_k = \mathbf{S}_k \mathbf{g}_k + \mathbf{n}_k.$$

With the Gaussian assuming of \mathbf{g}_k and \mathbf{n}_k , the distribution of echo signal is also Gaussian, and the logarithmic likelihood ratio is

$$\begin{aligned} l(\mathbf{y}_k | \mathbf{g}_k) &\triangleq \ln \frac{p(\mathbf{y}_k | \mathbf{g}_k, H_1)}{p(\mathbf{y}_k | H_0)} \\ &= \frac{1}{\sigma_n^2} (2\mathbf{y}_k^H - \mathbf{g}_k^H \mathbf{S}_k^H) \mathbf{S}_k \mathbf{g}_k, \end{aligned} \quad (10)$$

where $p(\mathbf{y}_k | H_0)$ is the conditional probability distribution, and the constant term is eliminated in the second

equation of (10). According to the GLRT theory, without the knowledge of TIR, the detection process can be expressed as

$$\{l(\mathbf{y}_k) \triangleq l(\mathbf{y}_k | \bar{\mathbf{g}}_k)\} \underset{H_0}{\overset{H_1}{\geq}} \lambda, \quad (11)$$

where

$$\begin{aligned} \bar{\mathbf{g}}_k &= \arg \max_{\mathbf{g}_k} l(\mathbf{y}_k | \mathbf{g}_k) \\ &= \arg \max_{\mathbf{g}_k} \|\mathbf{y}_k - \mathbf{S}_k \mathbf{g}_k\|_2 = \mathbf{S}_k^{-1} \mathbf{y}_k. \end{aligned} \quad (12)$$

Then we can obtain the GLRT detection

$$l(\mathbf{y}_k) = \frac{1}{\sigma_n^2} \mathbf{y}_k^H \mathbf{y}_k \underset{H_0}{\overset{H_1}{\geq}} \lambda. \quad (13)$$

Therefore, without the knowledge of TSC, the GLRT detection is equivalent to the energy detecting. If the energy $\|\mathbf{y}_k\|_2^2$ is greater than the threshold $\lambda \sigma_n^2$, the target exists, otherwise not.

In the CRS, CFAR is adopted as the target detection criterion, and the echo signal meets the χ^2 distribution with M degrees of freedom when the target is absent, so the false alarm rate is

$$\begin{aligned} P_{FA} &= P(l(\mathbf{y}_k) | H_0 \geq \lambda) \\ &= 1 - \frac{1}{\Gamma(M/2)} \gamma\left(\frac{M}{2}, \frac{\lambda}{2}\right), \end{aligned} \quad (14)$$

where $\Gamma(x)$ is the gamma function and $\gamma(s, t)$ is the lower incomplete gamma function. When $\lambda > M$, the Chernoff bound of false alarm rate can be expressed as

$$P_{FA} \leq \left(\frac{\lambda}{M} \exp\left(1 - \frac{\lambda}{M}\right)\right)^{M/2}. \quad (15)$$

Since the right side of (15) is the Lambert W function, the solution is not easy to obtain, so we use the Taylor series to approximate it, and we can get the detection threshold

$$\begin{aligned} \lambda(P_{FA}) &\approx MP_{FA}^{2/M} + M(1 - P_{FA}^{2/M}) - \frac{3M}{2}(1 - P_{FA}^{2/M}) \\ &\quad + \frac{8M}{3}(1 - P_{FA}^{2/M})^4 - \frac{125M}{24}(1 - P_{FA}^{2/M})^5. \end{aligned} \quad (16)$$

When the target is present, the process of target detection is

$$l(\mathbf{y}_k) | H_1 = \frac{1}{\sigma_n^2} (\mathbf{S}_k \mathbf{g}_k + \mathbf{n}_k)^H (\mathbf{S}_k \mathbf{g}_k + \mathbf{n}_k) \underset{H_0}{\overset{H_1}{\geq}} \lambda(P_{FA}). \quad (17)$$

Since $s_{k,m} g_{k,m} + n_{k,m} \sim \mathcal{N}(0, s_{k,m}^2 \sigma_{Gm}^2 + \sigma_n^2)$, where $s_{k,m}$, $g_{k,m}$, and $n_{k,m}$, respectively, denote the m th entry of \mathbf{s}_k , \mathbf{g}_k , and \mathbf{n}_k , $x \triangleq l(\mathbf{y}_k) | H_1$ meets generalized chi-squared distribution and the probability density function is [32]

$$f(x) = \sum_{m=1}^M \frac{\exp(-x/\xi_m^2)}{\xi_m^2 \prod_{i=1, i \neq m}^M (1 - (\xi_i^2/\xi_m^2))}, \quad (18)$$

where $\xi_m^2 = 1 + (s_{k,m}^2 \sigma_{Gm}^2 / \sigma_n^2)$ and $\xi_m^2 \neq \xi_n^2 (\forall m \neq n)$. According to the Neyman-Pearson lemma, we can attain the probability of GLRT detection based on CFAR

$$\begin{aligned} P_D &= \int_{\lambda(P_{FA})}^{\infty} f(x) dx \\ &= \sum_{m=1}^M \frac{\exp(-\lambda(P_{FA})/\xi_m^2)}{\prod_{i=1, i \neq m}^M (1 - (\xi_i^2/\xi_m^2))}. \end{aligned} \quad (19)$$

3.2. The TSC Estimation Based on MAP. After detecting the presence of target, the TSC can be estimated from the echo waveform. When MAP is adopted as the receiver filter, we have

$$\hat{\mathbf{g}}_k = \arg \max_{\mathbf{g}_k} p(\mathbf{g}_k | \mathbf{y}_k). \quad (20)$$

According to Bayes' theorem, we have

$$p(\mathbf{g}_k | \mathbf{y}_k) = \frac{p(\mathbf{g}_k, \mathbf{y}_k)}{p(\mathbf{y}_k)} = \frac{p(\mathbf{y}_k | \mathbf{g}_k) p(\mathbf{g}_k)}{p(\mathbf{y}_k)}, \quad (21)$$

where $p(\mathbf{y}_k | \mathbf{g}_k)$ and $p(\mathbf{g}_k)$ are both the function of Gaussian distribution and the distribution of \mathbf{y}_k is

$$p(\mathbf{y}_k) = \int p(\mathbf{y}_k | \mathbf{g}_k) p(\mathbf{g}_k) d\mathbf{g}_k. \quad (22)$$

We can obtain the posterior probability by substituting (22) into (21) as

$$\begin{aligned} p(\mathbf{g}_k | \mathbf{y}_k) &= \frac{p(\mathbf{y}_k | \mathbf{g}_k) p(\mathbf{g}_k)}{p(\mathbf{y}_k)} \\ &= \frac{\exp\left[\mathbf{g}_k^H \left(\left(1/\sigma_n^2\right) \mathbf{S}_k^H (\mathbf{y}_k - (1/2) \mathbf{S}_k \mathbf{g}_k) - (1/2) \mathbf{R}_k^{-1} \mathbf{g}_k\right)\right]}{\exp\left(\left(1/2\sigma_n^4\right) \mathbf{y}_k^H \mathbf{S}_k \mathbf{B}_k^{-H} \mathbf{S}_k^H \mathbf{y}_k\right) \sqrt{\det(\pi \mathbf{B}_k^{-1})}}, \end{aligned} \quad (23)$$

where $\mathbf{B}_k \triangleq (1/\sigma_n^2) \mathbf{S}_k^H \mathbf{S}_k + \mathbf{R}_k^{-1}$ and $\mathbf{B}_k^{-H} \triangleq (\mathbf{B}_k^H)^{-1}$. We can get the optimization objective function by substituting (23) into (20) as

$$\hat{\mathbf{g}}_k = \arg \max_{\mathbf{g}_k} \left\{ -\frac{1}{2} \mathbf{g}_k^H \mathbf{B}_k \mathbf{g}_k + \frac{1}{\sigma_n^2} (\mathbf{S}_k \mathbf{g}_k)^H \mathbf{y}_k \right\}. \quad (24)$$

Therefore, the estimated TSC by MAP estimation is

$$\hat{\mathbf{g}}_k = \frac{1}{\sigma_n^2} \mathbf{B}_k^{-1} \mathbf{S}_k^H \mathbf{y}_k. \quad (25)$$

The receiver filter can be written in the matrix form

$$\mathbf{Q}_k \triangleq \frac{1}{\sigma_n^2} \mathbf{B}_k^{-1} \mathbf{S}_k^H. \quad (26)$$

Then the MSE matrix of the estimated TSC is

$$\begin{aligned} \mathbf{P} &= \mathcal{E} \left\{ (\hat{\mathbf{g}}_k - \mathbf{g}_k) (\hat{\mathbf{g}}_k - \mathbf{g}_k)^H \right\} \\ &= \mathbf{Q}_k (\mathbf{S}_k \mathbf{R}_k \mathbf{S}_k^H + \sigma_n^2 \mathbf{I}_M) \mathbf{Q}_k^H \\ &\quad - \frac{1}{\sigma_n^2} \mathbf{B}_k^{-1} \mathbf{S}_k^H \mathbf{S}_k \mathbf{R}_k - \frac{1}{\sigma_n^2} \mathbf{R}_k \mathbf{S}_k^H \mathbf{S}_k \mathbf{B}_k^{-H} + \mathbf{R}_k. \end{aligned} \quad (27)$$

- (1) Set the pulse index $k = 1$, and the initial TSC estimation is based on MAP, where the estimated TSC $\hat{\mathbf{g}}_{k|k}$ can be calculated from (25) and the MSE matrix $\mathbf{P}_{k|k}$ can be obtained from (27). Set the maximum number of pulses K_{\max} .
- (2) Let $k = k + 1$;
- (3) **while** $k \leq K_{\max}$ and $k > 1$ **do**
- (4) According to the temporal correlation of TSC (4), the prediction of TSC is

$$\hat{\mathbf{g}}_{k|k-1} = e^{-T/\tau} \hat{\mathbf{g}}_{k-1|k-1}, \quad (\text{i})$$
 where $\hat{\mathbf{g}}_{k-1|k-1}$ is the estimated TSC during the $(k - 1)$ th pulse;
- (5) The MSE matrix of the predicted TSC is

$$\mathbf{P}_{k|k-1} = e^{-2T/\tau} \mathbf{P}_{k-1|k-1} + (1 - e^{-2T/\tau}) \boldsymbol{\Sigma}_g; \quad (\text{ii})$$
- (6) The KF gain matrix is defined as

$$\boldsymbol{\Phi}_k \triangleq \mathbf{P}_{k|k-1} \mathbf{S}_k^H (\sigma_n^2 \mathbf{Q}_k + \mathbf{Q}_k \mathbf{S}_k \mathbf{P}_{k|k-1} \mathbf{S}_k^H)^{-1}; \quad (\text{iii})$$
- (7) Update the estimated TSC

$$\hat{\mathbf{g}}_{k|k} = \hat{\mathbf{g}}_{k|k-1} + \boldsymbol{\Phi}_k (\hat{\mathbf{g}}_k - \mathbf{Q}_k \mathbf{S}_k \hat{\mathbf{g}}_{k|k-1}), \quad (\text{iv})$$
 where $\hat{\mathbf{g}}_k = \mathbf{Q}_k \mathbf{y}_k$;
- (8) Update the MSE matrix

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \boldsymbol{\Phi}_k \mathbf{Q}_k \mathbf{S}_k \mathbf{P}_{k|k-1}; \quad (\text{v})$$
- (9) Let $k = k + 1$.
- (10) **end while**

ALGORITHM 1: The TSC estimation based on KF.

3.3. *The TSC Estimation Based on KF.* We proposed an approach based on KF to estimate TSC according to the temporal correlation of TSC, when the GLRT detects the presence of target. Furthermore, we optimize the transmitted waveform to further improve the estimation performance by minimizing the trace of MSE matrix. The estimation approach based on KF is described in Algorithm 1.

In order to improve the performance of KF estimation, the trace of the MSE matrix $\mathbf{P}_{k|k}$ is utilized to measure this performance,

$$f(\mathbf{s}_k) = \text{Tr}\{\mathbf{P}_{k|k}\}. \quad (28)$$

The transmitted waveform should be optimized to minimize the MSE at each iteration of KF, which is equivalent to the following optimization problem:

$$\begin{aligned} \mathbf{s}_k^* &= \arg \min_{\mathbf{s}_k} f(\mathbf{s}_k) \\ \text{s.t.} \quad &\text{Tr}\{\mathbf{S}_k^H \mathbf{S}_k\} = E_s \\ &\mathbf{S}_k = \text{diag}\{\mathbf{s}_k\}, \end{aligned} \quad (29)$$

where E_s is the power of radar signal. The objective function can be simplified as

$$\begin{aligned} f(\mathbf{s}_k) &= \text{Tr}\{\mathbf{P}_{k|k}\} \\ &= \sum_{m=1}^M \frac{|s_m|^2 P_{k|k-1,m}^2}{\sigma_n^2 + P_{k|k-1,m} |s_m|^2}, \end{aligned} \quad (30)$$

where $P_{k|k-1,m}$ denotes the entry at m th row and m th column of $\mathbf{P}_{k|k}$. Then we can get the Lagrangian function of this optimization problem

$$g(\mathbf{s}_k, \zeta) = f(\mathbf{s}_k) - \zeta \left(\sum_{m=1}^M |s_m|^2 - E_s \right). \quad (31)$$

Let the derivation of (31) equal 0;

$$\frac{\partial g(\mathbf{s}_k, \zeta)}{\partial |s_m|^2} = 0. \quad (32)$$

Therefore, the optimal spectrum of transmitted waveform should satisfy the following condition:

$$|s_m^*|^2 = \max \left\{ 0, \alpha \sqrt{\sigma_n^2} - \frac{\sigma_n^2}{P_{k|k-1,m}} \right\}, \quad (33)$$

where α is the constraint of transmitted power $\text{Tr}(\mathbf{S}_k^H \mathbf{S}_k) = E_s$ and the optimal signal obtained in (34) is similar to the *water-filling* method in the communication systems [12].

3.4. *The Optimization of EBPSK-Based MCPC Signal.* In the practical radar systems, the waveforms are generated by setting the modulation parameters or choosing from the preset waveforms, so it is difficult to attain the optimal waveform calculated in Section 3.3. Therefore, we propose a suboptimal signal by tuning the parameters of EBPSK-based MCPC waveform in terms of power allocation among subcarriers and coding matrix. In addition, the process of waveform optimization is simple while maintaining the estimation performance of TSC.

The optimal waveform obtained in (33) is referred as \mathbf{s}_k^* ; then the best power allocation scheme for EBPSK-based MCPC signal should approximate the spectrum to this optimal one. Therefore, the MSE approximation can be expressed as an optimization problem

$$\mathbf{s}_k'^* = \arg \max_{\mathbf{s}_k'} \left\| |\mathbf{s}_k'|^2 - |\mathbf{s}_k^*|^2 \right\|_2^2, \quad (34)$$

where \mathbf{s}_k' denotes EBPSK-based MCPC signal in the frequency domain. Because of the sparsity of the EBPSK

spectrum [30], we only need to consider the power allocation in carrier frequencies of subcarriers. The amplitudes should satisfy the following condition:

$$\frac{a_m^*}{a_n^*} = \frac{|s_{k,m}^*|}{|s_{k,n}^*|}, \quad (35)$$

where $s_{k,m}^*$ denotes the m th entry of \mathbf{s}_k^* . The optimized power allocation (35) of EBPSK-based MCPC signal can improve the estimation performance of TSC via KF, and the simulation results will be given in Section 5.

We will also optimize the coding matrix in the following, which can further reduce the MSE of the estimated TSC based on KF. The optimal coding matrix should be the solution of the following optimization problem:

$$\mathbf{C}_k^* = \arg \min_{\mathbf{C}_k} \frac{|\tilde{\mathbf{s}}_k^H \mathbf{x}|}{\|\tilde{\mathbf{s}}_k\|_2 \|\mathbf{x}\|_2} \quad (36)$$

$$\text{s.t. } C_{lr} \in \{0, 1\},$$

where $\tilde{\mathbf{s}}_k$ denotes the discrete form the transmitted waveform $\tilde{S}_k(f) = \sum_{l=1}^L a_{k,l} \sum_{r=1}^R C_{k,lr} S_1(f, f_l) + (1 - C_{k,lr}) S_0(f, f_l) \exp(-j2\pi f T_s r)$ and \mathbf{x} denotes the optimal reference signal, which will be obtained by the estimated TSC and the target scattering model in the following.

When the estimation value of scattering coefficients is $\hat{\mathbf{g}}_{k-1|k-1}$, the predicted value is

$$\hat{\mathbf{g}}_{k|k-1} = e^{-T/\tau} \hat{\mathbf{g}}_{k-1|k-1}. \quad (37)$$

However, the real one is

$$\mathbf{g}_k = e^{-T/\tau} \mathbf{g}_{k-1} + \mathbf{u}_{k-1}. \quad (38)$$

We should design the transmitted signal \mathbf{x} , which minimizes the estimation MSE with the assumption of $\hat{\mathbf{g}}_{k-1} \approx \mathbf{g}_{k-1}$. Since we have proposed the optimal power allocation scheme by the water-filling method, the estimation scattering vector is chosen as the optimal reference waveform, and (37) is the optimization objective function. Let

$$\mathbf{x} = \hat{\mathbf{g}}_{k|k-1} + (1+i) \sqrt{\frac{1}{2} (1 - e^{-2T/\tau}) \boldsymbol{\Sigma}_g}, \quad (39)$$

where i is the imaginary unit. A simple iteration algorithm is proposed to calculate the optimal coding matrix, which is described in Algorithm 2.

4. Simulation Results

4.1. The TSC Estimation Based on MAP. First, we simulate the MSE of estimated TSC based on the MAP receiver filter and compare the simulation results with the theoretical ones. The parameters are set in Table 1, where the transmitted waveform is the EBPSK-based MCPC signal. In this work, all the simulation parameter are given based on the published literatures. For example, the parameters of EBPSK modulation are according to [20, 21, 27–29], and the parameters for

TABLE 1: Simulation parameters.

Parameter	Value
Carrier frequency f_c	100 MHz
Sample frequency f_s	1 GHz
The parameter of EBPSK modulation K	2
The parameter of EBPSK modulation N	20
The number of subcarriers L	16
The length of codes of each subcarrier N_C	32
Frequency interval Δf	20 MHz
The SNR of echo signal	$[-20, 20]$ dB

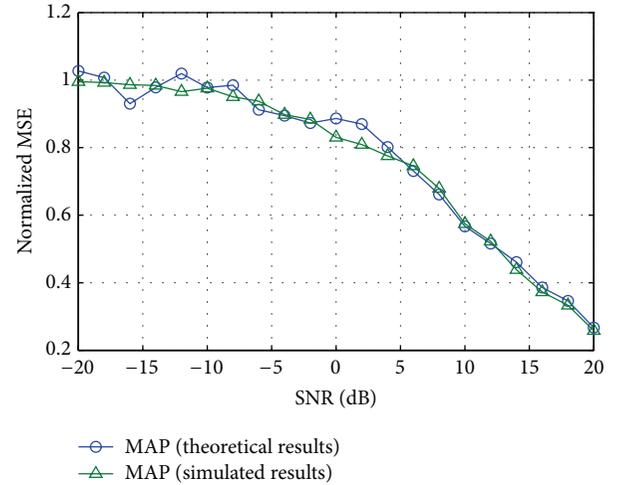


FIGURE 3: The estimation performance based on MAP.

the characteristics of target and radar are based on [5, 14]. Although the realistic data is not adopted in our work, the simulation results are reliable and practicable.

Figure 3 depicts a comparison between theoretical and simulation results of the normalized MSE of estimated TSC, where the theoretical results are calculated by the trace of MSE matrix in (28), and the normalized MSE is defined as

$$e(\hat{\mathbf{g}}_k) \triangleq \frac{\|\hat{\mathbf{g}}_k - \mathbf{g}_k\|_2^2}{\|\mathbf{g}_k\|_2^2}. \quad (40)$$

We can observe from this figure that the simulation and theoretical results have a relatively high degree of agreement, which verifies the efficiency to measure the estimation performance of MAP by the trace of MSE matrix.

4.2. The TSC Estimation Based on KF. This subsection gives the simulation results of the estimated TSC based on KF, in which the simulation parameters are the same with Section 4.1 except SNR = 5 dB, and the index of iteration is from 1 to 50. Figure 4 demonstrates the normalized MSE of estimated TSC, including the approach based on KF and MAP. Apparently, the estimation performance is improved about 25% by using the KF approach instead of MAP, since the temporal correlation is exploited by KF. In addition, the theoretical results given in (28) have high degree of

- (1) Initialize the coding matrix \mathbf{C}^* with a random ± 1 matrix, and set the iteration index $i = 1$ and the maximum number of iterations I_{\max} ;
- (2) **while** $i \leq I_{\max}$ and the coding matrix changes **do**
- (3) Let $r = i \bmod ML$, $C^*([r/M] + 1, r - M[r/M]) = 1 - C^*([r/M] + 1, r - M[r/M])$, where $C^*(m, n)$ denotes the entry at the m th row and n th column of \mathbf{C}^* ;
- (4) Generate the EBPSK-based MCPC waveform $\tilde{\mathbf{s}}_i$ with coding matrix \mathbf{C}^* and calculate
$$z_i = \|\tilde{\mathbf{s}}_i\|^2 - \|\mathbf{x}\|_2^2; \quad (\text{vi})$$
- (5) If $z_i > z_{i-1}$, $i = i + 1$, and go to Step 3, otherwise, let $C^*([r/M] + 1, r - M[r/M]) = 1 - C^*([r/M] + 1, r - M[r/M])$, $i = i + 1$ and go to Step 3;
- (6) Let $i = i + 1$.
- (7) **end while**

ALGORITHM 2: The algorithm to optimize the coding matrix.

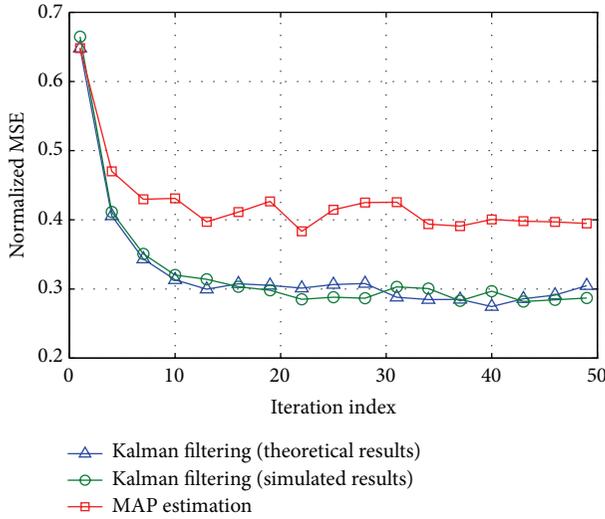


FIGURE 4: The estimation of TSC based on KF.

agreement with the simulation results. Therefore, the trace of MSE matrix can be utilized as an objective function, which accurately describes the estimation performance of KF, in the waveform optimization problem. However, during the simulation, the decreasing normalized MSE of MAP estimation is caused by the exponential correlation model $\mathbf{g}_k = e^{-T/\tau} \mathbf{g}_{k-1} + \mathbf{u}_{k-1}$.

Figure 5 illustrates the spectrum of transmitted waveform, including the power uniform distributed EBPSK-based MCPC signal and the optimal spectrum. The optimal spectrum, which can minimize the trace of MSE matrix and improve the estimation performance of KF approach, is obtained by the water-filling method, where the parameter α is determined by the dichotomy algorithm. In addition, as shown in this figure, the variance of TSC is also given, which is similar to the optimal spectrum. Therefore, more power should be concentrated on the frequencies with relatively large variance of TSC, which provides guidance on the optimization of EBPSK-based MCPC signal.

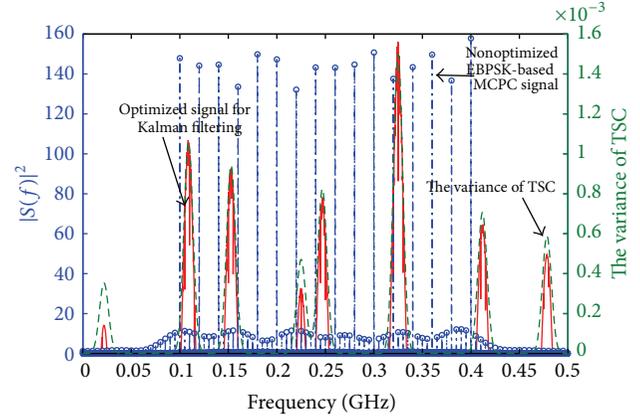


FIGURE 5: The optimized spectrum for the KF estimation.

Figure 6 shows the estimation performance of the methods based on MAP and KF, where the waveform with optimal spectrum and the random one is compared. We can observe that the simulated MSE and theoretical one have a relatively high degree of agreement. Moreover, the accuracy of estimation based on KF is more precise than the one based on MAP. In addition, optimizing the spectrum of transmitted waveform can further improve the estimation performance both by the approaches based on MAP and KF though the criterion of waveform optimization is the MSE matrix of the KF estimation. The performance of MAP estimation is improved by 90%, and the performance of KF estimation is also improved by 90%. Moreover, the KF estimation with optimized waveform is 18% better than the MAP one with optimized waveform.

4.3. The Optimization of EBPSK-Based MCPC Signal.

Figure 7 depicts the spectrum of the optimal power allocation for EBPSK-based MCPC signal, where the KF estimation is adopted. The optimized power allocation shows that more power should be concentrated on the frequencies with relatively large variance of TSC. Therefore, we should design the EBPSK-based MCPC signal to approximate the signal

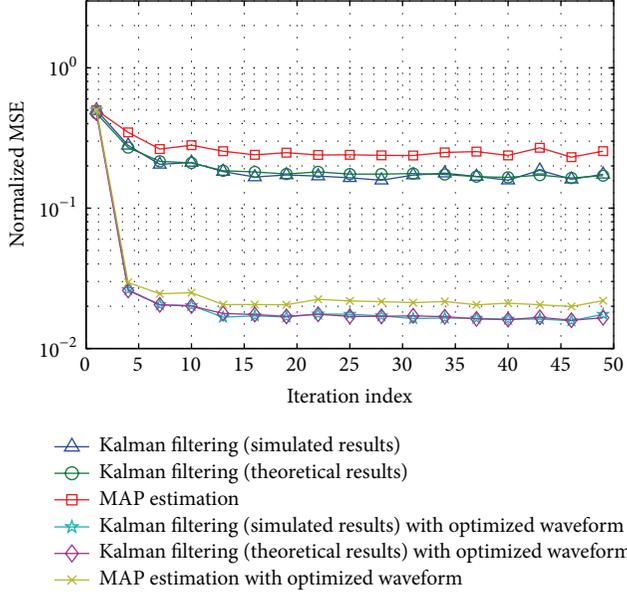


FIGURE 6: The comparison of estimation performance, including (a) the MAP approach with nonoptimized waveform; (b) the MAP approach with optimized waveform; (c) the KF approach with nonoptimized waveform; (d) the KF approach with optimized waveform.

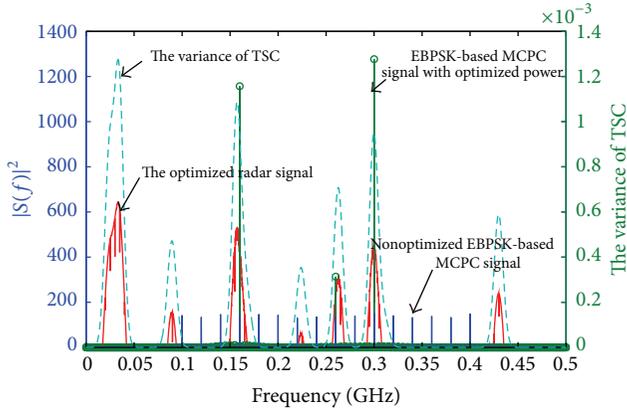


FIGURE 7: The spectrum of EBPSK-based MCPC signal with optimized power allocation.

with optimal spectrum, which can be calculated in (33). Figure 8 compares the normalized MSE of EBPSK-based MCPC signal with optimized power allocation with the one with uniform distributed power. We can observe that the estimation performance in the CRS can be improved significantly by optimizing the transmission power of each subcarrier, where the KF estimation with optimized power allocation of EBPSK-based MCPC signal achieves 7% better than the one with uniform power allocation. Although the estimation performance of the optimized EBPSK-based MCPC signal is worse than the waveform with optimal spectrum (33), we can generate the optimal EBPSK-based MCPC signal by tuning the amplitudes of each subcarrier, which is more simple and convenient.

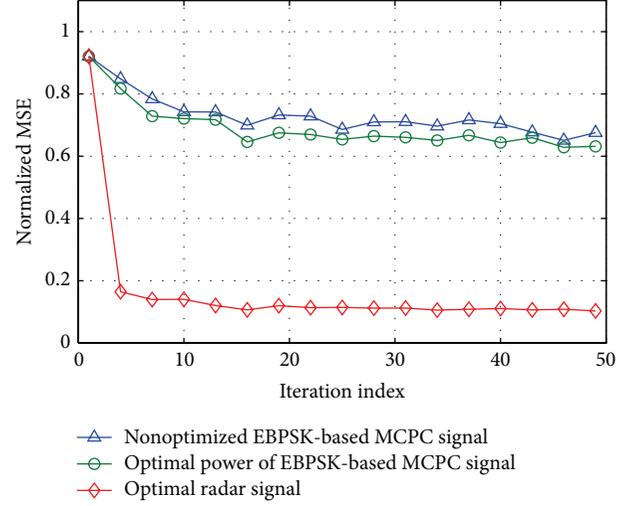


FIGURE 8: The comparison of estimation performance by KF approach, including (a) the EBPSK-based MCPC signal with uniform distributed power; (b) the EBPSK-based MCPC signal with optimized power allocation; (c) the optimal signal.

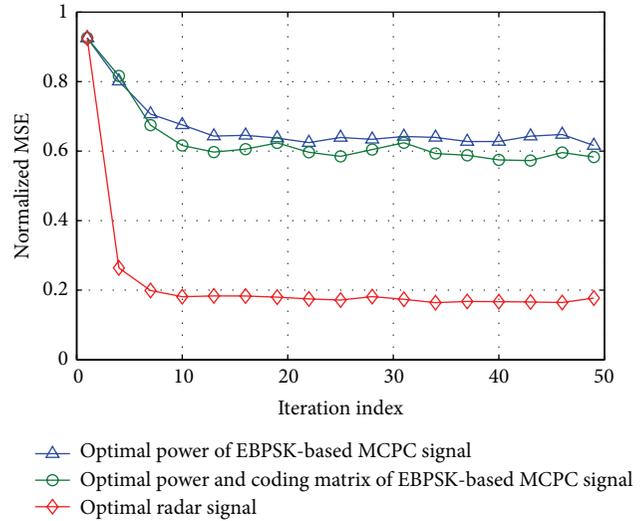


FIGURE 9: Optimize the power and coding matrix of the EBPSK-based MCPC signal. The comparison of estimation performance by KF approach, including (a) the EBPSK-based MCPC signal with optimized power allocation and coding matrix; (b) the EBPSK-based MCPC signal with optimized power allocation; (c) the optimal signal.

Figure 9 compares the estimation performance of two types of optimized EBPSK-based MCPC signal, where the first one optimizes the power allocation and coding matrix simultaneously, and the second one only optimizes the power allocation. As the figure shows, the estimation performance is improved 8% by simultaneously optimizing these two parameters. Furthermore, the algorithm proposed in this work to optimize the coding matrix is easy to realize.

5. Conclusions

In this work, we have investigated the problem of system optimization in the CRS with EBPSK-based MCPC signal, and a new radar working scheme has been proposed to take both the target detection and estimation into consideration. At the detection stage, we have deduced the theoretical threshold and the detection probability based on GLRT. At the estimation stage, the TSC is estimated by the approach based on KF. By exploiting the temporal correlation of TSC, the estimation performance of KF is 25% better than that of MAP. In addition, the transmitted waveform with optimal spectrum is obtained, which can improve the estimation performance of KF by 90%. For the practical consideration, the iteration algorithms are proposed to optimize the EBPSK-based MCPC signal in terms of power allocation and coding matrix, which can achieve 7% and 8% performance improvement, respectively. In the future work, we will put our force on the system optimization of the temporal correlated CRS in the presence of clutter and jam.

Conflict of Interests

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

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (Grant no. 61271204) and the National Key Technology R&D Program (Grant no. 2012BAH15B00).

References

- [1] S. Haykin, "Cognitive radar: A way of the future," *IEEE Signal Processing Magazine*, vol. 23, no. 1, pp. 30–40, 2006.
- [2] S. Haykin, Y. Xue, and T. N. Davidson, "Optimal waveform design for cognitive radar," in *Proceedings of the 42nd Asilomar Conference on Signals, Systems and Computers (ASILOMAR '08)*, pp. 3–7, Pacific Grove, Calif, USA, October 2008.
- [3] W. Huleihel, J. Tabrikian, and R. Shavit, "Optimal sequential waveform design for cognitive radar," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '12)*, pp. 2457–2460, Kyoto, Japan, March 2012.
- [4] Y. Zhang, Z. Dong, L. Wu, and S. Wang, "A hybrid method for mri brain image classification," *Expert Systems with Applications*, vol. 38, no. 8, pp. 10049–10053, 2001.
- [5] X. Zhang and C. Cui, "Range-spread target detecting for cognitive radar based on track-before-detect," *International Journal of Electronics*, vol. 101, no. 1, pp. 74–87, 2014.
- [6] P. Setlur, N. Devroye, and Z. Cheng, "Waveform scheduling via directed information in cognitive radar," in *Proceedings of the IEEE Statistical Signal Processing Workshop (SSP '12)*, pp. 864–867, Ann Arbor, Mich, USA, August 2012.
- [7] Y.-D. Zhang, L.-N. Wu, and G. Wei, "A new classifier for polarimetric SAR images," *Progress in Electromagnetics Research*, vol. 94, pp. 83–104, 2009.
- [8] Y. Zhang and L. Wu, "Crop classification by forward neural network with adaptive chaotic particle swarm optimization," *Sensors*, vol. 11, no. 5, pp. 4721–4743, 2011.
- [9] Y. Yu, A. P. Petropulu, and H. V. Poor, "Measurement matrix design for compressive sensing-based MIMO radar," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5338–5352, 2011.
- [10] A. Leshem, O. Napporstek, and A. Nehorai, "Information theoretic adaptive radar waveform design for multiple extended targets," *IEEE Journal on Selected Topics in Signal Processing*, vol. 1, no. 1, pp. 42–55, 2007.
- [11] S. P. Jacobs and J. A. O'Sullivan, "Automatic target recognition using sequences of high resolution radar range-profiles," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 36, no. 2, pp. 364–381, 2000.
- [12] M. R. Bell, "Information theory and radar waveform design," *IEEE Transactions on Information Theory*, vol. 39, no. 5, pp. 1578–1597, 1993.
- [13] R. A. Romero, J. Bae, and N. A. Goodman, "Theory and application of SNR and mutual information matched illumination waveforms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 2, pp. 912–927, 2011.
- [14] F. Z. Dai, H. W. Liu, P. H. Wang, and S. Z. Xia, "Adaptive waveform design for range-spread target tracking," *Electronics Letters*, vol. 46, no. 11, pp. 793–794, 2010.
- [15] B. Jiu, H. Liu, B. Chen, and Z. Liu, "Waveform design for wideband radar target recognition based on eigensubspace projection," *IET Radar, Sonar and Navigation*, vol. 7, no. 6, pp. 702–709, 2013.
- [16] 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.
- [17] X. Deng, C. Qiu, Z. Cao, M. Morelande, and B. Moran, "Waveform design for enhanced detection of extended target in signal-dependent interference," *IET Radar, Sonar and Navigation*, vol. 6, no. 1, pp. 30–38, 2012.
- [18] S. Kay, "Optimal signal design for detection of Gaussian point targets in stationary Gaussian clutter/reverberation," *IEEE Journal on Selected Topics in Signal Processing*, vol. 1, no. 1, pp. 31–41, 2007.
- [19] Y. Zhang, B. Peterson, G. Ji, and Z. Dong, "Energy preserved sampling for compressed sensing MRI," *Computational and Mathematical Methods in Medicine*, vol. 2014, Article ID 546814, 12 pages, 2014.
- [20] R. A. Romero and N. A. Goodman, "Waveform design in signal-dependent interference and application to target recognition with multiple transmissions," *IET Radar, Sonar and Navigation*, vol. 3, no. 4, pp. 328–340, 2009.
- [21] N. Levanon, "Multifrequency radar signals," in *Proceedings of the IEEE International Radar Conference*, pp. 683–688, Alexandria, Va, USA, May 2000.
- [22] N. Levanon and E. Mozeson, "Multicarrier radar signal—Pulse train and CW," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 2, pp. 707–720, 2002.
- [23] N. Levanon, "Multifrequency complementary phase-coded radar signal," *IEEE Proceedings: Radar, Sonar and Navigation*, vol. 147, no. 6, pp. 276–284, 2000.
- [24] S. Sen and C. W. Glover, "Optimal multicarrier phase-coded waveform design for detection of extended targets," in *Proceedings of the IEEE Radar Conference*, pp. 1–6, Ottawa, Canada, April-May 2013.

- [25] P. Chen and L. Wu, "Optimum weighted cumulation target detection for resonance region multi-carrier radar," in *Proceedings of the International Conference on Wireless Communications and Signal Processing (WCSP '13)*, pp. 1–4, Hangzhou, China, October 2013.
- [26] E. Mozeson and N. Levanon, "Multicarrier radar signals with low peak-to-mean envelope power ratio," *IEE Proceedings: Radar, Sonar and Navigation*, vol. 150, no. 2, pp. 71–77, 2003.
- [27] L. Wu, "The progress of the UNB speed communications," *Progress in Natural Science*, pp. 1467–1473, 2007.
- [28] X. Chen and L. Wu, "A novel detection scheme for EBPSK system," *Mathematical Problems in Engineering*, vol. 2012, Article ID 956191, 14 pages, 2012.
- [29] Y. Jin, L.-N. Wu, P.-K. Ying, and J. Yu, "RS coded EBPSK-MODEM based on SVM discrimination," *Journal of Applied Sciences*, vol. 31, no. 1, pp. 48–52, 2013.
- [30] P. Chen and L. Wu, "Compressive sensing for multi-carrier EBPSK radar," *Journal of Southeast University (Natural Science Edition)*, vol. 44, no. 1, pp. 23–27, 2014.
- [31] M. Feng, L. Wu, J. Ding, and C. Qi, "BER analysis and verification of EBPSK system in AWGN channel," *IEICE Transactions on Communications*, vol. 94, no. 3, pp. 806–809, 2011.
- [32] D. Hammarwall, M. Bengtsson, and B. Ottersten, "Acquiring partial CSI for spatially selective transmission by instantaneous channel norm feedback," *IEEE Transactions on Signal Processing*, vol. 56, no. 3, pp. 1188–1204, 2008.

Research Article

A New Asymptotic Notation: Weak Theta

Andrei-Horia Mogoş,¹ Bianca Mogoş,² and Adina Magda Florea¹

¹Faculty of Automatic Control and Computers, University Politehnica of Bucharest, Romania

²Faculty of Mathematics and Computer Science, University of Bucharest, Romania

Correspondence should be addressed to Andrei-Horia Mogoş; andrei.mogosh@cs.pub.ro

Received 30 July 2014; Accepted 3 October 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Andrei-Horia Mogoş 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.

Algorithms represent one of the fundamental issues in computer science, while asymptotic notations are widely accepted as the main tool for estimating the complexity of algorithms. Over the years a certain number of asymptotic notations have been proposed. Each of these notations is based on the comparison of various complexity functions with a given complexity function. In this paper, we define a new asymptotic notation, called “Weak Theta,” that uses the comparison of various complexity functions with two given complexity functions. Weak Theta notation is especially useful in characterizing complexity functions whose behaviour is hard to be approximated using a single complexity function. In addition, in order to highlight the main particularities of Weak Theta, we propose and prove several theoretical results: properties of Weak Theta, criteria for comparing two complexity functions, and properties of a new set of complexity functions (also defined in the paper) based on Weak Theta. Furthermore, to illustrate the usefulness of our notation, we discuss an application of Weak Theta in artificial intelligence.

1. Introduction

Computational complexity [1–6] is a very popular research area in computer science that covers algorithm design [3, 4], algorithm complexity [3, 4], asymptotic notations [3, 7–9], complexity recurrences [3, 10, 11], classes of problems and NP-completeness [12, 13], and heuristics and approximation algorithms [13, 14].

Tools from computational complexity are intensively used in various research areas such as computer networks [15], operating systems [16], parallel and distributed computing [17], and artificial intelligence [18, 19]. Artificial intelligence is particularly linked with computational complexity; see swarm intelligence [20–25], multiagent systems [26–28], machine learning [29, 30], and semantic web services [31, 32].

Algorithms represent one of the most important issues in computer science, while asymptotic notations are considered to be the main tool for estimating the complexity of algorithms. Several asymptotic notations have been proposed in the literature. They are all based on the comparison of various complexity functions with a given complexity function. Consequently, the main idea is to choose a complexity function and then to see what is the relation between this

function and other complexity functions. Some of these asymptotic notations provide relations that are too general, while others offer relations that are too specific and therefore not applicable for all the cases.

In this paper, we propose a new asymptotic notation that aims to offer a solution for the problem discussed above. It proposes the use of two given complexity functions to construct the set of all complexity functions that can be caught between these two given functions. The main advantage of our asymptotic notation is the fact that it can be used to describe the behaviour of a wide set of complexity functions (some of them hard to be approximated using a single complexity function). In this paper, we also propose and prove several properties that characterize this new asymptotic notation and outline its importance in the field of computational complexity.

The paper is organized as follows. Section 2 contains an overview on the asymptotic notations proposed in the literature. In Section 3, we define an asymptotic notation, called “Weak Theta,” we provide a motivation for the need of this notation, and we discuss an application of “Weak Theta” in artificial intelligence. In the end of this section, we also present a brief description of the main symbols used in

the paper. In Section 4, we propose several relations between “Weak Theta” and other asymptotic notations. In Section 5, we develop some properties of “Weak Theta” related to membership, inclusion, intersection, and union. In Section 6, we discuss several criteria concerning the comparison of two complexity functions. In Section 7, we define the set of complexity functions comparable with two given complexity functions and we propose several properties of this new set of functions. Section 8 contains the conclusions of the paper.

2. Related Work

In this section, we present several asymptotic notations proposed in the literature.

Definition 1 (see, e.g., [3, 6, 7]). A complexity function is a function $f : \mathbb{N}^* \rightarrow \mathbb{R}_+^*$, where \mathbb{N}^* is the set of positive integers and \mathbb{R}_+^* is the set of positive real numbers. One denotes by \mathcal{F} the set of complexity functions, $\mathcal{F} = \{f : \mathbb{N}^* \rightarrow \mathbb{R}_+^*\}$.

Remark 2. In this paper, a function $f \in \mathcal{F}$ will be denoted by $f(n)$ whenever this function is in relation with a complexity class that depends on a complexity function of the argument n (e.g., $f \in \mathcal{F}$, but $f(n) \in \Theta(f(n))$).

Definition 3 (see, e.g., [3, 4, 6, 7, 33–36]). Let $g \in \mathcal{F}$, an arbitrary fixed complexity function. The main asymptotic notations used in the literature are defined as follows:

$$\begin{aligned} \Theta(g(n)) &= \{f \in \mathcal{F} \mid \exists c_1, c_2 \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ &\quad \text{such that } c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n), \\ &\quad \forall n \geq n_0\}, \\ O(g(n)) &= \{f \in \mathcal{F} \mid \exists c \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ &\quad \text{such that } f(n) \leq c \cdot g(n), \\ &\quad \forall n \geq n_0\}, \\ \Omega(g(n)) &= \{f \in \mathcal{F} \mid \exists c \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ &\quad \text{such that } c \cdot g(n) \leq f(n), \\ &\quad \forall n \geq n_0\}, \\ o(g(n)) &= \{f \in \mathcal{F} \mid \forall c \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ &\quad \text{such that } f(n) < c \cdot g(n), \\ &\quad \forall n \geq n_0\}, \\ \omega(g(n)) &= \{f \in \mathcal{F} \mid \forall c \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ &\quad \text{such that } c \cdot g(n) < f(n), \\ &\quad \forall n \geq n_0\}. \end{aligned} \quad (1)$$

Proposition 4 (see, e.g., [3, 6]). *One has the following.*

- (a) *Reflexivity.* Let $f \in \mathcal{F}$. Then $f(n) \in \Theta(f(n))$, $f(n) \in O(f(n))$, and $f(n) \in \Omega(f(n))$.

- (b) *Transitivity.* Let $f, g, h \in \mathcal{F}$. Then,
- (b1) if $f(n) \in \Theta(g(n))$ and $g(n) \in \Theta(h(n))$ then $f(n) \in \Theta(h(n))$;
 - (b2) if $f(n) \in O(g(n))$ and $g(n) \in O(h(n))$ then $f(n) \in O(h(n))$;
 - (b3) if $f(n) \in \Omega(g(n))$ and $g(n) \in \Omega(h(n))$ then $f(n) \in \Omega(h(n))$;
 - (b4) if $f(n) \in o(g(n))$ and $g(n) \in o(h(n))$ then $f(n) \in o(h(n))$;
 - (b5) if $f(n) \in \omega(g(n))$ and $g(n) \in \omega(h(n))$ then $f(n) \in \omega(h(n))$.
- (c) *Symmetry.* Let $f, g \in \mathcal{F}$. If $f(n) \in \Theta(g(n))$ then $g(n) \in \Theta(f(n))$.
- (d) *Transpose Symmetry.* Let $f, g \in \mathcal{F}$. Then,
- (d1) $f(n) \in O(g(n))$ if and only if $g(n) \in \Omega(f(n))$;
 - (d2) $f(n) \in o(g(n))$ if and only if $g(n) \in \omega(f(n))$.
- (e) *Projection.* Let $f, g \in \mathcal{F}$. Then $f(n) \in \Theta(g(n))$ if and only if $f(n) \in O(g(n))$ and $f(n) \in \Omega(g(n))$.

Definition 5 (see [7]). Let $g \in \mathcal{F}$. The set of all complexity functions comparable with $g(n)$ is defined as follows:

$$\begin{aligned} C(g(n)) &= \Theta(g(n)) \cup O(g(n)) \cup \Omega(g(n)) \\ &\quad \cup o(g(n)) \cup \omega(g(n)). \end{aligned} \quad (2)$$

Definition 6 (see [7]). Let $g \in \mathcal{F}$. The asymptotic notations $o\Theta(g(n))$ and $\Theta\omega$ are defined as follows:

$$\begin{aligned} o\Theta(g(n)) &= O(g(n)) \setminus (o(g(n)) \cup \Theta(g(n))), \\ \Theta\omega(g(n)) &= \Omega(g(n)) \setminus (\Theta(g(n)) \cup \omega(g(n))). \end{aligned} \quad (3)$$

Proposition 7 (see [7]). *Let $g \in \mathcal{F}$. Then*

- (a) $O(g(n)) = o(g(n)) \cup o\Theta(g(n)) \cup \Theta(g(n))$;
- (b) $\Omega(g(n)) = \Theta(g(n)) \cup \Theta\omega(g(n)) \cup \omega(g(n))$.

Proposition 8 (see [7]). *Let $g \in \mathcal{F}$. Then,*

- (a) $C(g(n)) = o(g(n)) \cup o\Theta(g(n)) \cup \Theta(g(n)) \cup \Theta\omega(g(n)) \cup \omega(g(n))$;
- (b) $o(g(n))$, $o\Theta(g(n))$, $\Theta(g(n))$, $\Theta\omega(g(n))$, and $\omega(g(n))$ are pairwise disjoint;
- (c) $C(g(n)) = O(g(n)) \cup \Omega(g(n))$.

Other important asymptotic notations used in the literature are the following: soft O [3, 8], soft Θ [36, 37], Ω infinity [3, 9], \sim [35, 38], and almost [35].

3. Weak Theta

In this section, we propose a new asymptotic notation called “Weak Theta,” we present some motivation for using this notation, and we discuss an application of “Weak Theta” in artificial intelligence. In the end of the section we briefly describe the main symbols used in the paper.

3.1. Definition of Weak Theta

Definition 9. Let $g_1, g_2 \in \mathcal{F}$. One defines the asymptotic notation Weak Theta, denoted by $\overline{\Theta}$, as follows:

$$\begin{aligned} \overline{\Theta}(g_1(n), g_2(n)) &= \{f \in \mathcal{F} \mid \exists c_1, c_2 \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ &\text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_2(n), \\ &\forall n \geq n_0\}. \end{aligned} \quad (4)$$

Remark 10. $\overline{\Theta}(g_1(n), g_2(n))$ represents the set of all complexity functions that are bounded by the functions $c_1 \cdot g_1(n)$ and $c_2 \cdot g_2(n)$ for sufficiently large n .

Proposition 11. Let $g_1, g_2 \in \mathcal{F}$. Then

- (a) $\overline{\Theta}(g_1(n), g_2(n)) \neq \emptyset$ if and only if $g_1(n) \in O(g_2(n))$;
- (b) $\overline{\Theta}(g_1(n), g_2(n)) \neq \emptyset$ if and only if $g_2(n) \in \Omega(g_1(n))$.

Proof. (a) “ \Rightarrow ” Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n))$. Then,

$$\begin{aligned} \exists c_1, c_2 \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ \text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_2(n), \quad \forall n \geq n_0. \end{aligned} \quad (5)$$

It follows that

$$\begin{aligned} \exists c' = \frac{c_2}{c_1} \in \mathbb{R}_+^*, \exists n'_0 = n_0 \in \mathbb{N}^* \\ \text{such that } g_1(n) \leq c' \cdot g_2(n), \quad \forall n \geq n'_0. \end{aligned} \quad (6)$$

Consequently,

$$g_1(n) \in O(g_2(n)). \quad (7)$$

“ \Leftarrow ” Consider that $g_1(n) \in O(g_2(n))$; it follows that

$$\begin{aligned} \exists c \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ \text{such that } g_1(n) \leq c \cdot g_2(n), \quad \forall n \geq n_0. \end{aligned} \quad (8)$$

From (8), we obtain

$$\begin{aligned} \exists c \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ \text{such that } 1 \cdot g_1(n) \leq g_1(n) \leq c \cdot g_2(n), \quad \forall n \geq n_0. \end{aligned} \quad (9)$$

So, we have

$$\begin{aligned} \exists c'_1 = 1, c'_2 = c \in \mathbb{R}_+^*, \exists n'_0 = n_0 \in \mathbb{N}^* \\ \text{such that } c'_1 \cdot g_1(n) \leq g_1(n) \leq c'_2 \cdot g_2(n), \quad \forall n \geq n'_0. \end{aligned} \quad (10)$$

Consequently,

$$g_1(n) \in \overline{\Theta}(g_1(n), g_2(n)). \quad (11)$$

(b) The result follows from (a) using the relation

$$g_1(n) \in O(g_2(n)) \quad \text{iff} \quad g_2(n) \in \Omega(g_1(n)). \quad (12)$$

□

Remark 12. In all situations in which a proposition or a theorem has as a hypothesis a relation of the form $g_1(n) \in O(g_2(n))$, one implicitly understands that $\overline{\Theta}(g_1(n), g_2(n)) \neq \emptyset$.

3.2. Motivation. (a) The first use of our notation is for describing complexity functions with complex behaviour (i.e., functions difficult to be approximated by a single elementary complexity function). As an example, let $f \in \mathcal{F}$ with the form

$$f(n) = \begin{cases} n^2, & n = 3k, \\ n \lg n, & n = 3k + 1, \\ n, & n = 3k + 2. \end{cases} \quad (13)$$

If we use the asymptotic notations from Definition 3, we may obtain

$$f(n) \in O(n^2) \quad (14)$$

or

$$f(n) \in \Omega(n) \quad (15)$$

which are too general descriptions of the behavior of $f(n)$, while the use of Θ is not appropriate.

If we use the new asymptotic notation proposed in Definition 9, then we obtain

$$f(n) \in \overline{\Theta}(n, n^2). \quad (16)$$

(b) Usually, the best case, the worst case, and the average case of the running time of an algorithm are described by functions with different behaviours. Also, for all algorithms, the average case is situated between the best case and the worst case. For example, assume that for an algorithm A the best case is represented by $f_1(n) = n^2 \lg n + n$ and the worst case is represented by $f_2(n) = n^3 + n^2$.

Using Θ , we obtain that $f_1(n) \in \Theta(n^2 \lg n)$ and $f_2(n) \in \Theta(n^3)$. This is a valid description of the two cases of the algorithm, but we need two complexity classes ($\Theta(n^2 \lg n)$ and $\Theta(n^3)$) for this analysis.

If we use $\overline{\Theta}$, only one complexity class is necessary. The running time of the algorithm A is fully described by $\overline{\Theta}(n^2 \lg n, n^3)$.

3.3. An Application of Weak Theta in Artificial Intelligence.

In this subsection, we propose an artificial intelligence based solution for the following problem denoted by P_1 . “Let $SP = \{p_1, p_2, \dots, p_k\}$ be a set of software programs and P_2 a given problem. Each program p_i , $i = \overline{1, k}$, solves the problem P_2 . We make the following assumptions: (1) each program p_i , $i = \overline{1, k}$, terminates for each valid input; (2) for all programs p_i , $i = \overline{1, k}$, the inputs have the same form; (3) the codes of the programs are not available. One asks to find the fastest program(s) from the set SP .”

The problem P_1 can be seen as an unsupervised classification problem that can be solved in terms of “Weak Theta.” In this framework, we propose an algorithm that aims to solve the problem P_1 . This algorithm uses for the first two steps a simple and common idea for estimating the running time of a software program: “choose several inputs for the program; for each input measure the corresponding running time and then, using this information, find the function that best estimates the running time of the program.”

The algorithm can be described as follows.

Step 1. Consider a set of inputs $\{\text{input}_1, \text{input}_2, \dots, \text{input}_q\}$ of sizes $\{s_1, s_2, \dots, s_q\}$ (where s_j is the size of input input_j). We run each program p_i , $i = \overline{1, k}$, for all inputs and we obtain the corresponding running times $\{t_1^i, t_2^i, \dots, t_q^i\}$. In the end, for each program p_i , $i = \overline{1, k}$, we will have a set of pairs $S^i = \{(s_1, t_1^i), (s_2, t_2^i), \dots, (s_q, t_q^i)\}$.

Step 2. For each program p_i , $i = \overline{1, k}$, we use the data S^i for obtaining a complexity function g^i that best approximates the running time of the program. For this step we may use estimation techniques.

Step 3. First, we choose a set of complexity functions $\{f_1, f_2, \dots, f_r\}$ such that $f_j(n) \in O(f_{j+1}(n))$, for all $j = \overline{1, r-1}$; from Proposition 11, we have $\overline{\Theta}(f_j(n), f_{j+1}(n)) \neq \emptyset$, for all $j = \overline{1, r-1}$. Next, for each program p_i , $i = \overline{1, k}$, we find the complexity class $C_j(n) = \overline{\Theta}(f_j(n), f_{j+1}(n))$ such that $g^i(n) \in C_j(n)$. Sometimes, the function $g^i(n)$ can be simultaneous in two consecutive complexity classes $C_j(n)$ and $C_{j+1}(n)$ (according to Proposition 20, $C_j(n) \cap C_{j+1}(n) = \Theta(f_{j+1}(n))$); in such cases we consider that $g^i(n)$ is only in $C_j(n)$.

As a result of the method proposed above, we obtain a classification of the programs from the set SP that offers a solution for the problem P_1 . This classification is based on the complexity classes $\overline{\Theta}(f_1(n), f_2(n)), \overline{\Theta}(f_2(n), f_3(n)), \dots, \overline{\Theta}(f_{r-1}(n), f_r(n))$.

3.4. Main Symbols Used in the Paper. This subsection contains a brief description of the main symbols used in the paper. In Notations section, one can observe three types of symbols: sets, asymptotic notations, and functions and numbers.

4. Relations between Weak Theta and Other Asymptotic Notations

Proposition 13. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

$$\overline{\Theta}(g_1(n), g_2(n)) = \Omega(g_1(n)) \cap O(g_2(n)). \quad (17)$$

Proof. First, we prove the direct inclusion

$$\overline{\Theta}(g_1(n), g_2(n)) \subseteq \Omega(g_1(n)) \cap O(g_2(n)). \quad (18)$$

Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n))$. It follows that

$$\begin{aligned} \exists c_1, c_2 \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ \text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_2(n), \quad \forall n \geq n_0. \end{aligned} \quad (19)$$

From (19), we have that

$$\begin{aligned} \exists c' = c_1 \in \mathbb{R}_+^*, \exists n'_0 = n_0 \in \mathbb{N}^* \\ \text{such that } c' \cdot g_1(n) \leq f(n), \quad \forall n \geq n'_0, \\ \exists c'' = c_2 \in \mathbb{R}_+^*, \exists n''_0 = n_0 \in \mathbb{N}^* \\ \text{such that } f(n) \leq c'' \cdot g_2(n), \quad \forall n \geq n''_0. \end{aligned} \quad (20)$$

So, we have

$$f(n) \in \Omega(g_1(n)) \cap O(g_2(n)). \quad (21)$$

Next, we prove the reverse inclusion

$$\Omega(g_1(n)) \cap O(g_2(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n)). \quad (22)$$

Let $f(n) \in \Omega(g_1(n)) \cap O(g_2(n))$. It follows that

$$\begin{aligned} \exists c' \in \mathbb{R}_+^*, \exists n'_0 \in \mathbb{N}^* \\ \text{such that } c' \cdot g_1(n) \leq f(n), \quad \forall n \geq n'_0, \\ \exists c'' \in \mathbb{R}_+^*, \exists n''_0 \in \mathbb{N}^* \\ \text{such that } f(n) \leq c'' \cdot g_2(n), \quad \forall n \geq n''_0. \end{aligned} \quad (23)$$

From (23) we have that

$$\begin{aligned} \exists c_1 = c', c_2 = c'' \in \mathbb{R}_+^*, \\ \exists n_0 = \max(n'_0, n''_0) \in \mathbb{N}^* \end{aligned} \quad (24)$$

$$\text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_2(n), \quad \forall n \geq n_0.$$

Consequently,

$$f(n) \in \overline{\Theta}(g_1(n), g_2(n)). \quad (25)$$

□

Corollary 14. Let $g \in \mathcal{F}$. Then $\overline{\Theta}(g(n), g(n)) = \Theta(g(n))$.

Proof. From Proposition 13, we have that

$$\overline{\Theta}(g(n), g(n)) = \Omega(g(n)) \cap O(g(n)). \quad (26)$$

Next, using the relation

$$\Omega(g(n)) \cap O(g(n)) = \Theta(g(n)), \quad (27)$$

we obtain the result. □

Proposition 15. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

- (a) $\Theta(g_1(n)) = \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap O(g_1(n));$
 (b) $\Theta(g_2(n)) = \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \Omega(g_2(n)).$

Proof. (a) First, we prove that $\Theta(g_1(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap O(g_1(n))$. Let $f(n) \in \Theta(g_1(n))$. It follows that

$$\begin{aligned} \exists c_1, c_2 \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ \text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_1(n), \quad \forall n \geq n_0. \end{aligned} \quad (28)$$

From the hypothesis, we have that $g_1(n) \in O(g_2(n))$. So we have

$$\begin{aligned} \exists c' \in \mathbb{R}_+^*, \exists n'_0 \in \mathbb{N}^* \\ \text{such that } g_1(n) \leq c' \cdot g_2(n), \quad \forall n \geq n'_0. \end{aligned} \quad (29)$$

From (28) and (29), we obtain

$$\begin{aligned} \exists c_1'' = c_1, \quad c_2'' = c_2 \cdot c' \in \mathbb{R}_+^*, \\ \exists n_0'' = \max(n_0, n'_0) \in \mathbb{N}^* \\ \text{such that } c_1'' \cdot g_1(n) \leq f(n) \leq c_2'' \cdot g_2(n), \quad \forall n \geq n_0''. \end{aligned} \quad (30)$$

Consequently,

$$f(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n)). \quad (31)$$

Using (27), we have that

$$\Theta(g_1(n)) \subseteq O(g_1(n)). \quad (32)$$

From (31) and (32), we obtain the result.

We prove the reverse inclusion $\overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap O(g_1(n)) \subseteq \Theta(g_1(n))$. Let $f(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap O(g_1(n))$. It follows that

$$\begin{aligned} \exists c_1, c_2 \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ \text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_2(n), \quad \forall n \geq n_0, \end{aligned} \quad (33)$$

$$\exists c' \in \mathbb{R}_+^*, \exists n'_0 \in \mathbb{N}^*$$

$$\text{such that } f(n) \leq c' \cdot g_1(n), \quad \forall n \geq n'_0.$$

From (33), we obtain

$$\begin{aligned} \exists c_1'' = c_1, \quad c_2'' = c' \in \mathbb{R}_+^*, \\ \exists n_0'' = \max(n_0, n'_0) \in \mathbb{N}^* \\ \text{such that } c_1'' \cdot g_1(n) \leq f(n) \leq c_2'' \cdot g_1(n), \quad \forall n \geq n_0''. \end{aligned} \quad (34)$$

Consequently,

$$f(n) \in \Theta(g_1(n)). \quad (35)$$

(b) For the proof, one can follow the same idea used for proving (a) and the definition of Θ . \square

Corollary 16. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

- (a) $\Theta(g_1(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n));$
 (b) $\Theta(g_2(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)).$

Proof. The proof follows easily using Proposition 15. \square

Proposition 17. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then $\overline{\overline{\Theta}}(g_1(n), g_2(n)) = \Theta(g_1(n)) = \Theta(g_2(n))$ if and only if $g_1(n) \in \Theta(g_2(n))$.

Proof. “ \Rightarrow ” From the hypothesis, we have

$$g_1(n) \in \Theta(g_1(n)) = \overline{\overline{\Theta}}(g_1(n), g_2(n)) = \Theta(g_2(n)). \quad (36)$$

“ \Leftarrow ” We consider that $g_1(n) \in \Theta(g_2(n))$. It follows that

$$\begin{aligned} \exists c_1', c_2' \in \mathbb{R}_+^*, \exists n'_0 \in \mathbb{N}^* \\ \text{such that } c_1' \cdot g_2(n) \leq g_1(n) \leq c_2' \cdot g_2(n), \quad \forall n \geq n'_0. \end{aligned} \quad (37)$$

From Corollary 16 and the relation $g_1(n) \in \Theta(g_2(n))$, we have that

$$\Theta(g_1(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)), \quad (38)$$

$$\Theta(g_2(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)).$$

We have to prove that

$$\overline{\overline{\Theta}}(g_1(n), g_2(n)) \subseteq \Theta(g_1(n)), \quad (39)$$

$$\overline{\overline{\Theta}}(g_1(n), g_2(n)) \subseteq \Theta(g_2(n)). \quad (40)$$

Let $f(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n))$. It follows that

$$\begin{aligned} \exists c_1'', c_2'' \in \mathbb{R}_+^*, \exists n_0'' \in \mathbb{N}^* \\ \text{such that } c_1'' \cdot g_1(n) \leq f(n) \leq c_2'' \cdot g_2(n), \quad \forall n \geq n_0''. \end{aligned} \quad (41)$$

Using (37), we have that

$$\exists c_1''' = \frac{1}{c_1'} \in \mathbb{R}_+^*, \exists n_0''' = n'_0 \in \mathbb{N}^* \quad (42)$$

$$\text{such that } g_2(n) \leq c_1''' \cdot g_1(n), \quad \forall n \geq n_0'''.$$

From (41) and (42), we obtain that

$$\begin{aligned} \exists c_1 = c_1'', \quad c_2 = c_2'' \cdot c_1''' \in \mathbb{R}_+^*, \\ \exists n_0 = \max(n_0'', n_0''') \in \mathbb{N}^* \end{aligned} \quad (43)$$

$$\text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_1(n), \quad \forall n \geq n_0.$$

Consequently,

$$f(n) \in \Theta(g_1(n)). \quad (44)$$

The other inclusion, from (40), can be proved using the same idea. \square

Proposition 18. Let $g_1, g_2, g_3, g_4 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$ and $g_3(n) \in O(g_4(n))$. Then $\overline{\Theta}(g_1(n), g_2(n)) = \overline{\Theta}(g_3(n), g_4(n))$ if and only if $g_1(n) \in \Theta(g_3(n))$ and $g_2(n) \in \Theta(g_4(n))$.

Proof. “ \Rightarrow ” From the hypothesis and Corollary 16, we have that

$$\begin{aligned} g_1(n) &\in \overline{\Theta}(g_1(n), g_2(n)) = \overline{\Theta}(g_3(n), g_4(n)), \\ g_2(n) &\in \overline{\Theta}(g_1(n), g_2(n)) = \overline{\Theta}(g_3(n), g_4(n)). \end{aligned} \quad (45)$$

From (45) and Proposition 13, we obtain that

$$\begin{aligned} g_1(n) &\in \Omega(g_3(n)), \\ g_2(n) &\in O(g_4(n)). \end{aligned} \quad (46)$$

From the hypothesis and Corollary 16, we have that

$$\begin{aligned} g_3(n) &\in \overline{\Theta}(g_3(n), g_4(n)) = \overline{\Theta}(g_1(n), g_2(n)), \\ g_4(n) &\in \overline{\Theta}(g_3(n), g_4(n)) = \overline{\Theta}(g_1(n), g_2(n)). \end{aligned} \quad (47)$$

From (47) and Proposition 13, we obtain that

$$\begin{aligned} g_3(n) &\in \Omega(g_1(n)), \\ g_4(n) &\in O(g_2(n)). \end{aligned} \quad (48)$$

Next, from (48), we have

$$\begin{aligned} g_1(n) &\in O(g_3(n)), \\ g_2(n) &\in \Omega(g_4(n)). \end{aligned} \quad (49)$$

From (46) and (49), we deduce that

$$\begin{aligned} g_1(n) &\in \Theta(g_3(n)) \\ g_2(n) &\in \Theta(g_4(n)). \end{aligned} \quad (50)$$

“ \Leftarrow ” Consider that $g_1(n) \in \Theta(g_3(n))$ and $g_2(n) \in \Theta(g_4(n))$. It follows that

$$\begin{aligned} \exists c'_1, c'_2 \in \mathbb{R}_+^*, \exists n_{01} \in \mathbb{N}^* \\ \text{such that } c'_1 \cdot g_3(n) \leq g_1(n) \leq c'_2 \cdot g_3(n), \quad \forall n \geq n_{01}, \\ \exists c''_1, c''_2 \in \mathbb{R}_+^*, \exists n_{02} \in \mathbb{N}^* \\ \text{such that } c''_1 \cdot g_4(n) \leq g_2(n) \leq c''_2 \cdot g_4(n), \quad \forall n \geq n_{02}. \end{aligned} \quad (51)$$

Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n))$. It follows that

$$\begin{aligned} \exists c'_3, c'_4 \in \mathbb{R}_+^*, \exists n_{03} \in \mathbb{N}^* \\ \text{such that } c'_3 \cdot g_1(n) \leq f(n) \leq c'_4 \cdot g_2(n), \quad \forall n \geq n_{03}. \end{aligned} \quad (52)$$

Using (51) and (52), we have

$$\begin{aligned} \exists c''_3 = c'_1 \cdot c'_3, c''_4 = c'_2 \cdot c''_2 \in \mathbb{R}_+^*, \\ \exists n_{04} = \max(n_{01}, n_{02}, n_{03}) \in \mathbb{N}^* \end{aligned} \quad (53)$$

such that $c''_3 \cdot g_3(n) \leq f(n) \leq c''_4 \cdot g_4(n), \quad \forall n \geq n_{04}$.

Consequently,

$$f(n) \in \overline{\Theta}(g_3(n), g_4(n)). \quad (54)$$

The other inclusion, $\overline{\Theta}(g_3(n), g_4(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n))$, can be proved using a similar idea. \square

Proposition 19. Let $g_1, g_2 \in \mathcal{F}$. Then,

$$\begin{aligned} g_1(n) &\in \Theta(g_2(n)) \\ \text{iff } \overline{\Theta}(g_1(n), g_2(n)) &= \overline{\Theta}(g_2(n), g_1(n)) \neq \emptyset. \end{aligned} \quad (55)$$

Proof. “ \Rightarrow ” Consider that $g_1(n) \in \Theta(g_2(n))$. We prove that

$$\overline{\Theta}(g_1(n), g_2(n)) \subseteq \overline{\Theta}(g_2(n), g_1(n)). \quad (56)$$

Since $g_1(n) \in \Theta(g_2(n))$, it follows that $g_1(n) \in O(g_2(n))$. Consequently, we obtain the relation $\overline{\Theta}(g_1(n), g_2(n)) \neq \emptyset$.

Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n))$. It follows that

$$\begin{aligned} \exists c_1, c_2 \in \mathbb{R}_+^*, \exists n_0 \in \mathbb{N}^* \\ \text{such that } c_1 \cdot g_1(n) \leq f(n) \leq c_2 \cdot g_2(n), \quad \forall n \geq n_0. \end{aligned} \quad (57)$$

From the relation $g_1(n) \in \Theta(g_2(n))$, we have

$$\begin{aligned} \exists c'_1, c'_2 \in \mathbb{R}_+^*, \exists n'_0 \in \mathbb{N}^* \\ \text{such that } c'_1 \cdot g_2(n) \leq g_1(n) \leq c'_2 \cdot g_2(n), \quad \forall n \geq n'_0. \end{aligned} \quad (58)$$

Next, using the relation

$$g_1(n) \in \Theta(g_2(n)) \quad \text{iff } g_2(n) \in \Theta(g_1(n)), \quad (59)$$

we obtain that

$$\begin{aligned} \exists c''_1, c''_2 \in \mathbb{R}_+^*, \exists n''_0 \in \mathbb{N}^* \\ \text{such that } c''_1 \cdot g_1(n) \leq g_2(n) \leq c''_2 \cdot g_1(n), \quad \forall n \geq n''_0. \end{aligned} \quad (60)$$

From (57), (58), and (60), we have that

$$\begin{aligned} \exists \bar{c}_1 = c_1 \cdot c'_1, \bar{c}_2 = c_2 \cdot c''_2 \in \mathbb{R}_+^*, \\ \exists \bar{n}_0 = \max(n_0, n'_0, n''_0) \in \mathbb{N}^* \end{aligned} \quad (61)$$

such that $\bar{c}_1 \cdot g_2(n) \leq f(n) \leq \bar{c}_2 \cdot g_1(n), \quad \forall n \geq \bar{n}_0$.

Consequently,

$$f(n) \in \overline{\Theta}(g_2(n), g_1(n)). \quad (62)$$

The proof of $\overline{\Theta}(g_2(n), g_1(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n))$ follows the same idea used for the first inclusion.

“ \Leftarrow ” We consider that $\overline{\Theta}(g_1(n), g_2(n)) = \overline{\Theta}(g_2(n), g_1(n)) \neq \emptyset$.

Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n)) = \overline{\Theta}(g_2(n), g_1(n))$. It follows that

$$\begin{aligned} & \exists c'_1, c'_2 \in \mathbb{R}_+, \exists n'_0 \in \mathbb{N}^* \\ & \text{such that } c'_1 \cdot g_1(n) \leq f(n) \leq c'_2 \cdot g_2(n), \quad \forall n \geq n'_0, \\ & \exists c''_1, c''_2 \in \mathbb{R}_+, \exists n''_0 \in \mathbb{N}^* \\ & \text{such that } c''_1 \cdot g_2(n) \leq f(n) \leq c''_2 \cdot g_1(n), \quad \forall n \geq n''_0. \end{aligned} \quad (63)$$

From (63), we obtain

$$\begin{aligned} & \exists c'_1, c'_2 \in \mathbb{R}_+, \exists n'_0 \in \mathbb{N}^* \\ & \text{such that } g_1(n) \leq \frac{c'_2}{c'_1} \cdot g_2(n), \quad \forall n \geq n'_0, \\ & \exists c''_1, c''_2 \in \mathbb{R}_+, \exists n''_0 \in \mathbb{N}^* \\ & \text{such that } \frac{c''_1}{c''_2} \cdot g_2(n) \leq g_1(n), \quad \forall n \geq n''_0. \end{aligned} \quad (64)$$

Using (64), we have

$$\begin{aligned} & \exists c_1 = \frac{c''_1}{c''_2}, c_2 = \frac{c'_2}{c'_1} \in \mathbb{R}_+, \\ & \exists n_0 = \max(n'_0, n''_0) \in \mathbb{N}^* \\ & \text{such that } c_1 \cdot g_2(n) \leq g_1(n) \leq c_2 \cdot g_2(n), \quad \forall n \geq n_0. \end{aligned} \quad (65)$$

Consequently,

$$g_1(n) \in \Theta(g_2(n)). \quad (66)$$

□

5. Membership, Inclusion, Intersection, and Union

Proposition 20. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n))$. Then,

- (a) $\overline{\Theta}(g_1(n), f(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n))$;
- $\overline{\Theta}(f(n), g_2(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n))$;
- (b) $\overline{\Theta}(g_1(n), f(n)) \cap \overline{\Theta}(f(n), g_2(n)) = \Theta(f(n))$;
- (c) $\overline{\Theta}(g_1(n), f(n)) \cup \overline{\Theta}(f(n), g_2(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n))$.

Proof. (a) Let $h(n) \in \overline{\Theta}(g_1(n), f(n))$. It follows that

$$\begin{aligned} & \exists c_1, c_2 \in \mathbb{R}_+, \exists n_0 \in \mathbb{N}^* \\ & \text{such that } c_1 \cdot g_1(n) \leq h(n) \leq c_2 \cdot f(n), \quad \forall n \geq n_0. \end{aligned} \quad (67)$$

From the relation $f(n) \in \overline{\Theta}(g_1(n), g_2(n))$, we have

$$\begin{aligned} & \exists c'_1, c'_2 \in \mathbb{R}_+, \exists n'_0 \in \mathbb{N}^* \\ & \text{such that } c'_1 \cdot g_1(n) \leq f(n) \leq c'_2 \cdot g_2(n), \quad \forall n \geq n'_0. \end{aligned} \quad (68)$$

From (68), we obtain

$$\begin{aligned} & \exists c''_1, c''_2 \in \mathbb{R}_+, \exists n''_0 \in \mathbb{N}^* \\ & \text{such that } c_2 \cdot c'_1 \cdot g_1(n) \leq c_2 \cdot f(n) \leq c_2 \cdot c'_2 \cdot g_2(n), \quad \forall n \geq n''_0. \end{aligned} \quad (69)$$

Using (67) and (69), we have that

$$\begin{aligned} & \exists c''_1 = c_1, c''_2 = c_2 \cdot c'_2 \in \mathbb{R}_+, \\ & \exists n''_0 = \max(n_0, n''_0) \in \mathbb{N}^* \\ & \text{such that } c''_1 \cdot g_1(n) \leq h(n) \leq c''_2 \cdot g_2(n), \quad \forall n \geq n''_0. \end{aligned} \quad (70)$$

Consequently,

$$h(n) \in \overline{\Theta}(g_1(n), g_2(n)). \quad (71)$$

For proving that $\overline{\Theta}(f(n), g_2(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n))$, one can follow the same idea used for the first relation of (a).

(b) From Corollary 16, it follows that

$$\Theta(f(n)) \subseteq \overline{\Theta}(g_1(n), f(n)) \cap \overline{\Theta}(f(n), g_2(n)). \quad (72)$$

Let $h(n) \in \overline{\Theta}(g_1(n), f(n)) \cap \overline{\Theta}(f(n), g_2(n))$. It follows that

$$\begin{aligned} & \exists c'_1, c'_2 \in \mathbb{R}_+, \exists n'_0 \in \mathbb{N}^* \\ & \text{such that } c'_1 \cdot g_1(n) \leq h(n) \leq c'_2 \cdot f(n), \quad \forall n \geq n'_0, \\ & \exists c''_1, c''_2 \in \mathbb{R}_+, \exists n''_0 \in \mathbb{N}^* \\ & \text{such that } c''_1 \cdot f(n) \leq h(n) \leq c''_2 \cdot g_2(n), \quad \forall n \geq n''_0. \end{aligned} \quad (73)$$

From (73), we obtain

$$\begin{aligned} & \exists c_1 = c'_1, c_2 = c'_2 \in \mathbb{R}_+, \\ & \exists n_0 = \max(n'_0, n''_0) \in \mathbb{N}^* \\ & \text{such that } c_1 \cdot f(n) \leq h(n) \leq c_2 \cdot f(n), \quad \forall n \geq n_0. \end{aligned} \quad (74)$$

Consequently,

$$h(n) \in \Theta(f(n)). \quad (75)$$

(c) The proof follows from (a). □

Proposition 21. Let $g_1, g_2, g_3 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$ and $g_2(n) \in O(g_3(n))$. Then,

- (a) $\overline{\Theta}(g_1(n), g_3(n)) \neq \emptyset$,
- (b) $g_2(n) \in \overline{\Theta}(g_1(n), g_3(n))$.

Proof. (a) We use the transitivity of the complexity class “O” (see Proposition 4), described as

$$\begin{aligned} &\text{if } g_1(n) \in O(g_2(n)), g_2(n) \in O(g_3(n)) \\ &\text{then } g_1(n) \in O(g_3(n)). \end{aligned} \quad (76)$$

Hence, from the hypothesis and (76), we have

$$g_1(n) \in O(g_3(n)). \quad (77)$$

Next, using (77) and Proposition II(a), we obtain that

$$\overline{\Theta}(g_1(n), g_3(n)) \neq \emptyset. \quad (78)$$

(b) Using the hypothesis and Proposition 4(d), we have that

$$g_2(n) \in \Omega(g_1(n)). \quad (79)$$

Consequently, using (79) and the hypothesis, we have

$$g_2(n) \in \Omega(g_1(n)) \cap O(g_3(n)). \quad (80)$$

From (80) and Proposition 13, we have that

$$g_2(n) \in \overline{\Theta}(g_1(n), g_3(n)). \quad (81)$$

□

Proposition 22. Let $g_1, g_2, g_3 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$ and $g_2(n) \in O(g_3(n))$. Let $f_1(n) \in \overline{\Theta}(g_1(n), g_2(n))$, $f_2(n) \in \overline{\Theta}(g_2(n), g_3(n))$, and $f \in \mathcal{F}$ with the form

$$f(n) = \begin{cases} f_1(n), & \text{if } n \in M_1 \\ f_2(n), & \text{if } n \in M_2, \end{cases} \quad (82)$$

where M_1 and M_2 are two infinite subsets of \mathbb{N}^* such that $M_1 \cup M_2 = \mathbb{N}^*$ and $M_1 \cap M_2 = \emptyset$. Then $f(n) \in \overline{\Theta}(g_1(n), g_3(n))$.

Proof. Using the hypothesis and Proposition 21, we have that

$$g_2(n) \in \overline{\Theta}(g_1(n), g_3(n)). \quad (83)$$

Also, from the hypothesis and Proposition 4(b), we obtain

$$g_1(n) \in O(g_3(n)). \quad (84)$$

From (84), (83), and Proposition 20, we have

$$\begin{aligned} \overline{\Theta}(g_1(n), g_2(n)) &\subseteq \overline{\Theta}(g_1(n), g_3(n)), \\ \overline{\Theta}(g_2(n), g_3(n)) &\subseteq \overline{\Theta}(g_1(n), g_3(n)). \end{aligned} \quad (85)$$

Using (85) and the hypothesis, we obtain that

$$f_1(n) \in \overline{\Theta}(g_1(n), g_3(n)), \quad (86)$$

$$f_2(n) \in \overline{\Theta}(g_1(n), g_3(n)).$$

Consequently,

$$f(n) \in \overline{\Theta}(g_1(n), g_3(n)). \quad (87)$$

□

Proposition 23. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n))$. Then,

(a) if $f(n) \in \Theta(g_1(n))$ or $f(n) \in \Theta(g_2(n))$, then

$$\overline{\Theta}(g_1(n), f(n)) \cup \overline{\Theta}(f(n), g_2(n)) = \overline{\Theta}(g_1(n), g_2(n)); \quad (88)$$

(b) if $\overline{\Theta}(g_1(n), f(n)) \cup \overline{\Theta}(f(n), g_2(n)) = \overline{\Theta}(g_1(n), g_2(n))$, then

$$f(n) \notin \omega(g_1(n)) \quad \text{or} \quad f(n) \notin o(g_2(n)). \quad (89)$$

Proof. (a) Consider the following.

Case 1. Consider $f(n) \in \Theta(g_1(n))$. From Proposition 17, we have that

$$\overline{\Theta}(g_1(n), f(n)) = \Theta(f(n)). \quad (90)$$

From (90) and Corollary 16, we have that

$$\overline{\Theta}(g_1(n), f(n)) \subseteq \overline{\Theta}(f(n), g_2(n)). \quad (91)$$

Next, from the relation $f(n) \in \Theta(g_1(n))$ and Proposition 18, we have that

$$\overline{\Theta}(f(n), g_2(n)) = \overline{\Theta}(g_1(n), g_2(n)). \quad (92)$$

Case 2. Consider $f(n) \in \Theta(g_2(n))$. The proof follows the same idea used for proving Case 1.

(b) Suppose for a contradiction that $f(n) \in \omega(g_1(n))$ and $f(n) \in o(g_2(n))$.

Let $h \in \mathcal{F}$ such that

$$h(n) = \begin{cases} g_1(n), & \text{if } n \in M_1 \\ g_2(n), & \text{if } n \in M_2, \end{cases} \quad (93)$$

where M_1 and M_2 are infinite subsets of \mathbb{N}^* such that $M_1 \cup M_2 = \mathbb{N}^*$ and $M_1 \cap M_2 = \emptyset$. From the relations $g_1(n) \in \overline{\Theta}(g_1(n), f(n))$ and $g_2(n) \in \overline{\Theta}(f(n), g_2(n))$ and from Proposition 22 we have that

$$h(n) \in \overline{\Theta}(g_1(n), g_2(n)). \quad (94)$$

Since

$$\overline{\Theta}(g_1(n), g_2(n)) = \overline{\Theta}(g_1(n), f(n)) \cup \overline{\Theta}(f(n), g_2(n)), \quad (95)$$

then

$$h(n) \in \overline{\Theta}(g_1(n), f(n)) \quad (96)$$

or

$$h(n) \in \overline{\Theta}(f(n), g_2(n)). \quad (97)$$

Case 1. Consider $h(n) \in \overline{\Theta}(g_1(n), f(n))$.

From (93) and the relation $h(n) \in \overline{\Theta}(g_1(n), f(n))$, we have that

$$\begin{aligned} \exists c_1 \in \mathbb{R}_+^*, \exists n_{01} \in \mathbb{N}^* \\ \text{such that } g_2(n) \leq c_1 \cdot f(n), \quad \forall n \in M_2, n \geq n_{01}. \end{aligned} \quad (98)$$

Using the relation $f(n) \in o(g_2(n))$, it follows that

$$\begin{aligned} \forall c' \in \mathbb{R}_+^*, \exists n'_0 \in \mathbb{N}^* \\ \text{such that } f(n) < c' \cdot g_2(n), \quad \forall n \geq n'_0. \end{aligned} \quad (99)$$

We take in (99) $c' = 1/c_1$ and we obtain that

$$\exists n_{02} \in \mathbb{N}^* \text{ such that } f(n) < \frac{1}{c_1} \cdot g_2(n), \quad \forall n \geq n_{02}. \quad (100)$$

From (98), (100), and the fact that M_2 is an infinite subset of \mathbb{N}^* , we obtain that

$$\begin{aligned} \exists n_0 = \max(n_{01}, n_{02}), \exists n \in M_2, n \geq n_0 \\ \text{such that } g_2(n) \leq c_1 \cdot f(n) < g_2(n) \end{aligned} \quad (101)$$

which is a contradiction.

Case 2. Consider $h(n) \in \overline{\Theta}(f(n), g_2(n))$.

From (93) and the relation $h(n) \in \overline{\Theta}(f(n), g_2(n))$, we have that

$$\begin{aligned} \exists c'_1 \in \mathbb{R}_+^*, \exists n'_{01} \in \mathbb{N}^* \\ \text{such that } c'_1 \cdot f(n) \leq g_1(n), \quad \forall n \in M_1, n \geq n'_{01}. \end{aligned} \quad (102)$$

Using the relation $f(n) \in \omega(g_1(n))$, it follows that

$$\begin{aligned} \forall c'' \in \mathbb{R}_+^*, \exists n''_0 \in \mathbb{N}^* \\ \text{such that } c'' \cdot g_1(n) < f(n), \quad \forall n \geq n''_0. \end{aligned} \quad (103)$$

We take in (103) $c'' = 1/c'_1$ and we obtain that

$$\exists n'_{02} \in \mathbb{N}^* \text{ such that } \frac{1}{c'_1} \cdot g_1(n) < f(n), \quad \forall n \geq n'_{02}. \quad (104)$$

From (102), (104), and the fact that M_1 is an infinite subset of \mathbb{N}^* , we obtain that

$$\begin{aligned} \exists n'_0 = \max(n'_{01}, n'_{02}), \exists n \in M_1, n \geq n'_0 \\ \text{such that } g_1(n) < c'_1 \cdot f(n) \leq g_1(n) \end{aligned} \quad (105)$$

which is a contradiction. \square

Proposition 24. Let $g_1, g_2, g_3, g_4 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$ and $g_3(n) \in O(g_4(n))$. Then,

(a) if $g_2(n) \in \Theta(g_3(n))$, then

$$\begin{aligned} \overline{\Theta}(g_1(n), g_2(n)) \cap \overline{\Theta}(g_3(n), g_4(n)) \\ = \Theta(g_2(n)) = \Theta(g_3(n)); \end{aligned} \quad (106)$$

(b) if $g_2(n) \in o(g_3(n))$, then

$$\overline{\Theta}(g_1(n), g_2(n)) \cap \overline{\Theta}(g_3(n), g_4(n)) = \emptyset; \quad (107)$$

(c) if $g_2(n) \in \omega(g_3(n))$, $g_2(n) \in \overline{\Theta}(g_3(n), g_4(n))$, and $g_3(n) \in \overline{\Theta}(g_1(n), g_2(n))$, then

$$\overline{\Theta}(g_1(n), g_2(n)) \cap \overline{\Theta}(g_3(n), g_4(n)) = \overline{\Theta}(g_3(n), g_2(n)), \quad (108)$$

where $g_3(n) \notin \Theta(g_2(n))$.

Proof. (a) The relation

$$\Theta(g_2(n)) = \Theta(g_3(n)) \quad (109)$$

is obvious from $g_2(n) \in \Theta(g_3(n))$. We prove that

$$\overline{\Theta}(g_1(n), g_2(n)) \cap \overline{\Theta}(g_3(n), g_4(n)) = \Theta(g_2(n)). \quad (110)$$

(i) From the hypothesis, one can easily obtain that $\overline{\Theta}(g_1(n), g_2(n)) \cap \overline{\Theta}(g_3(n), g_4(n)) \neq \emptyset$.

Let $f(n) \in \overline{\Theta}(g_1(n), g_2(n)) \cap \overline{\Theta}(g_3(n), g_4(n))$. It follows that

$$\begin{aligned} \exists c'_1, c'_2 \in \mathbb{R}_+^*, \exists n_{01} \in \mathbb{N}^* \\ \text{such that } c'_1 \cdot g_1(n) \leq f(n) \leq c'_2 \cdot g_2(n), \quad \forall n \geq n_{01}, \\ \exists c''_1, c''_2 \in \mathbb{R}_+^*, \exists n_{02} \in \mathbb{N}^* \\ \text{such that } c''_1 \cdot g_3(n) \leq f(n) \leq c''_2 \cdot g_4(n), \quad \forall n \geq n_{02}. \end{aligned} \quad (111)$$

From the relation $g_2(n) \in \Theta(g_3(n))$, we have that

$$\begin{aligned} \exists c'''_1, c'''_2 \in \mathbb{R}_+^*, \exists n_{03} \in \mathbb{N}^* \\ \text{such that } c'''_1 \cdot g_3(n) \leq g_2(n) \leq c'''_2 \cdot g_3(n), \quad \forall n \geq n_{03}. \end{aligned} \quad (112)$$

From (112), we have that

$$\begin{aligned} \exists c_3 = \frac{1}{c'''_2} \in \mathbb{R}_+^*, \exists n_{04} = n_{03} \in \mathbb{N}^* \\ \text{such that } c_3 \cdot g_2(n) \leq g_3(n), \quad \forall n \geq n_{04}. \end{aligned} \quad (113)$$

Using (111) and (113), we obtain

$$\begin{aligned} \exists c_4 = c''_1 \cdot c_3, c_5 = c'_2 \in \mathbb{R}_+^*, \exists n_{05} = \max(n_{01}, n_{02}, n_{04}) \in \mathbb{N}^* \\ \text{such that } c_4 \cdot g_2(n) \leq f(n) \leq c_5 \cdot g_2(n), \quad \forall n \geq n_{05}. \end{aligned} \quad (114)$$

Consequently,

$$f(n) \in \Theta(g_2(n)). \quad (115)$$

(ii) From Corollary 16, we have

$$\Theta(g_2(n)) \subseteq \overline{\Theta}(g_1(n), g_2(n)). \quad (116)$$

From (109) and Corollary 16, we have

$$\Theta(g_2(n)) = \Theta(g_3(n)) \subseteq \overline{\overline{\Theta}}(g_3(n), g_4(n)). \quad (117)$$

Using (116) and (117), we obtain that

$$\Theta(g_2(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n)). \quad (118)$$

(b) Suppose for a contradiction that there exists $f \in \mathcal{F}$ such that $f(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n))$. It follows that

$$\begin{aligned} &\exists c'_1, c'_2 \in \mathbb{R}_+^*, \exists n_{01} \in \mathbb{N}^* \\ &\text{such that } c'_1 \cdot g_1(n) \leq f(n) \leq c'_2 \cdot g_2(n), \quad \forall n \geq n_{01}, \\ &\exists c''_1, c''_2 \in \mathbb{R}_+^*, \exists n_{02} \in \mathbb{N}^* \\ &\text{such that } c''_1 \cdot g_3(n) \leq f(n) \leq c''_2 \cdot g_4(n), \quad \forall n \geq n_{02}. \end{aligned} \quad (119)$$

From the relation $g_2(n) \in o(g_3(n))$, we have that

$$\begin{aligned} &\forall c_1''' \in \mathbb{R}_+^*, \exists n_0''' \in \mathbb{N}^* \\ &\text{such that } g_2(n) < c_1''' \cdot g_3(n), \quad \forall n \geq n_0'''. \end{aligned} \quad (120)$$

We take in (120) $c_1''' = c''_1/c'_2$ and we obtain that

$$\exists n_{03} \in \mathbb{N}^* \text{ such that } c'_2 \cdot g_2(n) < c''_1 \cdot g_3(n), \quad \forall n \geq n_{03}. \quad (121)$$

From (119) and (121), we have that

$$\begin{aligned} &\exists n_0 = \max(n_{01}, n_{02}, n_{03}) \in \mathbb{N}^* \\ &\text{such that } c'_1 \cdot g_1(n) \leq f(n) \leq c'_2 \cdot g_2(n) < c''_1 \cdot g_3(n) \\ &\leq f(n) \leq c''_2 \cdot g_4(n), \quad \forall n \geq n_0 \end{aligned} \quad (122)$$

which is a contradiction.

We conclude that

$$\overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n)) = \emptyset. \quad (123)$$

(c) From relation $g_2(n) \in \omega(g_3(n))$, we have that

$$g_2(n) \in \Omega(g_3(n)). \quad (124)$$

Thus, from Proposition 11, it follows that

$$\overline{\overline{\Theta}}(g_3(n), g_2(n)) \neq \emptyset. \quad (125)$$

Furthermore, we prove that $g_3(n) \notin \Theta(g_2(n))$. Suppose for a contradiction that $g_3(n) \in \Theta(g_2(n))$. It follows that

$$\begin{aligned} &\exists \bar{c}_1, \bar{c}_2 \in \mathbb{R}_+^*, \exists \bar{n}_0 \in \mathbb{N}^* \\ &\text{such that } \bar{c}_1 \cdot g_2(n) \leq g_3(n) \leq \bar{c}_2 \cdot g_2(n), \quad \forall n \geq \bar{n}_0. \end{aligned} \quad (126)$$

Thus, from (126), we have that

$$\exists \bar{c}_1 = \frac{1}{\bar{c}_1} \in \mathbb{R}_+^*, \exists \bar{n}_0 = \bar{n}_0 \in \mathbb{N}^* \quad (127)$$

$$\text{such that } g_2(n) \leq \bar{c}_1 \cdot g_3(n), \quad \forall n \geq \bar{n}_0.$$

From $g_2(n) \in \omega(g_3(n))$, we have that

$$\forall \bar{c}' \in \mathbb{R}_+^*, \exists \bar{n}'_0 \in \mathbb{N}^* \quad (128)$$

$$\text{such that } \bar{c}' \cdot g_3(n) < g_2(n), \quad \forall n \geq \bar{n}'_0.$$

We take in (128) $\bar{c}' = \bar{c}_1$ and we obtain that

$$\exists \bar{n}'_0 \in \mathbb{N}^* \text{ such that } \bar{c}_1 \cdot g_3(n) < g_2(n), \quad \forall n \geq \bar{n}'_0. \quad (129)$$

From (127) and (129), we obtain that

$$\exists n_0 = \max(\bar{n}_0, \bar{n}'_0) \in \mathbb{N}^* \quad (130)$$

$$\text{such that } g_2(n) \leq \bar{c}_1 \cdot g_3(n) < g_2(n), \quad \forall n \geq n_0$$

which is a contradiction. Consequently, we have $g_3(n) \notin \Theta(g_2(n))$.

Now, we prove that $\overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n)) = \overline{\overline{\Theta}}(g_3(n), g_2(n))$.

(i) First, we prove that $\overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n)) \subseteq \overline{\overline{\Theta}}(g_3(n), g_2(n))$.

From the hypothesis one can easily obtain that $\overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n)) \neq \emptyset$.

Let $f_1(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n))$. It follows that

$$\begin{aligned} &\exists c'_1, c'_2 \in \mathbb{R}_+^*, \exists n'_0 \in \mathbb{N}^* \\ &\text{such that } c'_1 \cdot g_1(n) \leq f_1(n) \leq c'_2 \cdot g_2(n), \quad \forall n \geq n'_0, \\ &\exists c''_1, c''_2 \in \mathbb{R}_+^*, \exists n''_0 \in \mathbb{N}^* \\ &\text{such that } c''_1 \cdot g_3(n) \leq f_1(n) \leq c''_2 \cdot g_4(n), \quad \forall n \geq n''_0. \end{aligned} \quad (131)$$

From (131), we obtain that

$$\exists c_1 = c'_1, c_2 = c'_2 \in \mathbb{R}_+^*, \exists n_0 = \max(n'_0, n''_0) \in \mathbb{N}^* \quad (132)$$

$$\text{such that } c_1 \cdot g_3(n) \leq f_1(n) \leq c_2 \cdot g_2(n), \quad \forall n \geq n_0.$$

Consequently,

$$f_1(n) \in \overline{\overline{\Theta}}(g_3(n), g_2(n)). \quad (133)$$

(ii) We prove that $\overline{\overline{\Theta}}(g_3(n), g_2(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n))$.

Let $f_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_2(n))$. From Proposition 13, we have

$$f_2(n) \in \Omega(g_3(n)), \quad (134)$$

$$f_2(n) \in O(g_2(n)). \quad (135)$$

From the hypothesis and Proposition 13, we have

$$g_2(n) \in \Omega(g_3(n)) \cap O(g_4(n)), \quad (136)$$

$$g_3(n) \in \Omega(g_1(n)) \cap O(g_2(n)). \quad (137)$$

From (134), (137), and Proposition 4 (transitivity of Ω), we have that

$$f_2(n) \in \Omega(g_1(n)). \quad (138)$$

From (135), (136), and Proposition 4 (transitivity of O), we have that

$$f_2(n) \in O(g_4(n)). \quad (139)$$

Using (135), (138), and Proposition 13, we have

$$f_2(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n)). \quad (140)$$

Using (134), (139), and Proposition 13, we have

$$f_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_4(n)). \quad (141)$$

Consequently,

$$f_2(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cap \overline{\overline{\Theta}}(g_3(n), g_4(n)). \quad (142)$$

□

Proposition 25. Let $g_1, g_2, \dots, g_k \in \mathcal{F}$ such that $g_i(n) \in O(g_{i+1}(n))$, $\forall i = 1, 2, \dots, k-1$. Then

$$(a) \overline{\overline{\Theta}}(g_i(n), g_{i+1}(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_k(n)), \forall i = 1, 2, \dots, k-1;$$

$$(b) \overline{\overline{\Theta}}(g_i(n), g_{i+1}(n)) \cap \overline{\overline{\Theta}}(g_{i+1}(n), g_{i+2}(n)) = \Theta(g_{i+1}(n)), \forall i = 1, 2, \dots, k-2;$$

$$(c) \bigcup_{i=1}^{k-1} \overline{\overline{\Theta}}(g_i(n), g_{i+1}(n)) \subseteq \overline{\overline{\Theta}}(g_1(n), g_k(n)).$$

Proof. (a) Let $i \in \{1, 2, \dots, k-1\}$ arbitrarily fixed.

From the hypothesis and Proposition 4 (transitivity of O), we have

$$g_1(n) \in O(g_i(n)), \quad (143)$$

$$g_{i+1}(n) \in O(g_k(n)). \quad (144)$$

Let $f(n) \in \overline{\overline{\Theta}}(g_i(n), g_{i+1}(n))$. Then, using Proposition 13, we have

$$f(n) \in \Omega(g_i(n)), \quad (145)$$

$$f(n) \in O(g_{i+1}(n)). \quad (146)$$

From (143) and Proposition 4 (transpose symmetry), we have

$$g_i(n) \in \Omega(g_1(n)). \quad (147)$$

Thus, from (145), (147), and Proposition 4 (transitivity of Ω), we obtain

$$f(n) \in \Omega(g_1(n)). \quad (148)$$

From (146), (144), and Proposition 4 (transitivity of O), we obtain

$$f(n) \in O(g_k(n)). \quad (149)$$

Using (148), (149), and Proposition 13, we have that

$$f(n) \in \overline{\overline{\Theta}}(g_1(n), g_k(n)). \quad (150)$$

(b) Let $i \in \{1, 2, \dots, k-2\}$ arbitrarily fixed. Since

$$g_i(n) \in O(g_{i+1}(n)) \quad (g_{i+1}(n) \in \Omega(g_i(n))), \quad (151)$$

$$g_{i+1}(n) \in O(g_{i+2}(n)),$$

we obtain, using Proposition 13, the relation

$$g_{i+1}(n) \in \overline{\overline{\Theta}}(g_i(n), g_{i+2}(n)). \quad (152)$$

Next, using Proposition 20(b), we obtain that

$$\overline{\overline{\Theta}}(g_i(n), g_{i+1}(n)) \cap \overline{\overline{\Theta}}(g_{i+1}(n), g_{i+2}(n)) = \Theta(g_{i+1}(n)). \quad (153)$$

(c) The proof results easily from (a). □

6. Some Relations between Complexity Functions

Proposition 26. Let $g_1, g_2, g_3, g_4 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$ and $g_3(n) \in O(g_4(n))$. Let $f_1, f_2 \in \mathcal{F}$ such that $f_1(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n))$ and $f_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_4(n))$. Then,

$$(a) \text{ if } g_2(n) \in O(g_3(n)), \text{ then } f_1(n) \in C(f_2(n));$$

$$(b) \text{ if } g_2(n) \in \Theta(g_3(n)), \text{ then } f_1(n) \in C(f_2(n));$$

$$(c) \text{ if } g_2(n) \in o(g_3(n)), \text{ then } f_1(n) \in C(f_2(n));$$

$$(d) \text{ if } g_2(n) \in o\Theta(g_3(n)), \text{ then } f_1(n) \in C(f_2(n)).$$

Proof. (a) From the hypothesis and Proposition 13, we have that

$$f_1(n) \in \Omega(g_1(n)) \cap O(g_2(n)), \quad (154)$$

$$f_2(n) \in \Omega(g_3(n)) \cap O(g_4(n)).$$

It follows that

$$f_1(n) \in O(g_2(n)), \quad (155)$$

$$g_3(n) \in O(f_2(n)).$$

Next, using the relation $g_2(n) \in O(g_3(n))$ and Proposition 4 (the transitivity of O), we obtain the relation

$$f_1(n) \in O(f_2(n)). \quad (156)$$

In the end, using Definition 5, we obtain that

$$f_1(n) \in C(f_2(n)). \quad (157)$$

((b), (c), and (d)) The proof follows from (a) using the relations

$$\Theta(g_3(n)) \subseteq O(g_3(n)),$$

$$o(g_3(n)) \subseteq O(g_3(n)), \quad (158)$$

$$o\Theta(g_3(n)) \subseteq O(g_3(n)).$$

□

Proposition 27. Let $g_1, g_2, g_3, g_4 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$, $g_2(n) \in \omega(g_3(n))$, $g_3(n) \in O(g_4(n))$, $g_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_4(n))$, and $g_3(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n))$. Let $f_1, f_2 \in \mathcal{F}$ such that $f_1(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n))$ and $f_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_4(n))$. Then,

- (a) if $f_1(n) \in \overline{\overline{\Theta}}(g_1(n), g_3(n))$ and $f_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_2(n))$, then $f_1(n) \in C(f_2(n))$;
- (b) if $f_1(n) \in \overline{\overline{\Theta}}(g_3(n), g_2(n))$ and $f_2(n) \in \overline{\overline{\Theta}}(g_2(n), g_4(n))$, then $f_1(n) \in C(f_2(n))$;
- (c) if $f_1(n) \in \overline{\overline{\Theta}}(g_1(n), g_3(n))$ and $f_2(n) \in \overline{\overline{\Theta}}(g_2(n), g_4(n))$, then $f_1(n) \in C(f_2(n))$.

Proof. (a) Since $g_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_4(n))$ then, from Proposition 13, we have that

$$g_2(n) \in \Omega(g_3(n)). \quad (159)$$

Next, from Proposition 11, we have that

$$\overline{\overline{\Theta}}(g_3(n), g_2(n)) \neq \emptyset. \quad (160)$$

From $f_1(n) \in \overline{\overline{\Theta}}(g_1(n), g_3(n))$, $f_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_2(n))$, and Proposition 13, we obtain that

$$\begin{aligned} f_1(n) &\in O(g_3(n)), \\ g_3(n) &\in O(f_2(n)). \end{aligned} \quad (161)$$

Thus, from Proposition 4 (the transitivity of O), we have

$$f_1(n) \in O(f_2(n)). \quad (162)$$

Consequently,

$$f_1(n) \in C(f_2(n)). \quad (163)$$

((b) and (c)) The proof follows the same idea used for (a). \square

Remark 28. In Proposition 27, the hypothesis $g_2(n) \in \omega(g_3(n))$ is not used in the proof. The purpose of this hypothesis is to emphasize that Propositions 26 and 27 discuss disjoint cases.

Proposition 29. There exist $g_1, g_2, g_3, g_4, f_1, f_2 \in \mathcal{F}$ with the following properties: (i1) $g_1(n) \in O(g_2(n))$, (i2) $g_2(n) \in \omega(g_3(n))$, (i3) $g_3(n) \in O(g_4(n))$, (i4) $g_3(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n))$, (i5) $g_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_4(n))$, (i6) $f_1(n) \in \overline{\overline{\Theta}}(g_1(n), g_2(n))$, and (i7) $f_2(n) \in \overline{\overline{\Theta}}(g_3(n), g_4(n))$ such that $f_1(n) \notin C(f_2(n))$.

Proof. Let $g_1(n) = n$, $g_2(n) = n^7$, $g_3(n) = n^2$, and $g_4(n) = n^9$. The properties (i1)–(i5) are easily verified since $n \in O(n^7)$, $n^7 \in \omega(n^2)$, $n^2 \in O(n^9)$, $n^2 \in \overline{\overline{\Theta}}(n, n^7)$, and $n^7 \in \overline{\overline{\Theta}}(n^2, n^9)$.

Let

$$\begin{aligned} f_1(n) &= n^5, \\ f_2(n) &= \begin{cases} n^4, & \text{if } n \in M_1 \\ n^6, & \text{if } n \in M_2, \end{cases} \end{aligned} \quad (164)$$

where M_1 and M_2 are two infinite subsets of \mathbb{N}^* such that $M_1 \cup M_2 = \mathbb{N}^*$ and $M_1 \cap M_2 = \emptyset$. One can easily verify that

$$\begin{aligned} f_1(n) &\in \overline{\overline{\Theta}}(n, n^7), \\ f_2(n) &\in \overline{\overline{\Theta}}(n^2, n^9). \end{aligned} \quad (165)$$

For proving that $f_1(n) \notin C(f_2(n))$, we start from the following relation (see Proposition 8):

$$C(f_2(n)) = O(f_2(n)) \cup \Omega(f_2(n)). \quad (166)$$

Suppose for a contradiction that

$$f_1(n) \in O(f_2(n)). \quad (167)$$

It follows that

$$\begin{aligned} \exists c_1 \in \mathbb{R}_+^*, \exists n_{01} \in \mathbb{N}^* \\ \text{such that } f_1(n) \leq c_1 \cdot f_2(n), \quad \forall n \geq n_{01}. \end{aligned} \quad (168)$$

Using (168) for $n \in \{m \in M_1 \mid m \geq n_{01}\}$, we obtain

$$n^5 \leq c_1 \cdot n^4 \quad (169)$$

which is false for $n > c_1$. We know that there exists $n > c_1$, $n \in M_1$, because M_1 is an infinite subset of \mathbb{N}^* . Consequently,

$$f_1(n) \notin O(f_2(n)). \quad (170)$$

Using the same idea one can prove that

$$f_1(n) \notin \Omega(f_2(n)). \quad (171)$$

From (170) and (171), it follows that

$$f_1(n) \notin C(f_2(n)). \quad (172)$$

\square

7. The Set of Complexity Functions Comparable with Two Given Complexity Functions

Definition 30. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. We define the set of complexity functions comparable with $g_1(n)$ and $g_2(n)$, denoted by $\overline{\overline{C}}(g_1(n), g_2(n))$, as follows:

$$\begin{aligned} \overline{\overline{C}}(g_1(n), g_2(n)) \\ = o(g_1(n)) \cup o\Theta(g_1(n)) \cup \overline{\overline{\Theta}}(g_1(n), g_2(n)) \\ \cup \Theta\omega(g_2(n)) \cup \omega(g_2(n)). \end{aligned} \quad (173)$$

Proposition 31. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

$$\begin{aligned} \overline{\overline{C}}(g_1(n), g_2(n)) &= O(g_1(n)) \cup (\Omega(g_1(n)) \cap O(g_2(n))) \cup \Omega(g_2(n)). \end{aligned} \quad (174)$$

Proof. From Definition 30, we have

$$\begin{aligned} \overline{\overline{C}}(g_1(n), g_2(n)) &= o(g_1(n)) \cup o\Theta(g_1(n)) \\ &\cup \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cup \Theta\omega(g_2(n)) \cup \omega(g_2(n)). \end{aligned} \quad (175)$$

From Corollary 16, we have that

$$\begin{aligned} \Theta(g_1(n)) &\subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)), \\ \Theta(g_2(n)) &\subseteq \overline{\overline{\Theta}}(g_1(n), g_2(n)). \end{aligned} \quad (176)$$

It follows that

$$\begin{aligned} \overline{\overline{\Theta}}(g_1(n), g_2(n)) &= \Theta(g_1(n)) \cup \overline{\overline{\Theta}}(g_1(n), g_2(n)) \cup \Theta(g_2(n)). \end{aligned} \quad (177)$$

Consequently, we obtain

$$\begin{aligned} \overline{\overline{C}}(g_1(n), g_2(n)) &= (o(g_1(n)) \cup o\Theta(g_1(n)) \cup \Theta(g_1(n))) \\ &\cup \overline{\overline{\Theta}}(g_1(n), g_2(n)) \\ &\cup (\Theta(g_2(n)) \cup \Theta\omega(g_2(n)) \cup \omega(g_2(n))). \end{aligned} \quad (178)$$

Next, using Propositions 7 and 13, we have that

$$\begin{aligned} \overline{\overline{C}}(g_1(n), g_2(n)) &= O(g_1(n)) \cup (\Omega(g_1(n)) \cap O(g_2(n))) \cup \Omega(g_2(n)). \end{aligned} \quad (179)$$

□

Proposition 32. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

- (a) $\overline{\overline{C}}(g_1(n), g_2(n)) \subseteq C(g_1(n))$;
- (b) if $g_1(n) \in \Theta(g_2(n))$, then $\overline{\overline{C}}(g_1(n), g_2(n)) = C(g_1(n))$.

Proof. (a) From Proposition 31 it follows that

$$\overline{\overline{C}}(g_1(n), g_2(n)) \subseteq O(g_1(n)) \cup \Omega(g_1(n)) \cup \Omega(g_2(n)). \quad (180)$$

Since $g_1(n) \in O(g_2(n))$, we have that

$$\Omega(g_2(n)) \subseteq \Omega(g_1(n)). \quad (181)$$

From (180), (181), and Proposition 8, we obtain that

$$\overline{\overline{C}}(g_1(n), g_2(n)) \subseteq O(g_1(n)) \cup \Omega(g_1(n)) = C(g_1(n)). \quad (182)$$

(b) Since $g_1(n) \in \Theta(g_2(n))$, one can easily prove that

$$\begin{aligned} O(g_1(n)) &= O(g_2(n)), \\ \Omega(g_1(n)) &= \Omega(g_2(n)). \end{aligned} \quad (183)$$

Using (183) and Proposition 31, we have

$$\begin{aligned} \overline{\overline{C}}(g_1(n), g_2(n)) &= O(g_1(n)) \cup (\Omega(g_1(n)) \cap O(g_1(n))) \cup \Omega(g_1(n)). \end{aligned} \quad (184)$$

Next, from Proposition 4(e) we obtain that

$$\overline{\overline{C}}(g_1(n), g_2(n)) = O(g_1(n)) \cup \Theta(g_1(n)) \cup \Omega(g_1(n)). \quad (185)$$

Consequently,

$$\overline{\overline{C}}(g_1(n), g_2(n)) = O(g_1(n)) \cup \Omega(g_1(n)). \quad (186)$$

Thus, from Proposition 8, it follows that

$$\overline{\overline{C}}(g_1(n), g_2(n)) = C(g_1(n)). \quad (187)$$

□

Proposition 33. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

- (a) $\overline{\overline{C}}(g_1(n), g_2(n)) \subseteq C(g_2(n))$;
- (b) if $g_1(n) \in \Theta(g_2(n))$ then $\overline{\overline{C}}(g_1(n), g_2(n)) = C(g_2(n))$.

Proof. The proof follows the same idea used for proving Proposition 32. □

Proposition 34. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

- (a) $\overline{\overline{C}}(g_1(n), g_2(n)) \subseteq C(g_1(n)) \cup C(g_2(n))$;
- (b) if $g_1(n) \in \Theta(g_2(n))$ then $\overline{\overline{C}}(g_1(n), g_2(n)) = C(g_1(n)) \cup C(g_2(n))$.

Proof. The proof follows easily using Propositions 32 and 33. □

Proposition 35. Let $g_1, g_2 \in \mathcal{F}$ such that $g_1(n) \in O(g_2(n))$. Then,

$$\overline{\overline{C}}(g_1(n), g_2(n)) = C(g_1(n)) \cap C(g_2(n)). \quad (188)$$

Proof. Using Proposition 8, we have that

$$\begin{aligned} & C(g_1(n)) \cap C(g_2(n)) \\ &= (O(g_1(n)) \cup \Omega(g_1(n))) \cap (O(g_2(n)) \cup \Omega(g_2(n))). \end{aligned} \quad (189)$$

From (189), using the distributivity of intersection over union, we obtain

$$\begin{aligned} & C(g_1(n)) \cap C(g_2(n)) \\ &= (O(g_1(n)) \cap (O(g_2(n)) \cup \Omega(g_2(n)))) \\ &\quad \cup (\Omega(g_1(n)) \cap (O(g_2(n)) \cup \Omega(g_2(n)))) \\ &= (O(g_1(n)) \cap O(g_2(n))) \cup (O(g_1(n)) \cap \Omega(g_2(n))) \\ &\quad \cup (\Omega(g_1(n)) \cap O(g_2(n))) \cup (\Omega(g_1(n)) \cap \Omega(g_2(n))). \end{aligned} \quad (190)$$

Next, using the relations $g_1(n) \in O(g_2(n))$ and $O(g_1(n)) \cap \Omega(g_2(n)) \subseteq O(g_1(n))$, we obtain that

$$\begin{aligned} & C(g_1(n)) \cap C(g_2(n)) \\ &= O(g_1(n)) \cup (\Omega(g_1(n)) \cap O(g_2(n))) \cup \Omega(g_2(n)). \end{aligned} \quad (191)$$

Consequently, using Proposition 31, we have

$$C(g_1(n)) \cap C(g_2(n)) = \overline{\overline{C}}(g_1(n), g_2(n)). \quad (192)$$

□

8. Conclusions

In this paper, we defined a new asymptotic notation based on two given complexity functions and then we proposed and proved several properties of this new notation, classified in the following four categories:

- (i) relations between “Weak Theta” and other asymptotic notations;
- (ii) properties concerning membership, inclusion, intersection, and union;
- (iii) some relations between complexity functions;
- (iv) properties related to “Weak Theta” and the set of complexity functions comparable with two given complexity functions.

The main benefit of this new asymptotic notation is the possibility to characterize a complexity function by catching it between two given complexity functions, thus allowing approximation of functions with complex behaviours that are hard to be analyzed using other existent asymptotic notations.

Notations

\mathbb{N}^* :	The set of positive integers
\mathbb{R}_+^* :	The set of positive real numbers
\mathcal{F} :	The set of complexity functions (see Definition 1)
M_1, M_2 :	Subsets of \mathbb{N}^*
$\Theta, O, \Omega, o, \omega$:	The main asymptotic notations used in the literature (see Definition 3)
$C, o\Theta, \Theta\omega$:	Other asymptotic notations used in this paper (see Definitions 5 and 6)
$\overline{\overline{\Theta}}$:	The new asymptotic notation weak theta defined in this paper (see Definition 9)
$f, g, h, f_1, f_2, g_1, g_2, g_3, g_4, g_1, g_2, \dots, g_k, g_i, g_{i+1}, g_{i+2}; c, c', c'', \bar{c}', c_1, c_1', c_1'', c_1''', \bar{c}_1, \bar{c}_1, c_2, c_2', c_2'', c_2''', \bar{c}_2, c_3, c_3', c_3'', c_4, c_4', c_4'', c_5;$	Complexity functions
$n_0, n_0', n_0'', n_0''', \bar{n}_0, \bar{n}_0', \bar{n}_0'', \bar{n}_0''', n_{01}, n_{01}', n_{02}, n_{02}', n_{03}, n_{04}, n_{05};$	Positive real constants
n :	Positive integer constants
n :	The argument of the complexity functions, positive integer.

Conflict of Interests

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

Acknowledgment

The work has been funded by the Sectoral Operational Programme Human Resources Development 2007–2013 of the Ministry of European Funds through the Financial Agreement POSDRU/159/1.5/S/132397.

References

- [1] S. Arora and B. Barak, *Complexity Theory: A Modern Approach*, Cambridge University Press, Cambridge, UK, 2008.
- [2] O. Goldreich, *Computational Complexity: A Conceptual Perspective*, Cambridge University Press, Cambridge, UK, 2008.
- [3] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, Cambridge, Mass, USA, 2nd edition, 2001.
- [4] D. E. Knuth, *The Art of Computer Programming, Vol. 1: Fundamental Algorithms*, Addison-Wesley, Reading, Mass, USA, 3rd edition, 1975.
- [5] R. Greenlaw and H. Hoover, *Fundamentals of the Theory of Computation: Principles and Practice*, Morgan Kaufmann Publishers, San Francisco, Calif, USA, 1998.

- [6] C. A. Giumale, *Introduction to the Analysis of Algorithms: Theory and Application*, Polirom, Bucharest, Romania, 2004 (Romanian).
- [7] A. H. Mogos and A. M. Florea, "A method to compare two complexity functions using complexity classes," *UPB Scientific Bulletin, Series A: Applied Mathematics and Physics*, vol. 72, no. 2, pp. 69–84, 2010.
- [8] J. H. Davenport, "Nauseating Notation Very Much in Draft," August 2013, <http://staff.bath.ac.uk/masjhd/Drafts/Notation.pdf>.
- [9] G. Brassard, "Crusade for a better notation," *ACM SIGACT News*, vol. 17, no. 1, pp. 60–64, 1985.
- [10] M. Akra and L. Bazzi, "On the solution of linear recurrence equations," *Computational Optimization and Applications*, vol. 10, no. 2, pp. 195–210, 1998.
- [11] S. Roura, "Improved master theorems for divide-and-conquer recurrences," *Journal of the ACM*, vol. 48, no. 2, pp. 170–205, 2001.
- [12] O. Goldreich, *P, NP, and NP-Completeness: The Basics of Complexity Theory*, Cambridge University Press, 2010.
- [13] F. D. Lewis, *Solving NP-Complete Problems*, University of Kentucky, 2013.
- [14] V. V. Vazirani, *Approximation Algorithms*, Springer, New York, NY, USA, 2003.
- [15] A. S. Tanenbaum and D. J. Wetherall, *Computer Networks*, Prentice Hall, 5th edition, 2011.
- [16] A. S. Tanenbaum, *Modern Operating Systems*, Prentice Hall, 3rd edition, 2008.
- [17] A. S. Tanenbaum and M. V. Steen, *Distributed Systems: Principles and Paradigms*, Prentice Hall, 2nd edition, 2007.
- [18] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Prentice-Hall, Upper Saddle River, NJ, USA, 3rd edition, 2009.
- [19] Y. Zhang, S. Balochian, P. Agarwal, V. Bhatnagar, and O. J. Housheya, "Artificial intelligence and its applications," *Mathematical Problems in Engineering*, vol. 2014, Article ID 840491, 10 pages, 2014.
- [20] Y. Zhang, P. Agarwal, V. Bhatnagar, S. Balochian, and X. Zhang, "Swarm intelligence and its applications 2014," *The Scientific World Journal*, vol. 2014, Article ID 204294, 4 pages, 2014.
- [21] J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence*, Morgan Kaufmann Publishers, 2001.
- [22] S.-C. Chu and P.-W. Tsai, "Computational intelligence based on the behavior of cats," *International Journal of Innovative Computing, Information and Control*, vol. 3, no. 1, pp. 163–173, 2007.
- [23] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm, a novel tool for complex optimisation problems," in *Proceedings of the 2nd International Virtual Conference on Intelligent Production Machines and Systems (IPROMS '06)*, pp. 454–459, July 2006.
- [24] 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.
- [25] Y. Zhang, S. Wang, Y. Sun, G. Ji, P. Phillips, and Z. Dong, "Binary structuring elements decomposition based on an improved recursive dilation-union model and RSAPSO method," *Mathematical Problems in Engineering*, vol. 2014, Article ID 272496, 12 pages, 2014.
- [26] M. Wooldridge, *An Introduction to MultiAgent Systems*, John Wiley & Sons, 2002.
- [27] S. Li, H. Du, and X. Lin, "Finite-time consensus algorithm for multi-agent systems with double-integrator dynamics," *Automatica*, vol. 47, no. 8, pp. 1706–1712, 2011.
- [28] Q. Liu, J. Ma, and W. Xie, "Multiagent reinforcement learning with regret matching for robot soccer," *Mathematical Problems in Engineering*, vol. 2013, Article ID 926267, 8 pages, 2013.
- [29] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, 2004.
- [30] A. Smola and S. V. N. Vishwanathan, *Introduction to Machine Learning*, Cambridge University Press, 2008.
- [31] R. Studer, S. Grimm, and A. Abecker, Eds., *Semantic Web Services: Concepts, Technologies and Applications*, Springer, New York, NY, USA, 2007.
- [32] A. H. Mogoş and A. M. Florea, "Classification and comparison of several semantic web services composition methods," in *Proceedings of the 13th International Conference on Informatics in Economy (IE '14)*, pp. 690–697, Bucharest, Romania, May 2014.
- [33] R. R. Howell, "On asymptotic notation with multiple variables," Tech. Rep. 2007-4, 2008, <http://people.cis.ksu.edu/~rhowell/asymptotic.pdf>.
- [34] A.-H. Mogos, "Three variants of the master theorem," in *Proceedings of the 19th International Conference on Control Systems and Computer Science (CSCS '13)*, pp. 162–166, Bucharest, Romania, May 2013.
- [35] N. Mondal and P. P. Ghosh, *Another Asymptotic Notation: 'Almost'*, Cornell University Library, 2013, <http://arxiv.org/pdf/1304.5617.pdf>.
- [36] Q. F. Stout, "Standard Computer Science Notation and Mathematics," 2014, <http://web.eecs.umich.edu/~qstout/notation.pdf>.
- [37] Q. F. Stout, "Isotonic regression for multiple independent variables," *Algorithmica*, 2013.
- [38] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers*, Oxford University Press, Oxford, UK, 6th edition, 2008.

Research Article

A Novel Tournament Selection Based Differential Evolution Variant for Continuous Optimization Problems

Qamar Abbas,¹ Jamil Ahmad,² and Hajira Jabeen¹

¹Computer Department, Iqra University, Islamabad 44000, Pakistan

²Computer Department, Abasyn University, Islamabad 44000, Pakistan

Correspondence should be addressed to Qamar Abbas; qamar.bhk@gmail.com

Received 19 September 2014; Revised 2 December 2014; Accepted 10 December 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Qamar Abbas 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.

Differential evolution (DE) is a powerful global optimization algorithm which has been studied intensively by many researchers in the recent years. A number of variants have been established for the algorithm that makes DE more applicable. However, most of the variants are suffering from the problems of convergence speed and local optima. A novel tournament based parent selection variant of DE algorithm is proposed in this research. The proposed variant enhances searching capability and improves convergence speed of DE algorithm. This paper also presents a novel statistical comparison of existing DE mutation variants which categorizes these variants in terms of their overall performance. Experimental results show that the proposed DE variant has significance performance over other DE mutation variants.

1. Introduction

Storn and Price [1] have proposed DE in 1995 which is a stochastic population based evolutionary algorithm to find global optima for any given function [2]. DE has been applied to variety of real life problems such as electrical power systems [3], control systems [4], electromagnetism [5], image processing [6], bioinformatics [7], signal processing [8], chemical engineering [9], and many more. Different state-of-the-art versions of DE algorithm like jDE [10], SaDE [11], JADE [12], EPSDE [13], and so forth are based on parameter selection, parameter adaption, strategy selection, and/or strategy adaption mechanisms. DE state-of-the-art proves to be very powerful and uses conventional DE strategies as a strategy selection pool/strategy adaption and/or uses parameter selection pool/parameter adaption. During the process of optimization, DE algorithm evolves population of potential solutions by exploring the entire search space overtime to locate the optima.

The DE algorithm mutation strategies are formed by the linear combination of existing population members. The trial vector and target vector forms the mutant vector in DE. DE mutation strategies can be formed by any combinations of current vector, random vector(s), better vector, and best vector. In any mutation strategy, the order, number, and

name of vector(s) are very important. The behavior of DE algorithm is influenced by the selection of mutation strategy and crossover scheme along with their control parameters: mutation probability “ F ” and crossover rate “CR.” As stated, there are many mutation strategies associated with DE algorithm but current literature does not show any concrete study where all of the popular mutation strategies are analyzed. It has been observed by the authors that some of the DE mutation strategies are more useful than the current dominating ones. Therefore, in this paper, a comparative analysis is also carried out to identify best performer DE mutation variants. Furthermore, this research also proposes a novel DE mutation variant. The proposed DE variant is based on the well known tournament selection criteria. The proposed DE mutation strategy is abbreviated as TSDE in this paper. To generate a mutant vector in DE algorithm, some parents will be selected through tournament selection criteria. The TSDE variant has the ability to increase the convergence speed since it uses the combination of best vectors and random vectors from the population. Tournament selection will be helpful to incorporate some diversity in TSDE since global best vector has less chance of selection as a parent. To generate a mutant vector, the proposed TSDE utilizes two best individuals instead of global best from the whole population which makes it more convenient to escape from the local optima problem.

The most state-of-the-art DE algorithms focuses on basic DE mutation strategies and the incorporation of this proposed TSDE mutation variant will be helpful in improving the performance of DE algorithm.

The rest of the paper is organized into the following sections. Section 2 presents literature survey. In Section 3, the DE algorithm operators are discussed. In Section 4, the proposed TSDE variant is presented. Section 5 presents test functions and parameter setting details. In Section 6, experimental results along with discussion are given. Section 7 contains statistical analysis. Section 8 compares proposed TSDE with other well known heuristic algorithms. Finally, conclusion and future work are presented in Section 9.

2. Literature Survey

As stated above, many state-of-the-art versions of the DE algorithms are developed, for example, jDE, SaDE, JADE, and DEGL. Variations in the mutation strategies of DE algorithm have also been the result of intensive research in this area. Qu and Suganthan [29] have introduced the concept of ensemble restricted tournament selection in DE algorithm. They have selected a set of w population members and calculated the Euclidean distance of w population members from the current individual. The member having smallest distance from current individual is selected as a tournament winner and it became the current individual if its fitness is also better. Epitropakis et al. [30] have introduced two new mutation strategies that incorporate spatial information of neighbors to form new population individual. Their new mutation strategies “*DE/nrand/1*” and “*DE/nrand/2*” corresponding to two well known mutation strategies “*DE/rand/1*” and “*DE/rand/2*” are important part of the current literature. Qiu et al. [31] have introduced the concept of tournament selection criteria in DE algorithm. They have selected all individuals used in their mutation strategy by using the concept of tournament selection by taking a tournament of size 3. Their research work is applied on “*DE/rand/1*,” “*DE/rand/2*,” “*DE/current to best/2*,” and “*DE/rand to best/2*” DE mutation strategies which proved to be competitive to original DE algorithm and also enhanced jDE algorithm. Neighbourhood based mutation strategy for DE algorithm was introduced by Das et al. [32]. This strategy implements the concepts of local model that works on small neighbourhood of each individual while global model mutates each individual using the neighbourhood concept of entire population. The concept of neighbourhood is implemented on a commonly used DE mutation strategy “*DE/target-to-best/1/bin*.” Sutton et al. [33] have introduced the concept of focus search by imposing a selective pressure for DE mutation strategies. The selective pressure is implemented by using the concept of rank based selection of vectors used in generating donor vector in DE mutation. A bias is imposed on selecting a parent using rank based selection criteria that is based on the fitness of population members. Epitropakis et al. [34] have introduced the concept of proximity based mutation operator in DE algorithm. Their new framework utilizes the information of neighbours in evolving the population. Each population member is assigned a selection probability

that is inversely proportional to the distance from mutated vector. Proximity based mutation is applied to “*DE/rand/1*,” “*DE/best/1*,” “*DE/rand/2*,” “*DE/best/2*,” “*DE/rand-to-best/1*,” and “*DE/rand-to-best/2*” mutation strategies. Research result shows the significance of proximity based mutation operator in DE algorithm.

DE algorithm has a number of conventional mutation strategies where the performances of these strategies may differ from each other. Zaharie [35] has analyzed the influence of crossover variant on the choice of control parameters of the DE algorithm. He has used “*DE/rand/1*” for different parameter settings of population size, mutation probability, and the crossover rate by binomial and exponential schemes. Babu and Munawar [36] have applied various DE strategies for the optimal design of shell and tube heat problem which performed better than “*DE/rand*” for the same problem. They have used different DE variants for the selection of optimal design using various control parameter settings. They conclude that “*DE/best*” strategies have better performance than “*DE/rand*” for shell and tube heat problem. Jeyakumar and Velayutham [24] present empirical performance analysis of DE variants for unconstrained global optimization problems where they have shown that binomial variants have superior performance compared to the exponential variants. Mezura-Montes et al. [37] have performed comparative study of DE mutation variants for continuous function optimization application. They found through the empirical study that “*DE/best/1/bin*” has produced good results. Ali et al. [38] have compared Mixed Strategies Differential Evolution (MSDE) with the five different DE variants for function optimization. They have shown that performance of MSDE is better than five variants of DE algorithm. Mutation operators are also used in other well known algorithms like PSO [39].

Qin et al. [11] introduced both control parameter adaption and strategy adaption mechanisms in DE algorithm (SaDE). In strategy adaption scheme, they used a strategy candidate pool of four conventional strategies: “*DE/rand/1/bin*,” “*DE/rand/2/bin*,” “*DE/rand-to-best/2/bin*,” and “*DE/current-to-rand/1*.” Each target vector generates a trial vector based on the learning period (LP) over previous generations based on the success rate. In parameter adaption, SaDE adjusts control parameter CR based on the median value of CR that is calculated based on previous CR values that have successfully generated trial vector. Zhang and Sanderson [12] introduced new parameter adaption method and have used “*DE/current to pbest/1*” conventional strategy that is based on “*DE/current to best/1*.” Mallipeddi et al. [13] introduced ensemble based crossover and mutation DE strategy (EPSDE) and their corresponding control parameter scheme. They have used a pool of different crossover and mutation strategies and a pool of values for each associated control parameter. Each target vector generates a trial vector based on the assigned strategy and the parameter values. Successful combination of the mutation and crossover strategy and associated parameters values are stored in the pool. EPSDE uses “*DE/current-to-rand/1/bin*” and JADE mutation strategy along with binomial and exponential crossover. Islam et al. [40] have introduced modified mutation strategy,

TABLE 1: List of DE mutation strategies.

Mutation strategy number	Variant name	Binomial/exponential Equation
V_1	DE/rand/1 [1]	$v_g^i = x_g^{r_1} + F(x_g^{r_2} - x_g^{r_3})$
V_2	DE/best/1 [14]	$v_g^i = x_g^{\text{best}} + F(x_g^{r_1} - x_g^{r_2})$
V_3	DE/rand/2 [14]	$v_g^i = x_g^{r_1} + F(x_g^{r_2} - x_g^{r_3}) + F(x_g^{r_4} - x_g^{r_5})$
V_4	DE/best/2 [14]	$v_g^i = x_g^{\text{best}} + F(x_g^{r_1} - x_g^{r_2}) + F(x_g^{r_3} - x_g^{r_4})$
V_5	DE/current to rand/1 [15]	$v_g^i = x_g^i + F(x_g^{r_1} - x_g^i) + F(x_g^{r_2} - x_g^{r_3})$
V_6	DE/Current-to-rand/1 [16]	$v_g^i = x_g^{r_1} + F(x_g^{r_2} - x_g^i) + F(x_g^{r_1} - x_g^{r_3})$
V_7	DE/current to best/1 [1] & DE/rand to best/1 [17] & DE/current to best/2 [18] & DE/rand to best/2 [19]	$v_g^i = x_g^i + F(x_g^{\text{best}} - x_g^i) + F(x_g^{r_1} - x_g^{r_2})$
V_8	DE/current to best/1 [20] & DE/rand to best/1 [21]	$v_g^i = x_g^i + F(x_g^{\text{best}} - x_g^{r_1}) + F(x_g^{r_1} - x_g^{r_2})$
V_9	DE/rand to best/1 [22] & DE/rand to best/2 [20]	$v_g^i = x_g^{r_1} + F(x_g^{\text{best}} - x_g^{r_2}) + F(x_g^{r_3} - x_g^{r_4})$
V_{10}	DE/rand to best/1 [23]	$v_g^i = x_g^{r_1} + F(x_g^{\text{best}} - x_g^{r_1}) + F(x_g^{r_2} - x_g^{r_3})$
V_{11}	DE/rand to best/1 [24]	$v_g^i = x_g^{r_1} + F(x_g^{\text{best}} - x_g^i) + F(x_g^{r_2} - x_g^{r_3})$
V_{12}	DE/current to best/2 [15] & DE/rand to best/2 [25]	$v_g^i = x_g^i + F(x_g^{\text{best}} - x_g^i) + F(x_g^{r_1} - x_g^{r_2}) + F(x_g^{r_3} - x_g^{r_4})$
V_{13}	DE/current to rand/2 [15]	$v_g^i = x_g^i + F(x_g^{r_1} - x_g^i) + F(x_g^{r_2} - x_g^{r_3}) + F(x_g^{r_4} - x_g^{r_5})$
V_{14}	DE/rand to best/2 [26]	$v_g^i = x_g^{r_1} + F(x_g^{\text{best}} - x_g^i) + F(x_g^{r_2} - x_g^{r_3}) + F(x_g^{r_4} - x_g^{r_5})$
V_{15}	DE/rand to best/2 [26]	$v_g^i = x_g^{r_1} + F(x_g^{\text{best}} - x_g^{r_1}) + F(x_g^{r_2} - x_g^{r_3}) + F(x_g^{r_4} - x_g^{r_5})$
V_{16}	DE/rand to current/2 [27]	$v_g^i = x_g^{r_1} + F(x_g^{r_2} - x_g^i) + F(x_g^{r_3} - x_g^{r_4})$
V_{17}	DE/rand to best and current/2 [27]	$v_g^i = x_g^{r_1} + F(x_g^{\text{best}} - x_g^{r_2}) + F(x_g^{r_3} - x_g^i)$
V_{18}	DE/mid-to-better/1 [28]	$v_g^i = F(x_g^{\text{better}} + x_g^i)/2 + F(x_g^{\text{better}} - x_g^i) + F(x_g^{r_1} - x_g^{r_2})$
V_{19}	DE/rand/3 [27]	$v_g^i = x_g^{r_1} + F(x_g^{r_2} - x_g^{r_3}) + F(x_g^{r_4} - x_g^{r_5}) + F(x_g^{r_6} - x_g^{r_7})$
V_{20}	DE/best/3 [27]	$v_g^i = x_g^{\text{best}} + F(x_g^{r_1} - x_g^{r_2}) + F(x_g^{r_3} - x_g^{r_4}) + F(x_g^{r_5} - x_g^{r_6})$

a modification in conventional binomial crossover scheme, and a parameter adaption scheme in DE algorithm. They have used “DE/current-to-gr_best/1” and call it a less greedy version of “DE/current-to-best/1.” Their mutation strategy “DE/current-to-gr_best/1” uses best vector from $q\%$ population of individuals to generate the trial vector for each target vector. In pbest crossover mutant vector can swap p -top ranked individuals of current generation instead of current parent using binomial crossover scheme. In parameter adaption scheme scale factor adaption is based on the Cauchy distribution and crossover probability adaption is based on the Gaussian distribution. Wang et al. [41] have introduced composite DE (CoDE) variant. In this scheme, they have used a pool of three trial vector generation strategies and a pool of three parameter setting combinations. The trial vector strategies used are “rand/1/bin,” “rand/2/bin,” and “current-to-rand/1” while the parameter setting combinations are [$F = 1.0$, $Cr = 0.1$], [$F = 1.0$, $Cr = 0.9$], and [$F = 0.8$, $Cr = 0.2$]. To generate a new solution in CoDE, each strategy is coupled with a randomly chosen parameter setting. Gong et al. [42] have introduced a new strategy adaptation mechanism (SaM) in their research work. They have combined SaM with JADE and named it as SaJADE. In this strategy, parameter $\eta_i \in [0, 1]$ is used to control the selection of any strategy from the strategy pool. They have chosen four various DE

strategies “DE/current-to-pbest” without archive, “DE/rand-to-pbest” without archive, “DE/current-to-pbest” with archive, and “DE/rand-to-pbest” with archive to form a strategy pool.

Table 1 shows different binomial and exponential mutation strategies used with algorithm. Current literature shows variation in naming in these strategies, which is already indicated in Table 1; for example, Four different names are used in the literature for mutation strategy V_7 . Throughout this paper, x_i denotes the target vector (or current vector), u_i represents the trial vector, and v_i as a mutant vector. In this paper, $x_g^{r_k}$ states the r_k th random vector for g th generation, v_g^i will refer to i th component of donor vector at g th generation, x_g^{best} states the best vector at g th generation, and x_g^i refers to current vector at g th generation.

3. DE Algorithm

DE algorithm has three different parameters: a population of size NP, crossover control parameter CR, and difference vector amplification parameter F . Each population member in DE is represented as a D -dimensional parameter vector. In DE algorithm, population is initialized randomly which is supposed to cover the entire search space. Each vector in the DE is represented by $x_{i,G}$, where $i = 1, 2, 3, \dots, NP$ and

G is generation number. New offspring in DE algorithm are generated by mutation, crossover, and selection operators. Three different terminologies of vectors donor vector, trial vector, and target vector are used in DE algorithm. Donor vector is a vector that is created in the mutation operation, trial vector is created in the crossover operation, and target vector is the current vector of population. The detail of these operators with index $i = 1, 2, 3, \dots, NP$, $j = 1, 2, 3, \dots, D$ is as follows.

Mutation. In mutation operation, mutant vector also called donor vector is created. Donor vector $v_{i,G+1}$ of i th population member is calculated by adding the weighted difference of two vectors to third vector:

$$v_{i,G+1} = x_{r_1,G} + F(x_{r_2,G} - x_{r_3,G}), \quad (1)$$

where random indices $r_1, r_2, r_3 \in \{1, 2, 3, \dots, NP\}$, $i \neq r_1 \neq r_2 \neq r_3$, and F is mutation probability parameter.

Crossover. DE crossover strategies control the number of inherited components from the mutant vector to form a target vector. Binomial and exponential are main crossover schemes. The DE crossover rate parameter (CR) influences the size of perturbation of the base (target) vector to ensure the population diversity [43]. Following are the binomial and exponential crossover schemes.

Binomial Crossover. In crossover operation of DE algorithm, a trial vector is formed. In binomial crossover scheme, the trial vector $u_{i,G} = (u_{i,1,g}, u_{i,2,g}, \dots, u_{i,D,g})$ is generated by the following equation:

$$u_{i,G} = \begin{cases} v_{i,j,G} & \text{if } (\text{rand}_j(0, 1) \leq \text{CR or } j = j_{\text{rand}}) \\ x_{i,j} & \text{otherwise,} \end{cases} \quad (2)$$

where j_{rand} is a randomly chosen integer in the range $[1, D]$, $\text{rand}_j(0, 1)$ is a random number in $(0, 1)$, and $v_{i,j,G}$ is the donor vector. CR is crossover control parameter in the range $\text{CR} \in (0, 1)$. Due to the range of j_{rand} , $u_{i,G}$ is always different from $x_{i,j}$ and index $i = 1, 2, 3, \dots, NP$, $j = 1, 2, 3, \dots, D$.

Exponential Crossover. In exponential crossover scheme, the trial $u_{i,G} = (u_{i,1,g}, u_{i,2,g}, \dots, u_{i,D,g})$ is created as follows:

$$u_{i,G} = \begin{cases} v_{i,j,G} & \text{for } j = \langle l \rangle_D + \langle l + 1 \rangle_D + \dots + \langle l + L - 1 \rangle_D \\ x_{i,j} & \text{otherwise,} \end{cases} \quad (3)$$

where $i = 1, 2, 3, \dots, NP$, $j = 1, 2, 3, \dots, D$, and $\langle \cdot \rangle_D$ denotes the modulo function with modulus D . The starting index l is chosen at random from $[1, D]$. L is also a randomly generated number from $[1, D]$. The parameters l and L are regenerated for each trial vector $u_{i,G}$.

Selection. In DE algorithm, new population members are formed using selection operation. Selection operator uses the greedy approach by comparing fitness of trial vector $u_{i,G+1}$ with the fitness of target vector $x_{i,G}$; the vector having

best fitness is selected as a member of new population. For selection, the following equation is used:

$$x_{i,G+1} = \begin{cases} u_{i,G+1} & \text{if } (\text{fitness}(u_{i,G+1}) < \text{fitness}(x_{i,G})) \\ x_{i,G} & \text{otherwise,} \end{cases} \quad (4)$$

where $\text{fitness}(\cdot)$ function calculates the fitness value of objective function.

There are several DE algorithm mutation variants/strategies that are formed by the linear combination of existing population members. The trial vector and target vector form the mutant vector in DE. Throughout this paper, x_i denotes the target vector (or current vector), i is the running index, u_i represents the trial vector, and v_i is a mutant vector. In DE algorithm, different mutation schemes are used to create the trial vector by using any combination of current, best, and random vectors. The behavior of DE algorithm is influenced by the selection of mutation strategy and crossover scheme along with their control parameters: mutation probability “ F ” and crossover rate “CR” [44].

Difference vector (DV) is the difference of two mutating vectors that is used to form offspring in the population [45]. To form the mutant vector in DE, some researcher uses a random value $\lambda \in (0, 1)$ as a coefficient multiplier with the first difference vector [1] and mutation probability “ F ” as a coefficient multiplier with the other difference vector(s) [17]. Some researchers have used only “ F ” as a coefficient multiplier with the difference vector(s) to form the mutant vector [44]. To reduce the number of control parameters of DE algorithm, we use $\lambda = F$ [28]. A list of binomial and exponential mutation strategies that are used in this research is given in Table 1.

4. Proposed Tournament Selection Based DE (TSDE) Variant

Tournament selection is one of the famous selection approaches used in genetic algorithms [46]. The parameter associated with this scheme is the size of tournament that selects number of individuals to participate in the competition of tournament. To decrease the risk of premature convergence, the loss of diversity should be kept as low as possible in the population [47]. To enhance the performance of population based algorithms, various researches have used tournament selection in their research work [48, 49].

The DE algorithm sometimes faces the problem of slow and/or premature convergence [32]. Exploration and exploitation are very important aspects that are helpful in improving the convergence acceleration and solution quality of evolutionary algorithms [50]. The mutation strategies “DE/rand/1” and “DE/rand/2” are helpful for exploration [11] and “DE/best/1,” “DE/rand-to-best,” and “DE/best/2” are more helpful for exploitation than exploration due to less population diversity [14]. To balance the exploration and exploitation ability of proposed TSDE variant, both random and best vectors are used.

A novel DE mutation strategy based on the selection of parent is introduced in this research work. The new mutation

strategy utilizes the knowledge of some best performing vectors and some random vectors in creating the mutant vector. Like mutation strategies “DE/rand/1” and “DE/rand/2,” the proposed mutation strategy uses random vector as a base vector along with two difference vectors. To generate a mutant vector in DE, each difference vector utilizes the knowledge of one distinct best vector that will be helpful in incorporating diversity and improving convergence in the proposed mutation strategy. The two difference vectors and two distinct best vectors selected through tournament selection criteria will be helpful in balancing the exploration and exploitation.

To minimize the loss diversity, a tournament of small size is selected so that less fit individuals may have a chance of selection as a parent. TSDE generates two best vectors using tournament selection mechanism. The selection of parents is based on the tournament selection criteria by taking a tournament of size 3 of randomly selected population members. The equation of the proposed variant TSDE is as follows:

$$v_g^i = x_g^{r_1} + F(x_g^{\text{best}_1} - x_g^{r_1}) + F(x_g^{\text{best}_2} - x_g^{r_2}). \quad (5)$$

The equation of this variant uses two best vectors $x_g^{\text{best}_1}$ and $x_g^{\text{best}_2}$ and two random vectors $x_g^{r_1}$ and $x_g^{r_2}$ that are selected from the current population. Vectors $x_g^{\text{best}_1}$ and $x_g^{\text{best}_2}$ are selected by using tournament selection mechanism by taking tournament of any size, say k . Vector $x_g^{\text{best}_1}$ is the best individual from first tournament and vector $x_g^{\text{best}_2}$ is the second best individual from the second tournament. The vectors used in each tournament are selected randomly from the current population.

The proposed TSDE variant has the ability to increase the convergence speed since it uses the combination of best vectors and random vectors from the population. Tournament selection may incorporate some diversity in TSDE since global best vector has less chance of selection as a parent because tournament of small size is used instead of whole population. Selection of two best individuals instead of global best from the whole population makes it more opportune to escape from the local optima problem that accordingly increases the solution quality and convergence speed of DE algorithm. The most state-of-the-art DE algorithm focuses on basic DE mutation strategies and the incorporation of this proposed mutation strategy will be helpful in improving the performance of DE algorithm.

The proposed mutation strategy has the following features.

- (1) A random number will be used as a base vector since mutation strategies “DE/rand/1” and “DE/rand/2” use random base vectors that maintain better exploration ability but have slow convergence speed [11].
- (2) Although DE mutation strategies “DE/best/1,” “DE/rand-to-best,” and “DE/best/2” have fast convergence [14], these are biased towards global best value that may have a chance to stick in local optima problem [11]. So to improve the convergence speed of proposed

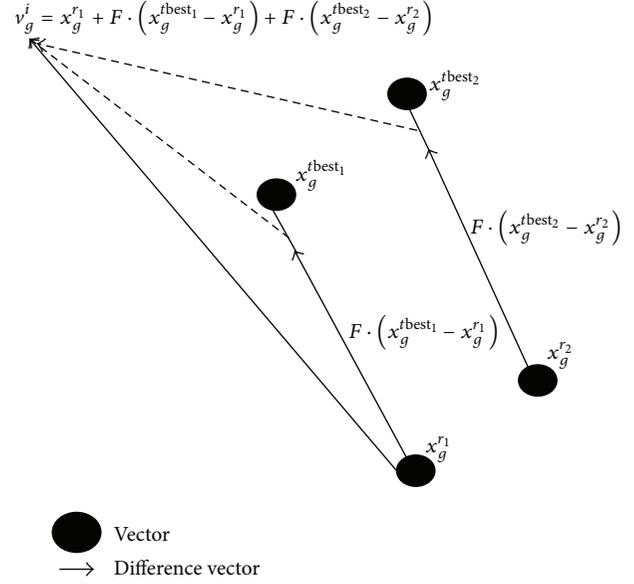


FIGURE 1: Graphical view of proposed TSDE.

TSDE, two best vectors (best_1 , best_2) are selected using tournament selection criteria.

- (3) The equation of proposed mutation strategy is

$$v_g^i = x_g^{r_1} + F(x_g^{\text{best}_1} - x_g^{r_1}) + F(x_g^{\text{best}_2} - x_g^{r_2}), \quad (6)$$

where F is scaling factor of each difference vector. The motivation of this mutation strategy is an existing DE variant “DE/rand to best/1” [23, 51] that places perturbation of random vector towards global best vector and contains two difference vectors with equation

$$v_g^i = x_g^{r_1} + F(x_g^{\text{best}_1} - x_g^{r_1}) + F(x_g^{r_2} - x_g^{r_3}). \quad (7)$$

Two best vectors (best_1 , best_2) in TSDE will be helpful in escaping local optima problem and will be helpful in improving the solution quality.

- (4) Two difference vectors are used where each difference vector utilizes one best vector selected through tournament criteria, since mutation strategies having two difference vectors (DV) produce better perturbation mode than one difference vector mutation strategies [41].

The members of tournament are selected randomly which almost makes it equal chance of selection of higher or lower performing individuals from the population. This variant maintains randomness since members of the tournament are selected randomly and we select the best performing amongst the tournament to ignore the poor performing individuals. The proposed variation is simple since it uses two best individuals selected using the tournament selection technique and other random vector selected randomly from the current population. The performance of TSDE is compared with other DE variants in Section 6.

A graphical representation of proposed mutation strategy is shown in Figure 1, where mutant vector v_g^i is generated

```

(1) Generate the initial population  $P_G = \{X_{1,G}, \dots, X_{NP,G}\}$  for generation  $G = 0$  and randomly initialize
each population member  $X_{i,G} = \{x_{i,G}^1, \dots, x_{i,G}^D\}$  where  $i = 1, \dots, NP$ 
(2) FOR  $i = 1$  to NP
    Calculate fitness  $f(X_{i,G})$  for each population member  $X_{i,G}$ 
END FOR
(3) WHILE the stopping criterion is not true
    /* Start of TSDE vectors selection */
    Step 3.1. TSDE vectors selection
    FOR  $n = 1$  to number of TSDE vectors
        FOR  $k = 1$  to Tournament_size
            Select  $k$ th tournament member with its fitness randomly from current population
        END FOR
        Select best of best member from the current tournament as nth tbest
        Return nth member index to be used as one of TSDE vectors in proposed mutation strategy
    END FOR
    /* End of TSDE vectors selection */
    Step 3.2. Mutation Step
    FOR  $i = 1$  to NP
        For the  $i$ th target vector  $X_{i,G}$  generate a donor vector  $V_{i,G} = \{v_{i,G}^1, \dots, v_{i,G}^D\}$  with the
        specified mutation strategy (From Table 1 strategies or proposed TSDE strategy)
    END FOR
    Step 3.3. Crossover Step
    FOR  $i = 1$  to NP
        For the  $i$ th target vector  $X_{i,G}$  generate a trial vector  $U_{i,G} = \{u_{i,G}^1, \dots, u_{i,G}^D\}$  with the
        specified crossover scheme (Equation (2) or Equation (3))
    END FOR
    Step 3.4. Selection Step
    FOR  $i = 1$  to NP
        Evaluate the trial vector  $U_{i,G}$  against the target vector  $X_{i,G}$  with fitness function  $f$ 
        IF  $f(U_{i,G}) \leq f(X_{i,G})$ , THEN  $X_{i,G+1} = U_{i,G}$ ,  $f(X_{i,G}) = f(U_{i,G})$ 
            IF  $f(U_{i,G}) \leq f(X_{best,G})$ , THEN  $X_{best,G+1} = U_{i,G}$ ,  $f(X_{best,G}) = f(U_{i,G})$ 
        END IF
    END FOR
    Step 3.5. increment generation number  $G = G + 1$ 
END WHILE

```

ALGORITHM 1: Pseudocode of tournament selection based DE algorithm (TSDE).

using four vectors, $x_g^{tbest_1}$, $x_g^{tbest_2}$, $x_g^{r_1}$, and $x_g^{r_2}$, shown in filled circles and directed arrow shows difference vector.

Algorithm 1 shows the pseudocode of the proposed DE algorithm (TSDE). The proposed method is also implemented through computer simulation which will be discussed in a later part in the paper.

5. Test Functions and Parameter Setting

In order to evaluate the performance of the proposed variant and existing variants of DE algorithm, a comprehensive set of 37 N -dimensional benchmark functions is used. These benchmark functions are commonly used for multidimensional global optimization problems. These functions are given in Table 2 along with the equation, search space, dimension, and optimum value of each function. N -dimensional functions have been used for extensive comparison of DE algorithm variants for various dimensions.

The control parameters of the DE algorithm are N_p (population size), D (dimension), F (scaling factor), and CR (crossover control parameter). Different researchers have used different F and CR parameter values for function optimization of N -dimensional functions. Like Noman and

Iba [52] have used $F = 0.9$, $CR = 0.9$, Brest et al. [10] have used $F = 0.5$, $CR = 0.9$; Ali and Torn [53] have used $CR = 0.5$, $F \in [0.4, 1]$; Piotrowski and Napiorkowski [54] have used $CR = 0.5$ and $F = 0.5$; Dong et al. [55] have used $F = 0.7$ and $CR = 0.5$ in their research work. In this research work, we have used control parameters $F = 0.7$, $CR = 0.5$ and population size $N_p = 30$. The dimensions size used is 10, 20, and 30 and the numbers of training iterations for average fitness value are 5000, 10000, and 15000 for 10D, 20D, and 30D, respectively. The DE algorithm iterates over 30 independent runs and the average fitness value of these 30 runs is used in the results. Same parameter values are used for all functions and all mutation strategies. Experimental results of number of function calls (NFC) are generated for maximum NFC $10^4 * DIM$ [56]. To find out the number of function calls, VTR is set to 0.0001 and Max-NFC for 10D, 20D, and 30D is, respectively, 100,000, 200,000, and 300,000 for all functions and all mutation strategies.

6. Experimental Results and Discussion

This section presents simulation results and its comparative analysis for DE mutation strategies given in Table 1 using

TABLE 2: Test benchmark function.

Function	Name of function (type)	Equation	Search space	Optima
f_1	Sphere model (separable, multimodal)	$f(x) = \sum_{i=0}^n x_i^2$	$-5.12 \leq x_i \leq 5.12$	0
f_2	Axis parallel hyperellipsoid (separable, unimodal)	$f(x) = \sum_{i=0}^n i \cdot x_i^2$	$-5.12 \leq x_i \leq 5.12$	0
f_3	Schwefel's problem 1.2 (nonseparable, unimodal)	$f(x) = \sum_{i=0}^n \left(\sum_{j=0}^i x_j \right)^2$	$-65 \leq x_j \leq 65$	0
f_4	Rosenbrock's valley (nonseparable, unimodal)	$f(x) = \sum_{i=1}^{n-1} \left[100(x_{i+1} - x_i)^2 + (1 - x_i)^2 \right]$	$-30 \leq x_i \leq 30$	0
f_5	Rastrigin's function (Separable, Multimodal)	$f(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$	$-5.12 \leq x_i \leq 5.12$	0
f_6	Griewank's function (nonseparable, multimodal)	$f(x) = \sum_{i=1}^n \left(\frac{x_i^2}{4000} - \prod_{i=1}^n \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \right)$	$-600 \leq x_i \leq 600$	0
f_7	Sum of different power (nonseparable, multimodal)	$f(x) = \sum_{i=1}^n x_i ^{(i+1)}$	$-1 \leq x_i \leq 1$	0
f_8	Ackley's path function (nonseparable, multimodal)	$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{\sum_{i=1}^n x_i^2}{n}}\right) - \exp\left(\frac{\sum_{i=1}^n \cos(2\pi x_i)}{n}\right) + 20 + e$	$-32 \leq x_i \leq 32$	0
f_9	Levy function (separable, multimodal)	$0.1 \left[\sin^2(3\pi x_1) + \sum_{i=1}^{n-1} (x_i - 1)^2 \times (1 + \sin^2(3\pi x_i + 1)) + (x_n - 1) (1 + \sin^2(2\pi x_n)) \right]$	$-10 \leq x_i \leq 10$	0
f_{10}	Zakharov function (nonseparable, multimodal)	$f(x) = \sum_{i=1}^n x_i^2 + \left(\sum_{i=1}^n 0.5i x_i \right)^2 + \left(\sum_{i=1}^n 0.5i x_i \right)^4$	$-5 \leq x_i \leq 10$	0
f_{11}	Schwefel's problem 2.22 (nonseparable, unimodal)	$f(x) = \sum_{i=1}^n x_i + \prod_{i=1}^n x_i $	$-10 \leq x_i \leq 10$	0
f_{12}	Step function (separable, unimodal)	$f(x) = \sum_{i=1}^n (1 x_i + 0.5)^2$	$-100 \leq x_i \leq 100$	0
f_{13}	De Jong's function-4 (no noise) (separable, unimodal)	$f(x) = \sum_{i=1}^n i x_i^4$	$-1.28 \leq x_i \leq 1.28$	0
f_{14}	Alpine function (separable, multimodal)	$f(x) = \sum_{i=1}^n x_i \sin(x_i) + 0.1x_i $	$-10 \leq x_i \leq 10$	0
f_{15}	Levy and Montalvo Problem (separable, multimodal)	$f(x) = \left(\frac{\pi}{n}\right) \left(10 \sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] \right) + (y_n - 1)^2$ where $y_i = 1 + \frac{1}{4}(x_i + 1)$	$-10 \leq x_i \leq 10$	0

TABLE 2: Continued.

Function	Name of function (type)	Equation	Search space	Optima
f_{16}	Neumaier 2 Problem (separable, unimodal)	$f(x) = \sum_{i=1}^n (x_i - 1)^2 - \sum_{i=2}^n (x_i x_{i-1})$	$-n^2 \leq x_i \leq n^2$	0
f_{17}	Cosine Mixture (separable, multimodal)	$f(x) = -0.1 \sum_{i=1}^n \cos(5\pi x_i) + \sum_{i=1}^n x_i^2 \quad \ x\ = \sqrt{\sum_{i=1}^n x_i^2}$	$-1 \leq x_i \leq 1$	$-0.1x(n)$
f_{18}	Cigar (separable, multimodal)	$f(x) = x_1^2 + 100000 \sum_{i=1}^n x_i^2$	$-10 \leq x_i \leq 10$	0
f_{19}	Function "15" (separable, multimodal)	$f(x) = \sum_{i=1}^{n-1} [0.2x_i^2 + 0.1x_i^2 \sin(2x_i)]$	$-10 \leq x_i \leq 10$	0
f_{20}	Ellipse Function (separable, unimodal)	$f(x) = \sum_{i=1}^n (10^{6((i-1)/(n-1))} \cdot x_i^2)$	$-100 \leq x_i \leq 100$	0
f_{21}	Tablet Function (separable, unimodal)	$f(x) = 10^4 x_1^2 + \sum_{i=2}^n x_i^2$	$-100 \leq x_i \leq 100$	0
f_{22}	Schewel (separable, multimodal)	$f(x) = \sum_{i=1}^n ((x_i - x_i^2)^2 + (x_i - 1)^2)$	$-32 \leq x_i \leq 32$	0
f_{23}	Deflected corrugated spring (separable, multimodal)	$f(x) = 0.1 \sum_{i=1}^n \left((x_i - \alpha)^2 - \cos \left(K \sqrt{\sum_{i=1}^n (x_i - \alpha)^2} \right) \right)$	$0 \leq x_i \leq 10$ $K = 5$ $\alpha = 5$ $x_i \in [0, 2\alpha]$	0
f_{24}	Mishra 1 global optimization problem (nonseparable, multimodal)	$f(x) = (1 + x_n)^{x_n} \text{ where } x_n = n - \sum_{i=1}^{n-1} x_i$	$0 \leq x_i \leq 1$	2
f_{25}	Mishra 2 global optimization problem (nonseparable, multimodal)	$f(x) = (1 + x_n)^{x_n} \text{ where } x_n = n - \sum_{i=1}^{n-1} \frac{(x_i + x_{i+1})}{2}$	$0 \leq x_i \leq 1$	2
f_{26}	MultiModal global optimization problem (separable, multimodal)	$f(x) = \left(\sum_{i=1}^n x_i \right) \left(\prod_{i=1}^n x_i \right)$	$-10 \leq x_i \leq 10$	0
f_{27}	Quintic global optimization problem (separable, multimodal)	$f(x) = \sum_{i=1}^n x_i^5 - 3x_i^4 + 4x_i^3 + 2x_i^2 - 10x_i - 4 $	$-10 \leq x_i \leq 10$	-1
f_{28}	Stochastic global optimization problem (separable, multimodal)	$f(x) = \sum_{i=1}^n \varepsilon_i \left x_i - \frac{1}{i} \right $	$-5 \leq x_i \leq 5$	0
f_{29}	Stretched V global optimization problem (nonseparable, multimodal)	$f(x) = \sum_{i=1}^{n-1} t^{1/4} \left[\sin(50t^{0.1}) + 1 \right]^2 \text{ where } t = x_{i+1}^2 + x_i^2$	$-10 \leq x_i \leq 10$	0
f_{30}	XinShe Yang (nonseparable, multimodal)	$f(x) = \frac{\left(\sum_{i=1}^n x_i \right)}{e^{\sum_{i=1}^n \sin(x_i^2)}}$	$-2\pi \leq x_i \leq 2\pi$	0

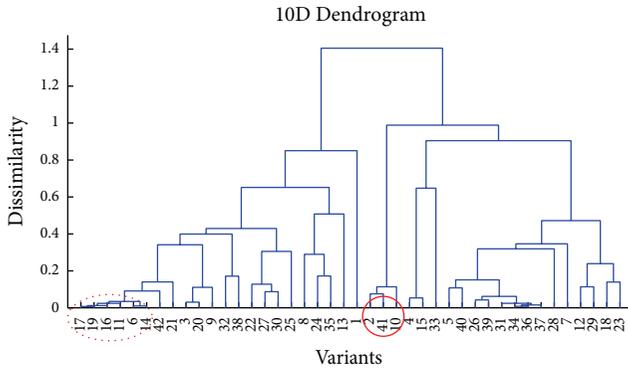


FIGURE 2: 10D: average fitness value Dendrogram of 42 variants.

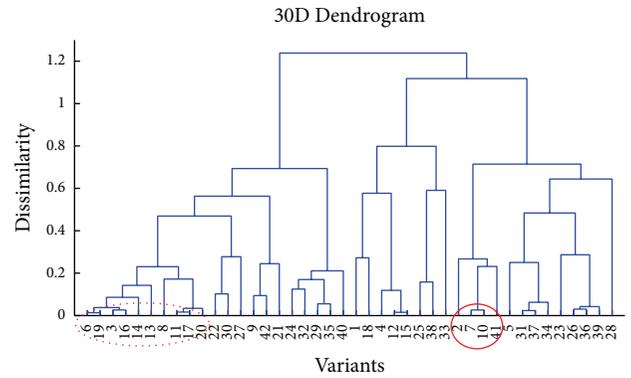


FIGURE 4: 30D: average fitness value Dendrogram of 42 variants.

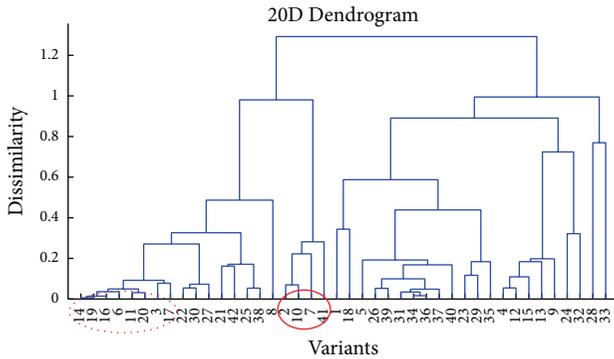


FIGURE 3: 20D: average fitness value Dendrogram of 42 variants.

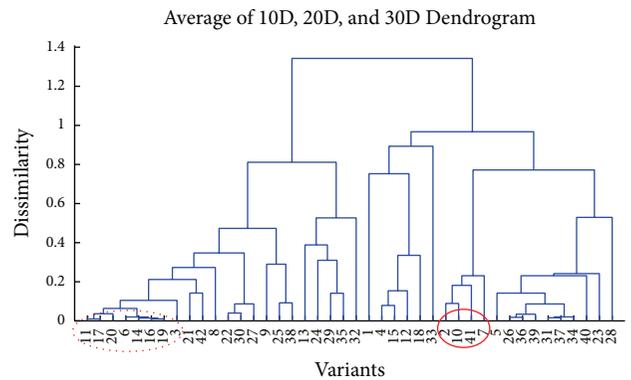


FIGURE 5: Average of average fitness value for 10D, 20D, and 30D Dendrogram of 42-DE variants.

parameter setting given in Section 5. In order to perform a fair comparison, all results are averaged over 30 independent runs. Both binomial and exponential crossover schemes are implemented for all benchmark functions given in Table 2 and DE mutation strategies given in Table 1 to obtain the results on a set of benchmark functions given in Table 2 for the DE mutation strategies given in Table 1. Both average fitness value and number of function calls (NFC) are used for performance evaluation of DE mutation strategies as used by most of the evolutionary computing algorithms [14]. First, average fitness values of the functions are considered over 30 independent runs for the performances evaluation of DE mutation strategies. For easy observation, best values of mutation strategies are given as bold face for every case. Experimental results of average fitness for all DE mutation strategies and benchmark functions are reported in Table 2. Because of space limitation, the results of binomial and exponential strategies of proposed TSDE, five most commonly used strategies [14], and one other best performing strategy are reported in Tables 3 and 4. Figures 2–9 show average fitness and NFC Dendrograms for the proposed DE mutation variant and existing DE mutation strategies. Figure 10 shows the convergence graph of binomial and exponential versions of the proposed TSDE and other selected mutation strategies. NFC experimental results for DE mutation strategies and benchmark functions are reported in Table 4.

The results of average fitness value and NFC are used to evaluate the performance of DE mutation strategies. Commonly used mutation strategies, proposed mutation

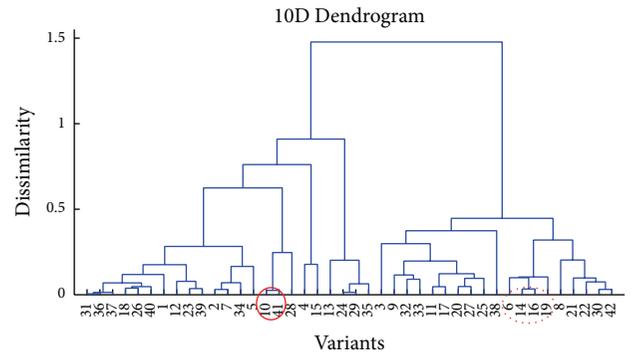


FIGURE 6: 10D: average number of function call values Dendrogram of 42-DE variants.

variant and, one other better performing mutation strategy results are included in this section since results are large enough to be managed in the tabular form for all variants in the paper. Experimental results are obtained for 40 DE mutation strategies given in Table 1 and the proposed TSDE mutation variant (V_{41} and V_{42}) but reported for selected mutation strategies only due to space issue. Experimental results are analyzed on the basis of commonly used performance parameters average fitness and NFC. The benchmark functions are N -dimensional functions having varying nature. From average fitness results, it can be observed that proposed

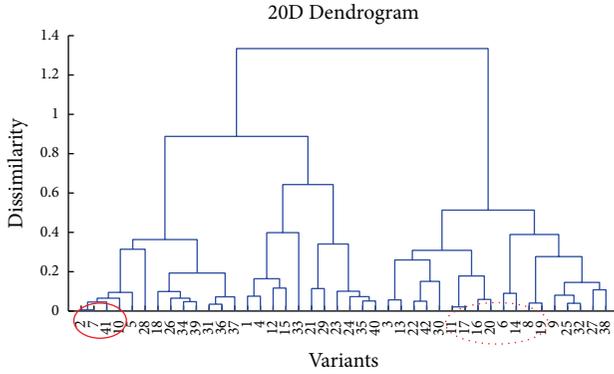


FIGURE 7: 20D: average number of function call values Dendrogram of 42 variants.

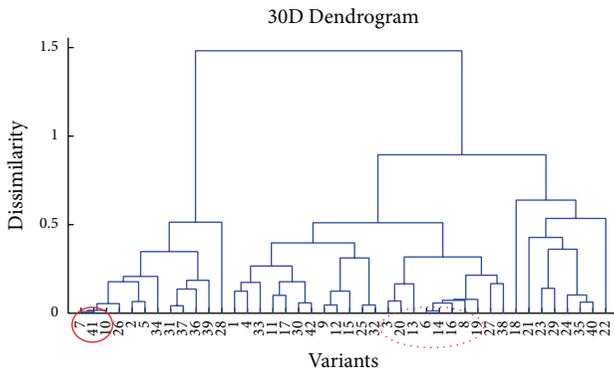


FIGURE 8: 30D: average number of function call values Dendrogram of 42-DE variants.

binomial TSDE(V_{41}) version has dominating performance among all strategies; the mutation strategy V_{10} that has never announced to be one of the best performing mutation strategies in DE has the second best performance; proposed exponential TSDE(V_{42}) and two other well known DE mutation strategies V_7 and V_2 have subsequent best performance. Although in most of the cases proposed V_{41} shares its best performance with other mutation strategies like V_{10} , V_7 , V_2 , and so forth, convergence graphs show that V_{41} has better convergence speed than these mutation strategies. Research result shows that proposed mutation strategies perform better for the functions having various characteristics like unimodal/multimodal and separable/nonseparable. The proposed mutation strategy (V_{41}) has better average fitness performance for separable functions f_1 , f_2 , f_{12} , f_{13} , f_{19} (10D, 20D), f_{16} (10D), f_{22} (10D, 20D), f_{18} , f_{19} , f_{20} , f_{21} , f_{22} (10D), f_{26} , and f_{28} ; nonseparable functions f_3 , f_7 , f_{10} , f_{11} , f_{18} , f_{24} , and f_{25} ; unimodal functions f_2 , f_3 , f_{11} , f_{12} , f_{13} , f_{16} (10D), f_{26} , and f_{21} ; and multimodal functions f_1 , f_7 , f_{10} , f_{18} , f_{19} (10D, 20D), f_{22} (10D, 20D), f_{18} , f_{19} , f_{22} (10D), f_{24} , f_{25} , f_{26} , and f_{28} . One of the rarely used DE mutation strategies V_{10} has second best average fitness performance for separable functions f_1 , f_2 , f_{12} (10D), f_{14} , f_{19} , f_{16} (10D), f_{18} , f_{24} , f_{20} , f_{21} , and f_{26} ; nonseparable functions f_7 , f_{18} , and f_{21} (10D, 30D); unimodal functions f_2 , f_{12} (10D), f_{14} , f_{16} (10D), f_{20} , and f_{21} ; and multimodal functions f_1 , f_7 , f_{18} , f_{19} , f_{21} (10D, 30D), f_{18} , f_{24} , and f_{26} . The proposed

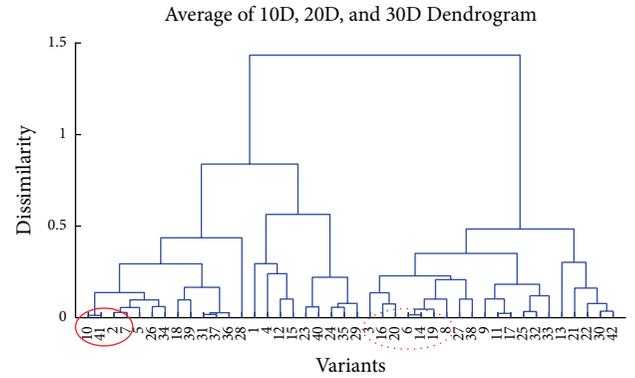


FIGURE 9: Average of average number of function call values for 10D, 20D, and 30D Dendrogram of 42-DE variants.

exponential mutation strategy (V_{42}) has better average fitness performance for separable functions f_5 (10D, 30D), f_{12} , f_{14} , f_{22} , f_{22} (10D), and f_{26} ; nonseparable functions f_6 (30D), f_{24} , f_{25} , and f_{30} ; unimodal functions f_{12} and f_{14} ; and multimodal functions f_5 (10D, 30D), f_6 (30D), f_{22} , f_{22} (10D), f_{24} , f_{25} , f_{26} , and f_{30} . Average fitness result shows that V_{41} has dominating performance among all strategies and although it shares its best performance with other mutation strategies like V_2 , V_7 , and V_{10} in most of the cases, the convergence graphs show that the proposed TSDE has fast convergence than other mutation strategies like V_2 , V_7 , and V_{10} and vice versa.

Results of number of function calls show that TSDE(V_{41}) has clearly leading performance among all other DE mutation strategies in most of the cases. The binomial TSDE mutation variant V_{41} has best performance in most of the functions having various characteristics like separable, nonseparable, unimodal, and multimodal. V_{41} outperforms other strategies for *Sphere model*, *Axis parallel hyperellipsoid*, *Schwefel's problem 1.2*, *Rosenbrock's valley*, *Griewank's function (20D, 30D)*, *Sum of different power*, *Levy function (30D)*, *Zakharov function*, *Schwefel's problem 2.22*, *Step function*, *De Jong's function-4*, *Levy and Montalvo Problem*, *Cosine Mixture*, *Cigar*, *Function "15,"* *Ellipse Function*, *Tablet Function*, *Schwefel*, *MultiModal global optimization problem*, *Quintic global optimization problem*, and *Stochastic global optimization problem*. The exponential version of TSDE mutation variant V_{42} has better performance for *Ackley's path function (30D)*, *Levy function (20D)*, and *Alpine function (20D)*. One of the commonly used mutation strategies of DE algorithm V_2 has better performance for *Griewank's function (10D)*, *Mishra-1 global optimization problem*, *Mishra-2 global optimization problem*, *Stretched-V global optimization problem (30D)*, and *XinSheYang (30D)* functions. The mutation strategies that fail to reach VTR for any function with specified parameter contain dashed values in Table 4 for the NFC results. It can be noted from the results that the proposed TSDE performs better than all others including V_{10} . Proposed TSDE can be one of the powerful mutation strategies in DE algorithm.

Experimental result can be summarized that the proposed binomial "TSDE(V_{41})" has the dominating performance among all DE mutation strategies for both average fitness values and number of function call values. The strategy V_{10} ("DE/rand repeated to best/1/bin") is the second best

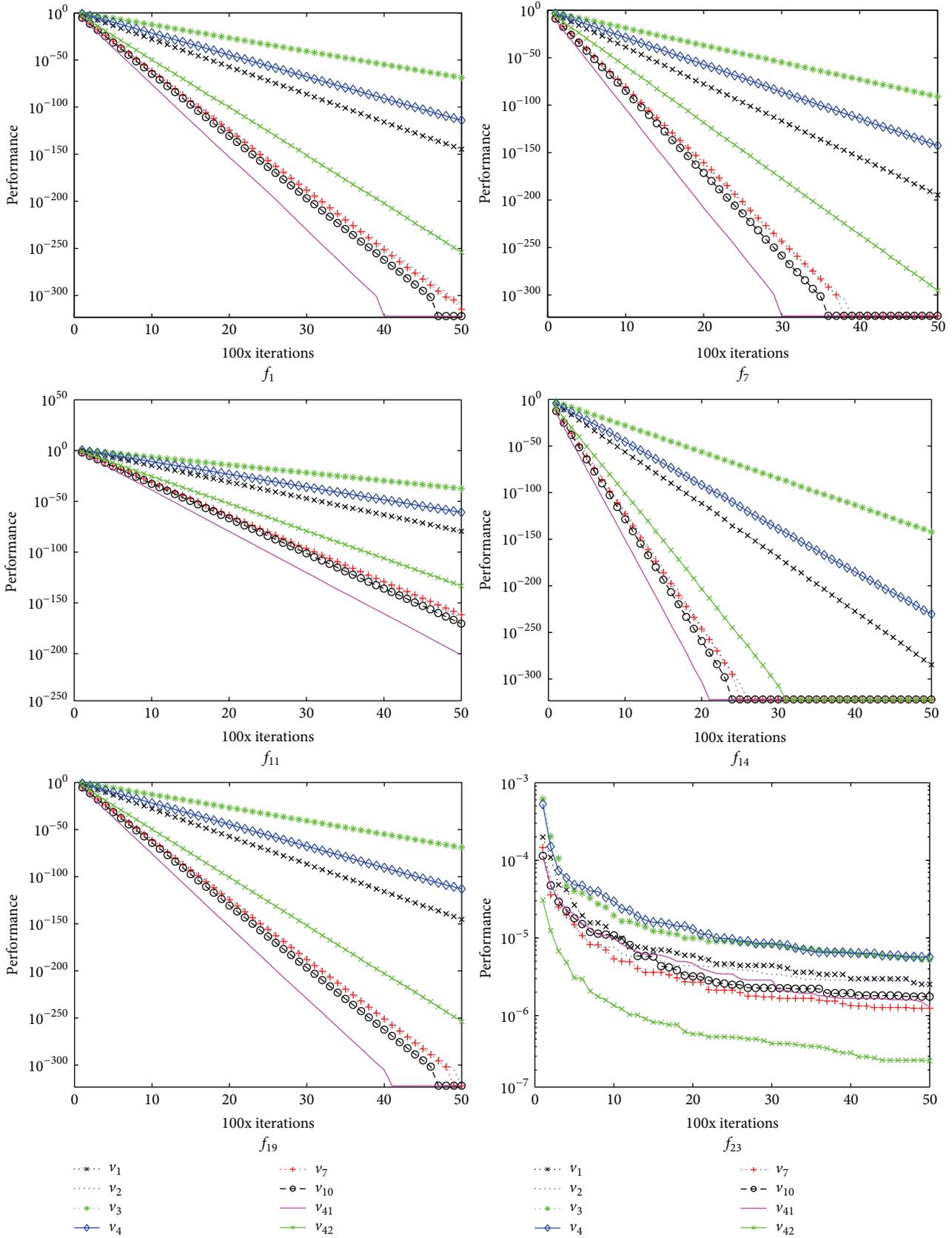


FIGURE 10: 10D: logarithmic convergence graphs of average fitness for some functions showing number of iterations horizontally and average fitness vertically.

TABLE 3: Fitness results (mean \pm SD) of DE variants for functions ($f_1 - f_{30}$).

Fun	DIM	Iter.	V_1	V_2	V_3	V_4	V_7	V_{10}	V_{11}	V_{12}	
f_1	10D	5000	1.76E-145 \pm 7.33E-145	2.69E-310 \pm 0.00E+00	1.53E-69 \pm 3.05E-69	1.09E-114 \pm 3.68E-114	2.47E-315 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	2.94E-255 \pm 0.00E+00
	20D	10000	3.25E-109 \pm 7.70E-109	2.22E-315 \pm 0.00E+00	9.92E-30 \pm 1.71E-29	1.64E-63 \pm 2.68E-63	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	9.86E-235 \pm 0.00E+00
	30D	15000	2.10E-80 \pm 7.19E-80	1.24E-313 \pm 0.00E+00	1.95E-12 \pm 1.18E-12	6.81E-38 \pm 9.97E-38	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.71E-227 \pm 0.00E+00
f_2	10D	5000	1.03E-165 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	3.53E-84 \pm 6.08E-84	9.30E-135 \pm 2.20E-134	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	5.95E-270 \pm 0.00E+00
	20D	10000	6.33E-118 \pm 1.08E-117	0.00E+00 \pm 0.00E+00	1.85E-33 \pm 2.16E-33	1.68E-70 \pm 3.09E-70	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	2.28E-240 \pm 0.00E+00
	30D	15000	3.65E-85 \pm 6.96E-85	9.88E-324 \pm 0.00E+00	3.23E-13 \pm 2.21E-13	4.26E-40 \pm 1.11E-39	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.44E-229 \pm 0.00E+00
f_3	10D	5000	2.01E-29 \pm 3.29E-29	1.39E-86 \pm 4.53E-86	3.33E-11 \pm 3.84E-11	3.37E-22 \pm 5.88E-22	2.70E-94 \pm 8.76E-94	1.03E-86 \pm 5.42E-86	9.92E-120 \pm 4.77E-118	4.73E-58 \pm 3.59E-58	4.73E-58 \pm 3.59E-58
	20D	10000	4.71E-01 \pm 6.91E-06	6.41E-26 \pm 1.32E-25	8.11E+02 \pm 2.40E+02	3.69E+01 \pm 2.20E+01	4.88E-34 \pm 1.79E-34	1.78E-28 \pm 3.51E-28	1.36E-47 \pm 6.90E-47	7.22E-24 \pm 1.15E-23	7.22E-24 \pm 1.15E-23
	30D	15000	2.19E+03 \pm 7.20E+02	4.16E-10 \pm 6.03E-10	7.93E+03 \pm 9.36E+02	4.70E+03 \pm 1.04E+03	4.45E-16 \pm 7.71E-16	1.06E-11 \pm 1.79E-11	4.00E-26 \pm 6.90E-26	1.50E-14 \pm 1.67E-14	1.50E-14 \pm 1.67E-14
f_4	10D	5000	4.07E-11 \pm 7.30E-11	5.32E-01 \pm 1.36E+00	8.29E-04 \pm 1.26E-03	2.89E-08 \pm 4.09E-08	4.51E-01 \pm 1.21E+00	2.66E-01 \pm 9.94E-01	2.66E-01 \pm 7.16E-01	6.91E-16 \pm 7.16E-01	6.91E-16 \pm 7.16E-01
	20D	10000	5.89E-06 \pm 6.91E-06	6.64E-01 \pm 1.49E+00	7.63E+00 \pm 6.43E-01	2.36E+00 \pm 1.24E+01	1.33E-01 \pm 7.16E-01	2.66E-01 \pm 9.94E-01	1.33E-01 \pm 7.16E-01	6.97E-01 \pm 1.21E+00	6.97E-01 \pm 1.21E+00
	30D	15000	2.66E-02 \pm 1.20E-02	7.97E-01 \pm 1.59E+00	3.45E+00 \pm 2.89E+00	9.00E+00 \pm 7.07E-01	3.99E-01 \pm 1.20E+00	3.99E-01 \pm 1.20E+00	3.99E-01 \pm 1.20E+00	9.34E+00 \pm 3.05E+00	9.34E+00 \pm 3.05E+00
f_5	10D	5000	6.63E-02 \pm 2.48E-01	4.15E+00 \pm 1.74E+00	2.67E+00 \pm 2.67E+00	1.76E+00 \pm 1.14E+00	2.18E+00 \pm 6.27E-01	9.95E-01 \pm 1.06E+00	1.53E+00 \pm 1.17E+00	3.32E-02 \pm 0.00E+00	3.32E-02 \pm 0.00E+00
	20D	10000	3.24E+01 \pm 6.67E+00	1.18E+01 \pm 3.55E+00	6.01E+01 \pm 6.13E+00	5.37E+01 \pm 5.16E+00	2.50E+01 \pm 5.62E+00	4.34E+00 \pm 1.35E+00	6.33E+00 \pm 2.03E+00	6.63E-02 \pm 2.48E-01	6.63E-02 \pm 2.48E-01
	30D	15000	1.02E+02 \pm 6.58E+00	2.34E+01 \pm 6.93E+00	1.48E+02 \pm 1.06E+01	1.32E+02 \pm 6.88E+00	7.02E+01 \pm 9.81E+00	1.00E+01 \pm 2.93E+00	1.34E+01 \pm 3.64E+00	1.66E-01 \pm 3.71E-01	1.66E-01 \pm 3.71E-01
f_6	10D	5000	3.20E-03 \pm 6.67E-03	4.96E-02 \pm 2.41E-02	1.68E-01 \pm 3.13E-02	4.70E-02 \pm 3.52E-02	1.76E-02 \pm 1.17E-02	2.08E-02 \pm 1.59E-02	1.44E-02 \pm 1.53E-02	9.86E-04 \pm 5.95E-03	9.86E-04 \pm 5.95E-03
	20D	10000	1.07E-03 \pm 2.75E-03	1.01E-02 \pm 1.53E-02	3.53E-02 \pm 7.05E-02	8.46E-03 \pm 8.64E-03	2.87E-03 \pm 5.65E-03	2.71E-03 \pm 4.51E-03	2.96E-03 \pm 4.82E-03	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00
	30D	15000	4.93E-04 \pm 1.84E-03	5.82E-03 \pm 9.68E-03	8.32E-02 \pm 7.92E-02	4.93E-03 \pm 3.36E-03	1.07E-03 \pm 2.75E-03	2.30E-03 \pm 4.71E-03	4.93E-04 \pm 1.84E-03	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00
f_7	10D	5000	2.59E-195 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	7.97E-92 \pm 1.43E-91	2.61E-143 \pm 1.06E-142	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	8.00E-294 \pm 0.00E+00	8.00E-294 \pm 0.00E+00
	20D	10000	6.79E-164 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.64E-40 \pm 6.56E-40	4.66E-84 \pm 2.59E-83	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.65E-299 \pm 0.00E+00	1.65E-299 \pm 0.00E+00
	30D	15000	2.01E-111 \pm 5.48E-111	0.00E+00 \pm 0.00E+00	7.97E-17 \pm 9.77E-17	1.28E-49 \pm 4.28E-49	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	2.10E-300 \pm 0.00E+00	2.10E-300 \pm 0.00E+00
f_8	10D	5000	1.25E-02 \pm 3.76E-02	2.56E-01 \pm 8.15E-02	3.36E-01 \pm 8.74E-02	1.26E-01 \pm 8.21E-02	1.61E-01 \pm 4.84E-02	7.33E-02 \pm 8.17E-02	9.79E-02 \pm 1.01E-01	4.18E-03 \pm 5.21E-13	4.18E-03 \pm 5.21E-13
	20D	10000	7.30E-01 \pm 2.75E-02	3.51E-01 \pm 7.27E-02	1.07E+00 \pm 7.52E-02	9.67E-01 \pm 6.48E-02	5.40E-01 \pm 9.69E-02	1.55E-01 \pm 6.27E-02	2.32E-01 \pm 6.28E-02	8.85E-03 \pm 6.66E-02	8.85E-03 \pm 6.66E-02
	30D	15000	1.09E+00 \pm 7.43E-02	3.52E-01 \pm 6.25E-02	1.57E+00 \pm 6.60E-02	1.34E+00 \pm 8.02E-02	8.14E-01 \pm 1.04E-01	2.26E-01 \pm 6.83E-02	2.73E-01 \pm 5.21E-02	7.22E-03 \pm 2.16E-02	7.22E-03 \pm 2.16E-02
f_9	10D	5000	3.88E-07 \pm 4.11E-07	2.66E-07 \pm 2.48E-07	1.02E-06 \pm 8.02E-07	7.74E-07 \pm 5.97E-07	4.14E-07 \pm 3.93E-07	3.49E-07 \pm 3.83E-07	3.31E-07 \pm 3.83E-07	1.97E-08 \pm 1.68E-08	1.97E-08 \pm 1.68E-08
	20D	10000	3.60E-07 \pm 3.74E-07	1.39E-07 \pm 1.38E-07	1.10E-06 \pm 7.89E-07	5.86E-07 \pm 6.51E-07	1.57E-07 \pm 1.46E-07	1.55E-07 \pm 1.24E-07	1.55E-07 \pm 1.44E-07	4.36E-09 \pm 5.29E-09	4.36E-09 \pm 5.29E-09
	30D	15000	2.23E-07 \pm 1.98E-07	1.03E-07 \pm 7.96E-08	1.10E-06 \pm 8.75E-07	5.59E-07 \pm 4.14E-07	7.24E-08 \pm 6.20E-08	3.16E-02 \pm 1.70E-01	1.86E-03 \pm 1.00E-02	1.42E-09 \pm 1.24E-09	1.42E-09 \pm 1.24E-09
f_{10}	10D	5000	9.21E-43 \pm 2.10E-42	9.04E-114 \pm 2.78E-113	3.46E-17 \pm 3.08E-17	1.10E-31 \pm 2.27E-31	3.32E-121 \pm 1.75E-120	1.16E-117 \pm 3.13E-117	2.11E-151 \pm 5.43E-151	1.71E-37 \pm 1.06E-36	1.71E-37 \pm 1.06E-36
	20D	10000	1.68E-09 \pm 1.15E-09	7.49E-44 \pm 1.88E-43	1.29E+00 \pm 3.74E-01	2.46E-04 \pm 1.36E-04	1.29E-53 \pm 3.18E-53	3.71E-50 \pm 1.09E-49	3.83E-71 \pm 6.60E-71	5.69E-02 \pm 4.26E-02	5.69E-02 \pm 4.26E-02
	30D	15000	4.80E-02 \pm 2.97E-02	2.26E-24 \pm 3.92E-24	4.92E+01 \pm 6.73E+00	8.27E+00 \pm 2.25E+00	1.39E-32 \pm 2.11E-32	5.45E-30 \pm 9.59E-30	1.11E-44 \pm 4.40E-44	2.08E+00 \pm 7.83E-01	2.08E+00 \pm 7.83E-01
f_{11}	10D	5000	3.69E-80 \pm 8.46E-80	2.82E-165 \pm 0.00E+00	4.43E-38 \pm 3.82E-38	1.26E-61 \pm 2.12E-61	2.58E-162 \pm 1.48E-162	3.46E-171 \pm 0.00E+00	7.33E-202 \pm 0.00E+00	5.07E-134 \pm 6.53E-134	5.07E-134 \pm 6.53E-134
	20D	10000	5.37E-62 \pm 7.70E-62	5.49E-170 \pm 0.00E+00	8.19E-17 \pm 5.19E-17	3.89E-35 \pm 5.50E-35	1.88E-179 \pm 0.00E+00	3.80E-190 \pm 0.00E+00	1.67E-232 \pm 0.00E+00	1.94E-124 \pm 2.30E-124	1.94E-124 \pm 2.30E-124
	30D	15000	9.57E-47 \pm 1.12E-46	2.46E-171 \pm 0.00E+00	6.24E-07 \pm 2.38E-07	5.79E-21 \pm 6.51E-21	1.12E-188 \pm 0.00E+00	2.16E-202 \pm 0.00E+00	1.60E-257 \pm 0.00E+00	3.84E-120 \pm 3.69E-120	3.84E-120 \pm 3.69E-120
f_{12}	10D	5000	0.00E+00 \pm 0.00E+00								
	20D	10000	0.00E+00 \pm 0.00E+00	1.00E-01 \pm 3.00E-01	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	3.33E-02 \pm 1.80E-01	3.33E-02 \pm 1.80E-01	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00
	30D	15000	0.00E+00 \pm 0.00E+00	6.67E-01 \pm 1.01E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	6.67E-02 \pm 2.49E-01	3.33E-02 \pm 1.80E-01	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00
f_{13}	10D	5000	2.56E-285 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	6.35E-143 \pm 1.79E-142	5.39E-231 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00
	20D	10000	7.71E-177 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	9.82E-49 \pm 1.53E-48	5.74E-104 \pm 1.53E-103	0.00E+00 \pm 0.00E+00				
	30D	15000	4.51E-114 \pm 2.31E-113	0.00E+00 \pm 0.00E+00	4.50E-17 \pm 4.05E-17	1.10E-53 \pm 4.00E-53	0.00E+00 \pm 0.00E+00				
f_{14}	10D	5000	4.65E-05 \pm 5.05E-05	4.72E-05 \pm 4.27E-05	5.14E-05 \pm 4.25E-05	6.14E-05 \pm 5.44E-05	5.24E-05 \pm 4.40E-05	4.91E-05 \pm 4.00E			

TABLE 3: Continued.

Fun	DIM	Iter.	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_{10}	V_{11}	V_{12}
f_{16}	10D	5000	8.53E+01 ± 7.11E-14	8.53E+01 ± 6.23E-14	8.53E+01 ± 6.95E-14	8.53E+01 ± 7.03E-14	8.53E+01 ± 5.58E-14	8.53E+01 ± 5.82E-14	8.53E+01 ± 7.11E-14	8.53E+01 ± 7.11E-14	8.53E+01 ± 7.11E-14	8.53E+01 ± 7.11E-14
	20D	10000	1.21E-04 ± 1.10E-04	7.97E-05 ± 8.70E-05	1.94E-04 ± 1.87E-04	1.86E-04 ± 1.92E-04	4.77E-05 ± 4.62E-05	3.96E-05 ± 4.79E-05	6.73E-05 ± 7.35E-05	6.73E-05 ± 7.35E-05	6.73E-05 ± 7.35E-05	6.73E-05 ± 7.35E-05
	30D	15000	1.49E-04 ± 1.67E-04	1.13E-04 ± 1.14E-04	4.53E-04 ± 5.16E-04	4.49E-04 ± 4.36E-04	6.63E-05 ± 6.82E-05	6.64E-05 ± 6.02E-05	7.43E-05 ± 7.72E-05	7.43E-05 ± 7.72E-05	7.43E-05 ± 7.72E-05	7.43E-05 ± 7.72E-05
f_{17}	10D	5000	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	20D	10000	0.00E+00 ± 0.00E+00	5.91E-02 ± 8.18E-02	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	9.85E-03 ± 3.69E-02	2.22E-17 ± 6.66E-17	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	30D	15000	0.00E+00 ± 0.00E+00	1.38E-01 ± 1.32E-01	1.15E-10 ± 1.18E-10	0.00E+00 ± 0.00E+00	2.46E-02 ± 5.51E-02	1.97E-02 ± 5.02E-02	1.97E-02 ± 5.02E-02	1.97E-02 ± 5.02E-02	1.97E-02 ± 5.02E-02	1.97E-02 ± 5.02E-02
f_{18}	10D	5000	2.38E-140 ± 7.10E-140	0.00E+00 ± 0.00E+00	6.26E-64 ± 9.94E-64	8.59E-109 ± 2.49E-108	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	7.88E-250 ± 0.00E+00
	20D	10000	6.90E-109 ± 1.01E-103	0.00E+00 ± 0.00E+00	4.09E-24 ± 6.53E-24	1.36E-57 ± 3.99E-57	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	2.86E-230 ± 0.00E+00
	30D	15000	3.95E-75 ± 7.03E-75	0.00E+00 ± 0.00E+00	8.54E-07 ± 6.45E-07	1.61E-32 ± 2.77E-32	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	1.24E-221 ± 0.00E+00
f_{19}	10D	5000	3.99E-146 ± 1.16E-145	0.00E+00 ± 0.00E+00	9.32E-70 ± 1.47E-69	1.17E-113 ± 5.13E-113	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	3.12E-255 ± 0.00E+00
	20D	10000	2.65E-109 ± 8.16E-109	0.00E+00 ± 0.00E+00	6.25E-30 ± 7.38E-30	1.53E-63 ± 3.40E-63	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	8.15E-236 ± 0.00E+00
	30D	15000	8.28E-81 ± 1.39E-80	0.00E+00 ± 0.00E+00	1.76E-12 ± 9.70E-13	3.91E-38 ± 5.30E-38	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	3.37E-227 ± 0.00E+00
f_{20}	10D	5000	1.75E-143 ± 4.10E-143	0.00E+00 ± 0.00E+00	3.61E-67 ± 6.53E-67	3.89E-112 ± 9.53E-112	1.51E-312 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	5.58E-253 ± 0.00E+00
	20D	10000	1.13E-106 ± 2.38E-106	2.20E-312 ± 0.00E+00	5.06E-27 ± 7.98E-27	4.83E-61 ± 9.92E-61	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	3.88E-233 ± 0.00E+00
	30D	15000	4.31E-78 ± 6.35E-78	5.90E-311 ± 0.00E+00	8.05E-10 ± 5.01E-10	1.89E-35 ± 2.98E-35	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	2.42E-224 ± 0.00E+00
f_{21}	10D	5000	1.79E-143 ± 3.30E-143	4.19E-307 ± 0.00E+00	1.75E-66 ± 3.86E-66	9.24E-112 ± 1.79E-111	3.34E-320 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	1.18E-252 ± 0.00E+00
	20D	10000	2.36E-106 ± 5.82E-106	3.94E-313 ± 0.00E+00	2.76E-27 ± 2.93E-27	5.25E-61 ± 8.13E-61	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	3.85E-232 ± 0.00E+00
	30D	15000	2.55E-77 ± 1.09E-76	5.21E-319 ± 0.00E+00	8.43E-10 ± 4.46E-10	2.67E-35 ± 3.14E-35	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	5.27E-224 ± 0.00E+00
f_{22}	10D	5000	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	8.22E-33 ± 4.43E-32	0.00E+00 ± 3.38E-32	0.00E+00 ± 0.00E+00
	20D	10000	8.63E-33 ± 6.65E-32	1.89E-32 ± 6.35E-32	5.70E-18 ± 7.17E-18	0.00E+00 ± 0.00E+00	2.34E-31 ± 3.59E-31	1.01E-31 ± 1.32E-31	1.51E-31 ± 1.56E-31	1.51E-31 ± 1.56E-31	1.51E-31 ± 1.56E-31	5.11E-31 ± 5.29E-31
	30D	15000	1.40E-31 ± 1.84E-31	2.20E-31 ± 3.06E-31	5.37E-05 ± 4.44E-05	1.58E-27 ± 2.62E-27	1.29E-30 ± 2.89E-30	5.06E-31 ± 3.40E-31	6.86E-31 ± 4.01E-31	6.86E-31 ± 4.01E-31	6.86E-31 ± 4.01E-31	1.06E-22 ± 3.32E-22
f_{23}	10D	5000	2.53E-06 ± 2.32E-06	2.67E-06 ± 2.58E-06	5.26E-06 ± 4.46E-06	5.69E-06 ± 6.10E-06	1.24E-06 ± 1.03E-06	1.76E-06 ± 2.20E-06	1.56E-06 ± 1.32E-06	1.56E-06 ± 1.32E-06	1.56E-06 ± 1.32E-06	2.88E-07 ± 1.91E-07
	20D	10000	2.94E-06 ± 2.94E-06	1.92E-06 ± 1.80E-06	7.99E-06 ± 6.28E-06	5.69E-06 ± 5.00E-06	1.34E-06 ± 1.45E-06	9.49E-07 ± 1.16E-06	1.57E-06 ± 1.36E-06	1.57E-06 ± 1.36E-06	1.57E-06 ± 1.36E-06	7.14E-08 ± 7.03E-08
	30D	15000	3.23E-06 ± 2.88E-06	1.69E-06 ± 1.71E-06	1.48E-05 ± 1.48E-05	6.77E-06 ± 5.84E-06	1.24E-06 ± 1.37E-06	1.42E-06 ± 1.04E-06	1.36E-06 ± 1.30E-06	1.36E-06 ± 1.30E-06	1.36E-06 ± 1.30E-06	4.64E-08 ± 5.37E-08
f_{24}	10D	5000	0.00E+00 ± 0.00E+00	7.40E-16 ± 1.66E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	2.96E-16 ± 1.11E-15	4.44E-16 ± 1.33E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	20D	10000	0.00E+00 ± 0.00E+00	6.19E-15 ± 5.31E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	5.34E-15 ± 5.10E-15	3.66E-15 ± 4.71E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	30D	15000	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	0.00E+00 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	0.00E+00 ± 0.00E+00
f_{25}	10D	5000	0.00E+00 ± 0.00E+00	2.22E-15 ± 2.22E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	1.04E-15 ± 1.88E-15	1.63E-15 ± 2.14E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	20D	10000	0.00E+00 ± 0.00E+00	6.19E-15 ± 4.32E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	5.06E-15 ± 4.13E-15	5.63E-15 ± 5.03E-15	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	30D	15000	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	0.00E+00 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	1.00E+09 ± 0.00E+00	0.00E+00 ± 0.00E+00
f_{26}	10D	5000	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	5.73E-224 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	20D	10000	0.00E+00 ± 0.00E+00	1.36E-94 ± 7.22E-94	1.08E-109 ± 0.00E+00	1.10E-109 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
	30D	15000	0.00E+00 ± 0.00E+00	3.01E-44 ± 7.05E-44	5.18E-118 ± 2.76E-117	5.18E-118 ± 2.76E-117	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00	0.00E+00 ± 0.00E+00
f_{27}	10D	5000	1.97E-15 ± 1.58E-16	1.30E-15 ± 2.62E-16	4.96E-08 ± 2.66E-07	1.65E-15 ± 2.27E-16	3.45E-13 ± 1.85E-12	1.57E-15 ± 3.46E-16	1.81E-15 ± 1.98E-16	1.81E-15 ± 1.98E-16	1.81E-15 ± 1.98E-16	1.89E-15 ± 2.14E-16
	20D	10000	4.00E-15 ± 1.93E-16	2.53E-15 ± 4.43E-16	8.05E+00 ± 1.15E+00	3.41E-07 ± 1.83E-06	2.97E-15 ± 4.67E-16	3.03E-15 ± 4.28E-16	3.42E-15 ± 3.81E-16	3.42E-15 ± 3.81E-16	3.42E-15 ± 3.81E-16	3.89E-15 ± 2.19E-16
	30D	15000	6.05E-15 ± 1.49E-16	3.65E-15 ± 5.32E-16	5.03E+01 ± 1.37E+01	2.13E+00 ± 4.20E+00	4.23E-15 ± 5.43E-16	4.46E-15 ± 5.77E-16	5.13E-15 ± 2.20E-15	5.13E-15 ± 2.20E-15	5.13E-15 ± 2.20E-15	5.86E-15 ± 2.61E-16
f_{28}	10D	5000	7.32E-13 ± 1.08E-12	2.66E-02 ± 7.93E-02	8.10E-01 ± 1.54E-01	8.45E-02 ± 1.30E-01	5.31E-53 ± 8.86E-52	1.98E-28 ± 1.05E-27	7.35E-118 ± 1.46E-117	7.35E-118 ± 1.46E-117	7.35E-118 ± 1.46E-117	6.82E-49 ± 2.60E-48
	20D	10000	3.80E+00 ± 4.43E-01	9.35E-01 ± 1.26E+00	4.86E+00 ± 5.78E-01	5.48E+00 ± 1.00E+00	1.30E-01 ± 3.36E-01	3.21E-02 ± 1.67E-01	4.98E-118 ± 1.94E-117	4.98E-118 ± 1.94E-117	4.98E-118 ± 1.94E-117	4.74E-22 ± 5.57E-22
	30D	15000	9.50E+00 ± 7.40E-01	4.88E+00 ± 3.39E+00	1.08E+01 ± 7.54E-01	1.28E+01 ± 1.33E+00	7.84E+00 ± 1.10E+00	6.29E-01 ± 1.46E+00	1.74E-120 ± 7.93E-120	1.74E-120 ± 7.93E-120	1.74E-120 ± 7.93E-120	3.87E-10 ± 4.29E-10
f_{29}	10D	5000	1.07E-08 ± 2.98E-08	3.12E-07 ± 8.42E-07	8.20E-10 ± 2.05E-09	2.08E-10 ± 1.49E-09	8.71E-10 ± 1.67E-09	9.37E-10 ± 2.28E-09	9.43E-08 ± 7.09E-07	9.43E-08 ± 7.09E-07	9.43E-08 ± 7.09E-07	4.86E-08 ± 2.13E-07
	20D	10000	1.58E-08 ± 5.52E-08	1.96E-07 ± 4.95E-07	2.32E-10 ± 3.77E-10	2.08E-10 ± 3.77E-10	8.01E-11 ± 1.80E-10	7.44E-10 ± 2.08E-09	1.11E-07 ± 4.98E-07	1.11E-07 ± 4.98E-07	1.11E-07 ± 4.98E-07	4.63E-08 ± 8.04E-08
	30D	15000	1.73E-08 ± 7.00E-08	3.01E-07 ± 5.49E-07	5.40E-11 ± 8.01E-11	4.47E-11 ± 5.97E-11	6.90E-11 ± 1.97E-10	2.75E-10 ± 1.03E-09	2.04E-07 ± 9.92E-07	2.04E-07 ± 9.92E-07	2.04E-07 ± 9.92E-07	6.45E-08 ± 1.30E-07
f_{30}	10D	5000	2.08E-04 ± 1.09E-19	2.18E-04 ± 1.60E-05	3.02E-04 ± 3.11E-05	2.26E-04 ± 2.49E-05	2.13E-04 ± 4.52E-06	2.10E-04 ± 9.85E-06	2.10E-04 ± 4.68E-06	2.10E-04 ± 4.68E-06	2.10E-04 ± 4.68E-06	2.08E-04 ± 2.21E-20
	20D	10000	1.74E-07 ± 3.77E-08	3.06E-08 ± 5.82E-09	2.67E-07 ± 5.91E-08	2.71E-07 ± 6.91E-08	1.75E-07 ± 4.85E-08	1.04E-07 ± 5.80E-08	2.64E-08 ± 4.35E-09	2.64E-08 ± 4.35E-09	2.64E-08 ± 4.35E-09	1.90E-08 ± 1.48E-24
	30D	15000										

TABLE 4: Number of function calls (mean \pm SD) of DE variants for functions (f_1-f_{30}).

Fun	DIM	Iter.	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_{10}	V_{11}	V_{12}
f_1	10D	10^5	$2.17E+02 \pm 1.02E+01$	$1.05E+02 \pm 5.36E+00$	$4.40E+02 \pm 1.20E+01$	$2.73E+02 \pm 1.31E+01$	$1.03E+02 \pm 3.88E+00$	$9.68E+01 \pm 3.77E+00$	$1.03E+02 \pm 3.88E+00$	$9.68E+01 \pm 3.77E+00$	$8.38E+01 \pm 4.86E+00$	$1.29E+02 \pm 5.97E+00$
	20D	2×10^5	$6.22E+02 \pm 1.84E+01$	$2.18E+02 \pm 9.01E+00$	$2.19E+03 \pm 7.94E+01$	$1.04E+03 \pm 3.78E+01$	$2.01E+02 \pm 5.41E+00$	$1.91E+02 \pm 6.81E+00$	$2.01E+02 \pm 5.41E+00$	$1.91E+02 \pm 6.81E+00$	$1.56E+02 \pm 5.21E+00$	$2.97E+02 \pm 9.27E+00$
	30D	3×10^5	$1.27E+03 \pm 3.04E+01$	$3.42E+02 \pm 1.21E+01$	$7.56E+03 \pm 1.95E+02$	$2.63E+03 \pm 9.07E+01$	$2.91E+02 \pm 7.50E+00$	$2.77E+02 \pm 9.53E+00$	$2.91E+02 \pm 7.50E+00$	$2.77E+02 \pm 9.53E+00$	$2.21E+02 \pm 7.02E+00$	$4.80E+02 \pm 1.09E+01$
f_2	10D	10^5	$2.07E+02 \pm 9.44E+00$	$1.04E+02 \pm 4.47E+00$	$4.01E+02 \pm 1.95E+01$	$2.55E+02 \pm 1.02E+01$	$1.03E+02 \pm 3.46E+00$	$9.78E+01 \pm 4.21E+00$	$1.03E+02 \pm 3.46E+00$	$9.78E+01 \pm 4.21E+00$	$8.47E+01 \pm 3.34E+00$	$1.30E+02 \pm 5.28E+00$
	20D	2×10^5	$6.33E+02 \pm 1.53E+01$	$2.35E+02 \pm 1.03E+01$	$2.10E+03 \pm 4.47E+01$	$1.03E+03 \pm 3.19E+01$	$2.12E+02 \pm 4.52E+00$	$2.03E+02 \pm 6.60E+00$	$2.12E+02 \pm 4.52E+00$	$2.03E+02 \pm 6.60E+00$	$1.69E+02 \pm 5.48E+00$	$3.29E+02 \pm 7.92E+00$
	30D	3×10^5	$1.36E+03 \pm 3.12E+01$	$3.78E+02 \pm 1.32E+01$	$7.55E+03 \pm 2.18E+02$	$2.76E+03 \pm 9.85E+01$	$3.26E+02 \pm 8.83E+00$	$3.09E+02 \pm 8.87E+00$	$3.26E+02 \pm 8.83E+00$	$3.09E+02 \pm 8.87E+00$	$2.46E+02 \pm 7.61E+00$	$5.40E+02 \pm 1.19E+01$
f_3	10D	10^5	$1.28E+03 \pm 7.67E+01$	$4.55E+02 \pm 3.15E+01$	$2.97E+03 \pm 1.24E+02$	$1.67E+03 \pm 9.23E+01$	$4.22E+02 \pm 3.13E+01$	$4.56E+02 \pm 3.00E+01$	$4.22E+02 \pm 3.13E+01$	$4.56E+02 \pm 3.00E+01$	$3.30E+02 \pm 1.91E+01$	$6.73E+02 \pm 3.11E+01$
	20D	2×10^5	$2.00E+04 \pm 1.10E+03$	$3.04E+03 \pm 1.60E+02$	$1.06E+05 \pm 5.17E+03$	$3.74E+04 \pm 2.03E+03$	$2.35E+03 \pm 1.28E+02$	$2.75E+02 \pm 1.88E+02$	$2.35E+03 \pm 1.28E+02$	$2.75E+02 \pm 1.88E+02$	$1.72E+03 \pm 1.05E+02$	$3.22E+03 \pm 1.38E+02$
	30D	3×10^5	$1.82E+05 \pm 1.06E+04$	$9.64E+03 \pm 8.13E+02$	—	—	$6.96E+03 \pm 4.12E+02$	$8.81E+03 \pm 6.54E+02$	$6.96E+03 \pm 4.12E+02$	$8.81E+03 \pm 6.54E+02$	$4.57E+03 \pm 2.43E+02$	$7.31E+03 \pm 2.20E+02$
f_4	10D	10^5	$3.08E+03 \pm 1.36E+02$	$1.31E+03 \pm 2.23E+02$	$6.46E+03 \pm 3.24E+02$	$3.67E+03 \pm 2.24E+02$	$2.83E+03 \pm 7.80E+03$	$1.43E+03 \pm 6.87E+01$	$2.83E+03 \pm 7.80E+03$	$1.43E+03 \pm 6.87E+01$	$1.23E+03 \pm 5.85E+01$	$2.59E+03 \pm 3.02E+02$
	20D	2×10^5	$9.58E+03 \pm 2.89E+02$	$3.17E+03 \pm 6.65E+02$	$3.74E+04 \pm 1.53E+03$	$1.61E+04 \pm 7.13E+02$	$3.24E+03 \pm 5.66E+02$	$3.38E+03 \pm 3.04E+02$	$3.24E+03 \pm 5.66E+02$	$3.38E+03 \pm 3.04E+02$	$2.84E+03 \pm 4.03E+02$	$1.47E+04 \pm 2.95E+03$
	30D	3×10^5	$2.11E+04 \pm 5.34E+02$	$5.72E+03 \pm 8.36E+02$	$1.54E+05 \pm 7.14E+03$	$4.23E+04 \pm 2.36E+03$	$4.87E+03 \pm 1.17E+03$	$5.59E+03 \pm 2.73E+02$	$4.87E+03 \pm 1.17E+03$	$5.59E+03 \pm 2.73E+02$	$4.78E+03 \pm 5.89E+02$	$3.87E+04 \pm 7.47E+03$
f_5	10D	10^5	$1.43E+03 \pm 2.46E+02$	—	$5.97E+03 \pm 7.61E+02$	$4.33E+03 \pm 2.28E+02$	$1.93E+04 \pm 0.00E+00$	$7.29E+02 \pm 2.13E+02$	$1.93E+04 \pm 0.00E+00$	$7.29E+02 \pm 2.13E+02$	$5.21E+02 \pm 1.85E+02$	$2.88E+02 \pm 1.37E+01$
	20D	2×10^5	$2.09E+04 \pm 5.74E+03$	—	—	—	—	$2.70E+03 \pm 0.00E+00$	—	—	—	$6.33E+02 \pm 3.50E+01$
	30D	3×10^5	$1.52E+05 \pm 3.90E+04$	—	—	—	—	—	—	—	—	$1.00E+03 \pm 4.41E+01$
f_6	10D	10^5	$1.96E+03 \pm 4.08E+02$	$3.57E+02 \pm 9.12E+01$	$1.29E+04 \pm 2.58E+03$	$1.99E+03 \pm 0.00E+00$	$2.96E+03 \pm 1.48E+03$	$1.19E+03 \pm 0.00E+00$	$2.96E+03 \pm 1.48E+03$	$1.19E+03 \pm 0.00E+00$	$3.92E+02 \pm 2.44E+02$	$5.62E+02 \pm 1.01E+02$
	20D	2×10^5	$1.32E+03 \pm 3.11E+02$	$3.30E+02 \pm 1.24E+01$	$1.12E+04 \pm 2.59E+03$	$2.37E+03 \pm 3.45E+02$	$5.19E+02 \pm 2.21E+02$	$2.88E+02 \pm 1.76E+01$	$5.19E+02 \pm 2.21E+02$	$2.88E+02 \pm 1.76E+01$	$2.36E+02 \pm 1.67E+01$	$6.02E+02 \pm 1.37E+02$
	30D	3×10^5	$2.09E+03 \pm 3.06E+02$	$4.81E+02 \pm 2.38E+01$	$1.94E+04 \pm 3.12E+03$	$4.37E+03 \pm 5.69E+02$	$5.21E+02 \pm 1.85E+02$	$3.95E+02 \pm 2.10E+01$	$5.21E+02 \pm 1.85E+02$	$3.95E+02 \pm 2.10E+01$	$3.17E+02 \pm 1.14E+01$	$7.40E+02 \pm 5.74E+01$
f_7	10D	10^5	$1.13E+02 \pm 8.69E+00$	$5.74E+01 \pm 4.95E+00$	$2.28E+02 \pm 1.90E+01$	$1.45E+02 \pm 1.23E+01$	$5.59E+01 \pm 3.92E+00$	$5.46E+01 \pm 3.73E+00$	$5.59E+01 \pm 3.92E+00$	$5.46E+01 \pm 3.73E+00$	$4.50E+01 \pm 3.38E+00$	$7.39E+01 \pm 6.10E+00$
	20D	2×10^5	$2.62E+02 \pm 2.15E+01$	$9.43E+01 \pm 1.01E+01$	$8.84E+02 \pm 6.55E+01$	$4.39E+02 \pm 3.89E+01$	$8.02E+01 \pm 6.20E+00$	$7.62E+01 \pm 4.87E+00$	$8.02E+01 \pm 6.20E+00$	$7.62E+01 \pm 4.87E+00$	$6.14E+01 \pm 4.93E+00$	$1.44E+02 \pm 1.29E+01$
	30D	3×10^5	$5.20E+02 \pm 4.12E+01$	$1.24E+02 \pm 1.20E+01$	$2.70E+03 \pm 2.99E+02$	$1.01E+03 \pm 9.80E+01$	$1.03E+02 \pm 9.07E+00$	$9.56E+01 \pm 8.35E+00$	$1.03E+02 \pm 9.07E+00$	$9.56E+01 \pm 8.35E+00$	$7.06E+01 \pm 5.21E+00$	$2.15E+02 \pm 2.95E+01$
f_8	10D	10^5	$1.93E+03 \pm 3.42E+02$	—	$7.25E+03 \pm 1.02E+03$	$3.31E+03 \pm 4.53E+02$	$2.38E+04 \pm 1.11E+04$	$8.50E+02 \pm 1.38E+02$	$2.38E+04 \pm 1.11E+04$	$8.50E+02 \pm 1.38E+02$	$5.53E+02 \pm 1.32E+02$	$3.60E+02 \pm 2.12E+01$
	20D	2×10^5	$2.65E+04 \pm 6.98E+03$	—	—	—	—	$3.69E+03 \pm 2.17E+03$	—	—	—	$7.98E+02 \pm 2.34E+01$
	30D	3×10^5	$1.96E+05 \pm 4.43E+04$	—	—	—	—	—	—	—	—	$1.26E+03 \pm 3.37E+01$
f_9	10D	10^5	$4.60E+02 \pm 4.40E+02$	$2.28E+02 \pm 1.52E+02$	$9.09E+02 \pm 6.08E+02$	$6.01E+02 \pm 4.11E+02$	$2.24E+02 \pm 1.38E+02$	$2.64E+02 \pm 2.25E+02$	$2.24E+02 \pm 1.38E+02$	$2.64E+02 \pm 2.25E+02$	$2.40E+02 \pm 1.66E+02$	$1.07E+02 \pm 3.17E+01$
	20D	2×10^5	$5.91E+02 \pm 2.58E+02$	$2.49E+02 \pm 1.21E+02$	$2.22E+03 \pm 7.05E+02$	$1.07E+03 \pm 4.29E+02$	$3.13E+02 \pm 1.34E+02$	$2.62E+02 \pm 1.56E+02$	$3.13E+02 \pm 1.34E+02$	$2.62E+02 \pm 1.56E+02$	$2.24E+02 \pm 1.17E+02$	$1.77E+02 \pm 1.68E+01$
	30D	3×10^5	$9.93E+02 \pm 3.39E+02$	$3.06E+02 \pm 9.95E+01$	$5.29E+03 \pm 1.24E+03$	$2.16E+03 \pm 1.05E+03$	$2.98E+02 \pm 1.07E+02$	$2.57E+02 \pm 1.15E+02$	$2.98E+02 \pm 1.07E+02$	$2.57E+02 \pm 1.15E+02$	$2.53E+02 \pm 1.22E+02$	$2.64E+02 \pm 1.79E+01$
f_{10}	10D	10^5	$9.25E+02 \pm 4.04E+01$	$3.63E+02 \pm 1.99E+01$	$2.08E+03 \pm 1.08E+02$	$1.17E+03 \pm 6.41E+01$	$3.38E+02 \pm 1.41E+01$	$3.54E+02 \pm 2.02E+01$	$3.38E+02 \pm 1.41E+01$	$3.54E+02 \pm 2.02E+01$	$2.68E+02 \pm 1.31E+01$	$9.56E+02 \pm 6.02E+01$
	20D	2×10^5	$7.02E+03 \pm 2.73E+02$	$1.92E+03 \pm 1.04E+02$	$2.67E+04 \pm 1.01E+03$	$1.17E+04 \pm 5.27E+02$	$1.61E+03 \pm 5.94E+01$	$1.69E+03 \pm 8.28E+01$	$1.61E+03 \pm 5.94E+01$	$1.69E+03 \pm 8.28E+01$	$1.26E+03 \pm 5.44E+01$	$4.06E+02 \pm 6.37E+03$
	30D	3×10^5	$2.55E+04 \pm 8.58E+02$	$5.02E+03 \pm 2.14E+02$	$1.66E+05 \pm 5.05E+03$	$5.18E+04 \pm 1.75E+03$	$3.92E+03 \pm 1.71E+02$	$4.25E+03 \pm 2.40E+02$	$3.92E+03 \pm 1.71E+02$	$4.25E+03 \pm 2.40E+02$	$2.96E+03 \pm 1.28E+02$	—
f_{11}	10D	10^5	$3.97E+02 \pm 9.40E+00$	$1.95E+02 \pm 5.49E+00$	$8.19E+02 \pm 2.16E+01$	$5.20E+02 \pm 1.53E+01$	$2.00E+02 \pm 5.59E+00$	$1.90E+02 \pm 5.08E+00$	$2.00E+02 \pm 5.59E+00$	$1.90E+02 \pm 5.08E+00$	$1.61E+02 \pm 4.08E+00$	$2.40E+02 \pm 6.35E+00$
	20D	2×10^5	$1.09E+03 \pm 2.38E+01$	$4.03E+02 \pm 1.17E+01$	$3.79E+03 \pm 9.29E+01$	$1.92E+03 \pm 5.53E+01$	$3.88E+02 \pm 8.09E+00$	$3.63E+02 \pm 8.34E+00$	$3.88E+02 \pm 8.09E+00$	$3.63E+02 \pm 8.34E+00$	$2.96E+02 \pm 6.82E+00$	$5.48E+02 \pm 1.18E+01$
	30D	3×10^5	$2.22E+03 \pm 4.22E+01$	$6.15E+02 \pm 1.91E+01$	$1.27E+04 \pm 2.45E+02$	$4.73E+03 \pm 1.08E+02$	$5.68E+02 \pm 1.50E+01$	$5.28E+02 \pm 1.43E+01$	$5.68E+02 \pm 1.50E+01$	$5.28E+02 \pm 1.43E+01$	$4.17E+02 \pm 1.15E+01$	$8.78E+02 \pm 1.37E+01$
f_{12}	10D	10^5	$1.48E+02 \pm 7.39E+00$	$7.12E+01 \pm 4.46E+00$	$3.04E+02 \pm 2.03E+01$	$1.87E+02 \pm 1.42E+01$	$6.92E+01 \pm 4.36E+00$	$6.62E+01 \pm 3.75E+00$	$6.92E+01 \pm 4.36E+00$	$6.62E+01 \pm 3.75E+00$	$5.66E+01 \pm 5.33E+00$	$8.46E+01 \pm 5.40E+00$
	20D	2×10^5	$4.16E+02 \pm 1.84E+01$	$1.51E+02 \pm 9.34E+00$	$1.49E+03 \pm 6.56E+01$	$7.40E+02 \pm 4.56E+01$	$1.44E+02 \pm 3.95E+01$	$1.31E+02 \pm 1.32E+01$	$1.44E+02 \pm 3.95E+01$	$1.31E+02 \pm 1.32E+01$	$1.07E+02 \pm 5.02E+00$	$1.99E+02 \pm 8.18E+00$
	30D	3×10^5	$8.74E+02 \pm 3.96E+01$	$2.35E+02 \pm 1.89E+01$	$5.56E+03 \pm 3.45E+02$	$1.93E+03 \pm 1.33E+02$	$2.07E+02 \pm 1.26E+01$	$3.80E+02 \pm 9.15E+02$	$2.07E+02 \pm 1.26E+01$	$3.80E+02 \pm 9.15E+02$	$1.50E+02 \pm 7.71E+00$	$3.24E+02 \pm 1.03E+01$
f_{13}	10D	10^5	$9.34E+01 \pm 8.06E+00$	$4.56E+01 \pm 4.36E+00$	$1.78E+02 \pm 1.42E+01$	$1.13E+02 \pm 6.94E+01$	$4.55E+01 \pm 4.43E+00$	$4.34E+01 \pm 2.92E+00$	$4.55E+01 \pm 4.43E+00$	$4.34E+01 \pm 2.92E+00$	$3.69E+01 \pm 4.15E+00$	$5.51E+01 \pm 3.91E+00$
	20D	2×10^5	$3.43E+02 \pm 1.89E+01$	$1.18E+02 \pm 7.01E+00$	$1.19E+03 \pm 7.86E+01$	$5.38E+02 \pm 4.07E+01$	$1.05E+02 \pm 7.79E+00$	$9.95E+01 \pm 5.08E+00$	$1.05E+02 \pm 7.79E+00$	$9.95E+01 \pm 5.08E+00$	$8.27E+01 \pm 4.63E+00$	$1.46E+02 \pm 7.33E+00$
	30D	3×10^5	$8.31E+02 \pm 3.93E+01$	$2.00E+02 \pm 1.27E+01$	$5.18E+03 \pm 1.82E+02$	$1.70E+03 \pm 9.52E+01$	$1.72E+02 \pm 7.38E+00$	$1.60E+02 \pm 6.86E+00$	$1.72E+02 \pm 7.38E+00$	$1.60E+02 \pm 6.86E+00$	$1.29E+02 \pm 6.28E+00$	$2.49E+02 \pm 1.06E+01$
f_{14}												

TABLE 4: Continued.

Fun	DIM	Iter.	V_1	V_5	V_4	V_7	V_{10}	V_{41}	V_{12}
f_{16}	10D	10^5	701E + 04 ± 5.68E + 04	4.68E + 04 ± 4.65E + 04	8.44E + 04 ± 5.47E + 04	4.21E + 04 ± 3.51E + 04	5.45E + 04 ± 5.36E + 04	4.42E + 04 ± 4.67E + 04	5.05E + 03 ± 6.35E + 03
	20D	2×10^5	1.13E + 05 ± 8.24E + 04	1.00E + 05 ± 8.88E + 04	1.05E + 05 ± 7.06E + 04	1.06E + 05 ± 7.77E + 04	9.72E + 04 ± 6.93E + 04	1.18E + 05 ± 7.21E + 04	7.72E + 03 ± 6.50E + 03
	30D	3×10^5	2.19E + 02 ± 1.04E + 01	1.06E + 02 ± 5.23E + 00	2.92E + 02 ± 1.62E + 01	1.05E + 02 ± 6.10E + 00	9.91E + 01 ± 4.34E + 00	8.36E + 01 ± 3.87E + 00	1.25E + 02 ± 6.82E + 00
f_{17}	10D	10^5	6.44E + 02 ± 2.46E + 01	2.20E + 02 ± 8.57E + 00	1.17E + 03 ± 6.68E + 01	2.12E + 02 ± 8.93E + 00	1.93E + 02 ± 8.35E + 00	1.58E + 02 ± 7.40E + 00	2.92E + 02 ± 8.07E + 00
	20D	2×10^5	1.38E + 03 ± 6.13E + 01	3.52E + 02 ± 1.24E + 01	3.06E + 03 ± 1.86E + 02	3.12E + 02 ± 1.24E + 01	2.84E + 02 ± 1.12E + 01	2.18E + 02 ± 7.89E + 00	4.68E + 02 ± 1.02E + 01
	30D	3×10^5	4.08E + 02 ± 1.14E + 01	1.93E + 02 ± 8.80E + 00	5.12E + 02 ± 1.68E + 01	1.90E + 02 ± 5.31E + 00	1.83E + 02 ± 4.36E + 00	1.57E + 02 ± 4.75E + 00	2.35E + 02 ± 5.47E + 00
f_{18}	10D	10^5	1.11E + 03 ± 2.26E + 01	3.95E + 02 ± 1.11E + 01	1.89E + 03 ± 5.70E + 01	3.57E + 02 ± 8.23E + 00	3.39E + 02 ± 8.29E + 00	2.80E + 02 ± 7.64E + 00	5.33E + 02 ± 8.07E + 00
	20D	2×10^5	2.31E + 03 ± 4.39E + 01	6.02E + 02 ± 1.59E + 01	4.77E + 03 ± 1.30E + 02	5.22E + 02 ± 1.37E + 01	4.93E + 02 ± 1.10E + 01	3.94E + 02 ± 9.44E + 00	8.39E + 02 ± 8.88E + 00
	30D	3×10^5	2.17E + 02 ± 1.03E + 01	1.04E + 02 ± 4.55E + 00	2.69E + 02 ± 1.37E + 01	1.01E + 02 ± 4.82E + 00	9.74E + 01 ± 4.45E + 00	8.26E + 01 ± 4.92E + 00	1.24E + 02 ± 6.16E + 00
f_{19}	10D	10^5	6.07E + 02 ± 1.66E + 01	2.16E + 02 ± 7.38E + 00	1.03E + 03 ± 4.26E + 01	1.94E + 02 ± 7.15E + 00	1.87E + 02 ± 6.10E + 00	1.54E + 02 ± 5.76E + 00	2.95E + 02 ± 6.46E + 00
	20D	2×10^5	1.38E + 03 ± 4.71E + 01	3.33E + 02 ± 1.19E + 01	2.59E + 03 ± 5.35E + 01	2.89E + 02 ± 9.29E + 00	2.73E + 02 ± 7.16E + 00	2.18E + 02 ± 6.53E + 00	4.76E + 02 ± 1.06E + 01
	30D	3×10^5	3.07E + 02 ± 8.64E + 00	1.45E + 02 ± 5.27E + 00	3.84E + 02 ± 1.49E + 01	1.44E + 02 ± 5.10E + 00	1.38E + 02 ± 5.36E + 00	1.16E + 02 ± 5.17E + 00	1.78E + 02 ± 6.31E + 00
f_{20}	10D	10^5	8.41E + 02 ± 2.20E + 01	2.99E + 02 ± 1.23E + 01	1.43E + 03 ± 4.08E + 01	2.70E + 02 ± 7.63E + 00	2.62E + 02 ± 6.50E + 00	2.12E + 02 ± 7.97E + 00	4.07E + 02 ± 1.02E + 01
	20D	2×10^5	1.74E + 03 ± 5.34E + 01	4.61E + 02 ± 1.76E + 01	3.59E + 03 ± 8.78E + 01	4.00E + 02 ± 1.09E + 01	3.77E + 02 ± 7.68E + 00	3.02E + 02 ± 7.37E + 00	6.48E + 02 ± 1.06E + 01
	30D	3×10^5	3.21E + 02 ± 1.14E + 01	1.54E + 02 ± 5.74E + 00	4.07E + 02 ± 1.78E + 01	1.54E + 02 ± 5.77E + 00	1.47E + 02 ± 5.88E + 00	1.25E + 02 ± 4.85E + 00	1.89E + 02 ± 4.21E + 00
f_{21}	10D	10^5	8.62E + 02 ± 1.85E + 01	3.10E + 02 ± 1.07E + 01	1.46E + 03 ± 4.60E + 01	2.80E + 02 ± 7.51E + 00	2.69E + 02 ± 8.40E + 00	2.20E + 02 ± 6.22E + 00	4.23E + 02 ± 6.09E + 00
	20D	2×10^5	1.77E + 03 ± 2.94E + 01	4.75E + 02 ± 1.38E + 01	3.64E + 03 ± 1.04E + 02	4.11E + 02 ± 1.07E + 01	3.87E + 02 ± 9.58E + 00	3.05E + 02 ± 8.38E + 00	6.68E + 02 ± 9.36E + 00
	30D	3×10^5	3.84E + 02 ± 1.48E + 01	1.70E + 02 ± 7.68E + 00	4.92E + 02 ± 2.60E + 01	1.65E + 02 ± 5.94E + 00	1.63E + 02 ± 5.52E + 00	1.37E + 02 ± 6.72E + 00	2.43E + 02 ± 1.45E + 01
f_{22}	10D	10^5	1.09E + 03 ± 3.17E + 01	3.65E + 02 ± 1.83E + 01	1.93E + 03 ± 5.76E + 01	3.40E + 02 ± 1.83E + 01	3.33E + 02 ± 2.85E + 01	2.98E + 02 ± 2.05E + 01	6.37E + 02 ± 6.63E + 01
	20D	2×10^5	2.37E + 03 ± 8.71E + 01	6.07E + 02 ± 3.64E + 01	4.99E + 03 ± 1.97E + 02	5.81E + 02 ± 4.08E + 01	5.49E + 02 ± 4.04E + 01	5.07E + 02 ± 3.02E + 01	1.10E + 03 ± 1.93E + 02
	30D	3×10^5	1.54E + 03 ± 1.17E + 03	1.03E + 03 ± 8.73E + 02	2.13E + 03 ± 1.98E + 03	9.50E + 02 ± 9.39E + 02	8.58E + 02 ± 1.05E + 03	1.27E + 03 ± 1.25E + 03	2.19E + 02 ± 1.61E + 02
f_{23}	10D	10^5	2.41E + 03 ± 1.94E + 03	1.97E + 03 ± 2.01E + 03	5.67E + 03 ± 4.37E + 03	1.64E + 03 ± 1.48E + 03	1.40E + 03 ± 1.53E + 03	1.42E + 03 ± 1.59E + 03	2.28E + 02 ± 1.45E + 02
	20D	2×10^5	3.83E + 03 ± 3.61E + 03	2.84E + 03 ± 2.76E + 03	4.12E + 04 ± 1.11E + 04	2.53E + 03 ± 2.72E + 03	2.34E + 03 ± 2.06E + 03	2.34E + 03 ± 1.89E + 03	2.80E + 02 ± 1.54E + 02
	30D	3×10^5	1.94E + 02 ± 6.83E + 00	8.57E + 01 ± 2.27E + 00	1.44E + 02 ± 5.16E + 01	1.14E + 02 ± 3.75E + 00	1.03E + 02 ± 5.04E + 00	9.76E + 01 ± 4.62E + 00	1.76E + 02 ± 4.43E + 00
f_{24}	10D	10^5	4.04E + 02 ± 9.17E + 00	1.60E + 02 ± 4.81E + 00	3.33E + 02 ± 1.02E + 01	2.11E + 02 ± 8.69E + 00	1.90E + 02 ± 7.16E + 00	1.83E + 02 ± 4.67E + 00	4.37E + 02 ± 8.56E + 00
	20D	2×10^5	6.39E + 02 ± 1.56E + 01	2.27E + 02 ± 6.36E + 00	5.74E + 02 ± 1.97E + 01	2.97E + 02 ± 7.23E + 00	2.64E + 02 ± 7.03E + 00	2.60E + 02 ± 7.35E + 00	7.17E + 02 ± 1.31E + 01
	30D	3×10^5	2.11E + 02 ± 7.63E + 00	9.13E + 01 ± 4.25E + 00	1.57E + 02 ± 6.62E + 00	1.22E + 02 ± 5.43E + 00	1.11E + 02 ± 3.99E + 00	1.04E + 02 ± 3.15E + 00	1.86E + 02 ± 5.67E + 00
f_{25}	10D	10^5	4.27E + 02 ± 1.29E + 01	1.67E + 02 ± 6.41E + 00	3.48E + 02 ± 1.24E + 01	2.20E + 02 ± 6.27E + 00	1.98E + 02 ± 5.03E + 00	1.89E + 02 ± 4.89E + 00	4.47E + 02 ± 9.85E + 00
	20D	2×10^5	6.67E + 02 ± 1.29E + 01	2.31E + 02 ± 6.24E + 00	6.01E + 02 ± 2.29E + 01	3.03E + 02 ± 8.41E + 00	2.68E + 02 ± 6.62E + 00	2.68E + 02 ± 6.37E + 00	7.28E + 02 ± 9.54E + 00
	30D	3×10^5	4.15E + 01 ± 7.49E + 00	2.65E + 01 ± 6.34E + 00	6.42E + 01 ± 1.25E + 01	2.84E + 01 ± 5.12E + 00	2.65E + 01 ± 3.68E + 00	2.12E + 01 ± 4.78E + 00	2.90E + 01 ± 4.75E + 00
f_{26}	10D	10^5	8.44E + 01 ± 1.75E + 01	4.05E + 01 ± 9.54E + 00	1.39E + 02 ± 3.52E + 01	5.31E + 01 ± 5.60E + 01	4.26E + 01 ± 2.76E + 01	3.12E + 01 ± 1.33E + 01	4.66E + 01 ± 7.03E + 00
	20D	2×10^5	1.46E + 02 ± 2.78E + 01	5.32E + 01 ± 1.02E + 01	3.45E + 02 ± 7.42E + 01	5.37E + 01 ± 1.13E + 01	4.65E + 01 ± 7.45E + 00	3.65E + 01 ± 5.91E + 00	6.69E + 01 ± 6.10E + 00
	30D	3×10^5	7.74E + 02 ± 7.73E + 01	2.63E + 02 ± 1.45E + 01	1.01E + 03 ± 1.36E + 02	6.25E + 02 ± 2.29E + 02	2.72E + 02 ± 2.20E + 01	2.31E + 02 ± 2.05E + 01	3.45E + 02 ± 2.14E + 01
f_{27}	10D	10^5	3.10E + 03 ± 4.42E + 02	5.39E + 02 ± 2.74E + 01	4.25E + 03 ± 8.10E + 02	8.94E + 02 ± 2.79E + 02	5.05E + 02 ± 2.77E + 01	4.09E + 02 ± 2.27E + 01	7.92E + 02 ± 4.47E + 01
	20D	2×10^5	7.68E + 03 ± 1.37E + 03	8.10E + 02 ± 3.98E + 01	1.53E + 04 ± 2.16E + 03	1.08E + 03 ± 3.05E + 02	7.16E + 02 ± 3.18E + 01	5.49E + 02 ± 2.28E + 01	1.25E + 03 ± 5.70E + 01
	30D	3×10^5	2.02E + 03 ± 1.97E + 02	4.01E + 02 ± 5.92E + 01	2.43E + 04 ± 1.39E + 04	4.14E + 02 ± 3.94E + 03	9.85E + 02 ± 1.42E + 01	5.94E + 02 ± 2.49E + 01	5.94E + 02 ± 2.49E + 01
f_{28}	10D	10^5	9.21E + 02 ± 1.05E + 02	—	—	1.62E + 03 ± 1.45E + 03	8.08E + 03 ± 3.38E + 04	5.12E + 02 ± 1.81E + 01	2.68E + 03 ± 1.19E + 02
	20D	2×10^5	—	—	—	9.55E + 03 ± 2.29E + 04	2.55E + 04 ± 6.30E + 04	7.79E + 02 ± 2.87E + 01	8.54E + 03 ± 5.10E + 02
	30D	3×10^5	3.16E + 01 ± 3.55E + 01	2.60E + 01 ± 1.96E + 01	3.81E + 01 ± 4.75E + 01	3.27E + 01 ± 3.60E + 01	3.93E + 01 ± 4.95E + 01	3.42E + 01 ± 3.39E + 01	7.96E + 01 ± 9.05E + 00
f_{29}	10D	10^5	2.01E + 01 ± 1.80E + 01	3.34E + 01 ± 3.33E + 01	3.09E + 01 ± 3.12E + 01	3.44E + 01 ± 3.38E + 01	3.12E + 01 ± 3.40E + 01	3.45E + 01 ± 3.05E + 01	1.80E + 02 ± 2.73E + 02
	20D	2×10^5	3.46E + 01 ± 2.25E + 01	1.86E + 01 ± 1.66E + 01	2.84E + 01 ± 2.43E + 01	4.05E + 01 ± 3.80E + 01	2.87E + 01 ± 2.63E + 01	3.05E + 01 ± 2.44E + 01	2.24E + 02 ± 2.44E + 02
	30D	3×10^5	1.08E + 02 ± 6.26E + 01	7.48E + 01 ± 5.54E + 01	1.47E + 02 ± 6.36E + 01	9.35E + 01 ± 5.22E + 01	7.79E + 01 ± 5.93E + 01	6.68E + 01 ± 2.77E + 01	3.11E + 01 ± 8.45E + 00
f_{30}	10D	10^5	1.12E + 01 ± 5.90E + 00	1.06E + 01 ± 4.91E + 00	1.08E + 01 ± 5.40E + 00	1.09E + 01 ± 5.38E + 00	1.17E + 01 ± 6.30E + 00	1.12E + 01 ± 6.17E + 00	1.34E + 01 ± 5.17E + 00
	20D	2×10^5	—	—	—	—	—	—	—
	30D	3×10^5	—	—	—	—	—	—	—

performing mutation strategy in average fitness value and is runner-up in most of the cases of number of function call values. One interesting aspect of this variant (V_{10}) is that the performance of this strategy is never announced to be one the best strategies of the DE algorithm. The mutation strategies V_7 and V_2 (“DE/best/1/bin”) that are already known to be best performing DE variant have subsequent best performance along with exponential “TSDE(V_{42})” and vice versa. It can also be summarized that the worst performing mutation strategies among the considered strategies are V_6 (“DE/rand repeating & current to rand/1/bin”), V_8 , V_{11} (DE/rand to best/1), V_{14} (“DE/rand to best/2”) V_{16} (“DE/rand to current/2”), V_{19} (“DE/rand/3”), and V_{20} (“DE/best/3”) and vice versa.

7. Statistical Analysis

The results of DE mutation strategies are analyzed statistically by calculating the rank correlation and by generated Dendrograms. To find the rank correlation, rank of each DE mutation strategy is calculated to find the correlation between DE strategies. The performance of mutation strategies will be observed during the analysis for various dimensions with the help of Dendrogram. Similarity/dissimilarity matrix is used to generate the Dendrogram. Similarity matrixes are obtained using “Statistical Package for the Social Sciences” (SPSS). After similarity matrix, pairwise distances between DE strategies are calculated. Pairwise distance is used to create hierarchical cluster tree linkage that is used to create the Dendrogram. Dendrogram actually generates the clusters from the dissimilarity matrix using agglomerative clustering. Dendrograms of DE mutation strategy for test suit of benchmark functions are shown in Figures 1–3 and Algorithm 1 for different dimensions.

Pairwise dissimilarities between objects can be easily depicted using Dendrograms. Dendrograms are considered to be an effective way of cluster analysis that specifies the level of similarity joining of any two clusters. The leaf of clusters in Dendrograms contains the DE mutation strategy’s number and y -axis contains the dissimilarity scale that shows the joining distance between clusters. The height of each branch in Dendrograms shows that which DE strategies are similar or different from each other. The greater height of DE strategies branches means that strategies are more different from each other. Consider the Dendrograms generated from average fitness values and number of function call values given in Figures 2–9, respectively. A similarity measure among the DE strategies is explored by generating Dendrograms from the average fitness values and number of function call performance parameters. The best performing mutation strategies are contained in the similar performing cluster shown in transparent circles and most worst performing strategies are contained in the dotted circle in each Dendrogram. The Dendrograms are generated for 10D, 20D, 30D and the average of all (10D, 20D, and 30D) is generated for both performance metrics of fitness value and number of function calls. The best performing and worst performing strategies among all mutation strategies are contained in the corresponding groups in most of the cases. The best performing strategies “TSDE(V_{41})”, “DE/rand

repeated to best/1/bin (V_{10})”, “DE/best/1/bin (V_2)”, and “(V_7)” are contained in the same group in most of the cases that can be considered as best performing mutation strategies that are group shown in the transparent circle in Figures 2–9. The worst performing group of mutation strategies is encircled dashed line and it contains “DE/rand repeating & current to rand/1/bin (V_6)”, “DE/rand & current to best/2/bin (V_{14})”, “DE/rand & current to rand/1/bin (V_{16})”, “DE/rand/3/bin (V_{19})”, and “DE/best/3 (V_{20})” and vice versa.

It is important to discuss that a number of DE mutation strategies are considered in this research work and all these strategies perform different from each other. Dendrograms are also considered to be an effective tool to determine the clusters number. For DE strategies, more number of clusters in the Dendrogram indicates that DE mutation strategies have varying performance from one another, which shows the diverse nature of these strategies in DE algorithm. The diverse nature of DE variants will be helpful to make this algorithm more broadly applicable in various optimization problems of distinct characteristics.

The convergence graphs of selected benchmark functions for 10 dimensions (10D) are given in Figure 10. Logarithmic convergence graphs of average fitness show number of iterations horizontally and performance vertically. Horizontal iterations are shown as a multiple of 100. Figure 10 contains the convergence graphs of five commonly used DE mutation strategies (V_1 , V_2 , V_3 , V_4 , and V_7), the proposed mutation variants (V_{41} and V_{42}), and one other better performing variant V_{10} . Convergence graphs of few selected functions are given in the paper. Convergence graph shows that the proposed “TSDE(V_{41})” mutation variant has better performance in most of the cases and proposed that “TSDE(V_{42})” has better performance in few cases. The mutation strategy V_{10} (“DE/rand to best/1/bin”) proves itself to be another better performing DE strategy; it is following “TSDE (V_{41})” in most cases of convergence graphs. It is important to mention here that V_{10} (“DE/rand to best/1/bin”) strategy has never been declared to be one of the best performing variants or among top performing variants of DE algorithm. The convergence graph clearly shows the dominating performance of the proposed DE mutation variant “TSDE(V_{41})”. Most of the researchers have concentrated on very few basic DE strategies like “DE/rand/1/bin (V_1)”, “DE/best/1/bin (V_2)”, “DE/rand/2/bin (V_3)”, “DE/best/2/bin (V_4)”, and so forth, and some key performing mutation strategies remained unnoticed in the DE research. This study will prove to be fruitful for DE researchers having choice of different variants for scientific and real time applications.

8. Comparison of TSDE with PSO and GA

PSO and GA are two well known metaheuristics that are applied on variety of real world problems [57]. The simulation results of proposed “TSDE(V_{41})” is compared with two other well known metaheuristics: particle swarm optimization and genetic algorithm. In order to compare TSDE with PSO and GA, the parameter setting for GA is crossover rate $CR = 0.5$ and mutation rate $F = 0.05$ [58] while for PSO inertia weight $\omega \in (0.4, 0.7)$, $C1 = 1.49618$, and $C2 = 1.49618$ are used [59].

TABLE 5: Fitness results (mean \pm SD) of TSDE(V_{41}), PSO, and GA for functions (f_1-f_{30}).

Fun	10D			20D			30D		
	TSDE(V_{41})	PSO	GA	TSDE(V_{41})	PSO	GA	TSDE(V_{41})	PSO	GA
f_1	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	3.95E-09 \pm 1.90E-08	0.00E+00 \pm 0.00E+00	6.63E-176 \pm 0.00E+00	1.17E-04 \pm 2.77E-04	0.00E+00 \pm 0.00E+00	5.24E-01 \pm 3.67E+00	2.48E-05 \pm 6.55E-05
f_2	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	3.55E-06 \pm 8.35E-06	0.00E+00 \pm 0.00E+00	4.19E+00 \pm 1.32E+01	1.05E-03 \pm 2.25E-03	0.00E+00 \pm 0.00E+00	3.15E-01 \pm 6.14E+01	1.98E-04 \pm 6.17E-04
f_3	9.92E-120 \pm 4.77E-118	1.01E-186 \pm 0.00E+00	1.99E-01 \pm 3.35E-01	1.36E-47 \pm 6.90E-47	4.23E+01 \pm 2.96E+02	2.23E+00 \pm 2.56E+00	4.00E-26 \pm 6.90E-26	2.96E+02 \pm 8.46E+02	7.18E+00 \pm 7.83E+00
f_4	2.66E-01 \pm 7.16E-01	2.86E+01 \pm 6.47E+01	7.25E+00 \pm 1.43E+01	1.33E-01 \pm 7.16E-01	8.18E+05 \pm 7.48E+05	3.82E+01 \pm 6.52E+01	3.99E-01 \pm 1.20E+00	4.13E+06 \pm 3.62E+06	3.16E+01 \pm 5.14E+01
f_5	1.53E+00 \pm 1.17E+00	5.61E+00 \pm 6.35E+00	3.98E-08 \pm 2.62E-07	6.33E+00 \pm 2.03E+00	2.98E+01 \pm 2.81E+01	4.17E-05 \pm 8.59E-05	1.34E+01 \pm 3.64E+00	6.09E+01 \pm 5.48E+01	6.11E-06 \pm 3.31E-05
f_6	1.44E-02 \pm 1.53E-02	4.54E-02 \pm 4.41E-02	4.40E-02 \pm 5.85E-02	2.96E-03 \pm 4.82E-03	2.46E-02 \pm 3.46E-02	1.41E-01 \pm 1.57E-01	4.93E-04 \pm 1.84E-03	3.71E+00 \pm 1.78E+01	1.64E-01 \pm 2.07E-01
f_7	0.00E+00 \pm 0.00E+00	1.32E-295 \pm 0.00E+00	4.40E-06 \pm 6.35E-06	0.00E+00 \pm 0.00E+00	6.35E-82 \pm 4.45E-81	2.29E-06 \pm 2.95E-06	0.00E+00 \pm 0.00E+00	6.29E-61 \pm 4.40E-60	8.24E-07 \pm 1.23E-06
f_8	9.79E-02 \pm 1.01E-01	3.36E-01 \pm 3.30E-01	2.44E-07 \pm 4.00E-07	2.32E-01 \pm 6.28E-02	6.48E-01 \pm 5.75E-01	1.98E-05 \pm 4.47E-05	2.73E-01 \pm 5.21E-02	1.06E+00 \pm 9.64E-01	4.34E-06 \pm 1.43E-05
f_9	3.31E-07 \pm 3.83E-07	6.09E-18 \pm 2.33E-17	2.50E-04 \pm 5.86E-04	1.55E-07 \pm 1.44E-07	1.27E-02 \pm 3.84E-02	5.14E-02 \pm 7.54E-02	1.86E-03 \pm 1.00E-02	6.33E-01 \pm 1.99E+00	2.62E-01 \pm 2.85E-01
f_{10}	2.11E-151 \pm 5.43E-151	2.82E-220 \pm 0.00E+00	6.17E-03 \pm 1.64E-02	3.83E-71 \pm 6.60E-71	2.10E-94 \pm 1.43E-93	2.96E-01 \pm 4.39E-01	1.11E-44 \pm 4.40E-44	4.69E-30 \pm 3.13E-29	2.78E-01 \pm 3.35E-01
f_{11}	7.33E-202 \pm 0.00E+00	1.73E-60 \pm 1.21E-59	5.54E-30 \pm 3.36E-29	1.67E-232 \pm 0.00E+00	4.00E-01 \pm 1.96E+00	1.33E-03 \pm 4.88E-03	1.60E-257 \pm 0.00E+00	9.05E-01 \pm 2.99E+00	8.61E-06 \pm 5.45E-05
f_{12}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	6.00E-02 \pm 2.37E-01	0.00E+00 \pm 0.00E+00	3.14E+00 \pm 1.01E+01	9.60E-01 \pm 1.51E+00	0.00E+00 \pm 0.00E+00	3.21E+01 \pm 7.21E+01	4.46E+00 \pm 4.50E+00
f_{13}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	2.39E-12 \pm 7.46E-12	0.00E+00 \pm 0.00E+00	5.37E-02 \pm 3.76E-01	5.08E-09 \pm 1.60E-08	0.00E+00 \pm 0.00E+00	5.37E-01 \pm 2.01E+00	2.61E-10 \pm 1.00E-09
f_{14}	3.39E-05 \pm 5.08E-05	6.74E-16 \pm 1.36E-15	4.78E-05 \pm 1.03E-04	3.05E-05 \pm 3.73E-05	4.74E-15 \pm 2.00E-14	5.82E-05 \pm 1.13E-04	1.79E-05 \pm 1.77E-05	2.88E-12 \pm 2.02E-11	3.60E-05 \pm 1.26E-04
f_{15}	3.27E-31 \pm 1.31E-46	1.24E-02 \pm 6.09E-02	5.33E-03 \pm 6.89E-03	1.63E-31 \pm 6.57E-47	6.88E-02 \pm 2.70E-01	1.23E-02 \pm 1.24E-02	3.46E-03 \pm 1.86E-02	1.88E-01 \pm 4.06E-01	4.24E-02 \pm 5.34E-02
f_{16}	8.53E+01 \pm 7.11E-14	4.48E+01 \pm 2.24E+02	5.55E+01 \pm 4.70E+01	6.73E-05 \pm 7.35E-05	2.19E+01 \pm 1.15E+02	5.26E+01 \pm 6.25E+01	7.43E-05 \pm 7.72E-05	3.94E+05 \pm 3.32E+05	2.32E+02 \pm 3.02E+02
f_{17}	0.00E+00 \pm 0.00E+00	2.36E-02 \pm 6.17E-02	5.72E-12 \pm 1.49E-11	0.00E+00 \pm 0.00E+00	2.42E-01 \pm 2.57E-01	2.08E-06 \pm 9.39E-06	1.97E-02 \pm 5.02E-02	6.59E-01 \pm 6.66E-01	1.67E-07 \pm 5.35E-07
f_{18}	0.00E+00 \pm 0.00E+00	2.63E+05 \pm 2.37E+05	8.44E-03 \pm 4.42E-02	0.00E+00 \pm 0.00E+00	2.62E+06 \pm 2.24E+06	1.99E+01 \pm 4.60E+01	0.00E+00 \pm 0.00E+00	5.62E+06 \pm 4.63E+06	1.78E+00 \pm 4.46E+00
f_{19}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.84E-08 \pm 1.10E-07	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	4.70E-05 \pm 9.73E-05	0.00E+00 \pm 0.00E+00	7.27E-01 \pm 3.56E+00	2.36E-05 \pm 1.53E-04
f_{20}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	9.51E-06 \pm 4.62E-05	0.00E+00 \pm 0.00E+00	3.74E-202 \pm 0.00E+00	1.96E-02 \pm 4.10E-02	0.00E+00 \pm 0.00E+00	9.98E-05 \pm 6.86E-04	7.45E-03 \pm 3.35E-02
f_{21}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.20E-03 \pm 5.65E-03	0.00E+00 \pm 0.00E+00	3.85E-134 \pm 2.70E-133	1.24E-01 \pm 2.93E-01	0.00E+00 \pm 0.00E+00	7.66E-09 \pm 3.70E-08	1.03E-01 \pm 6.10E-01
f_{22}	0.00E+00 \pm 3.38E-32	0.00E+00 \pm 0.00E+00	5.05E-01 \pm 5.19E-01	1.51E-31 \pm 1.56E-31	1.19E-07 \pm 7.97E-07	2.93E+00 \pm 2.84E+00	6.86E-31 \pm 4.01E-31	4.79E+04 \pm 5.51E+04	5.45E+00 \pm 4.63E+00
f_{23}	1.56E-06 \pm 1.32E-06	1.92E+00 \pm 1.66E+00	7.67E-03 \pm 1.90E-02	1.57E-06 \pm 1.36E-06	5.64E+00 \pm 4.69E+00	6.95E-02 \pm 1.17E-01	1.36E-06 \pm 1.30E-06	9.25E+00 \pm 7.67E+00	1.71E+00 \pm 1.55E+00
f_{24}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	7.15E-01 \pm 1.64E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.21E+02 \pm 4.28E+02	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	2.95E+04 \pm 1.43E+05
f_{25}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	1.16E+00 \pm 1.68E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	5.86E+01 \pm 1.54E+02	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	5.10E+04 \pm 1.61E+05
f_{26}	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00	0.00E+00 \pm 0.00E+00
f_{27}	7.35E-118 \pm 1.46E-117	8.31E-15 \pm 1.98E-16	1.63E+00 \pm 1.48E+00	3.42E-15 \pm 3.81E-16	2.07E-01 \pm 1.39E+00	5.35E+00 \pm 4.52E+00	5.13E-15 \pm 2.20E-15	1.62E+00 \pm 3.02E+00	1.50E+00 \pm 1.25E+01
f_{28}	9.43E-08 \pm 7.09E-07	1.87E-29 \pm 3.05E-29	8.32E-10 \pm 2.89E-09	1.11E-07 \pm 4.93E-07	9.70E-01 \pm 1.04E+00	4.70E-19 \pm 3.29E-18	7.04E-120 \pm 7.93E-120	3.43E+00 \pm 3.27E+00	4.27E-235 \pm 0.00E+00
f_{29}	2.10E-04 \pm 4.68E-06	4.02E-04 \pm 3.78E-04	1.68E-03 \pm 3.62E-03	2.64E-08 \pm 4.35E-09	1.65E-29 \pm 2.73E-29	3.25E-09 \pm 9.17E-09	2.04E-07 \pm 9.92E-07	2.94E-29 \pm 4.86E-29	2.43E-10 \pm 3.39E-10
f_{30}					5.36E-08 \pm 4.59E-08	1.13E-04 \pm 2.58E-04	2.54E-12 \pm 6.83E-13	4.83E-12 \pm 4.22E-12	2.39E-05 \pm 7.86E-05

TABLE 6: Number of function calls (mean \pm SD) of $TSDE(V_{41})$, PSO, and GA for functions ($f_1 - f_{30}$).

Fun	10D			20D			30D		
	$TSDE(V_{41})$	PSO	GA	$TSDE(V_{41})$	PSO	GA	$TSDE(V_{41})$	PSO	GA
f_1	8.38E+01 \pm 4.86E+00	8.70E+01 \pm 6.98E+00	1.94E+03 \pm 7.64E+02	1.56E+02 \pm 5.21E+00	1.78E+02 \pm 2.56E+01	1.22E+04 \pm 3.35E+03	2.21E+02 \pm 7.02E+00	3.74E+02 \pm 9.70E+01	1.23E+04 \pm 2.88E+03
f_2	8.47E+01 \pm 3.34E+00	8.86E+01 \pm 6.04E+00	4.47E+03 \pm 2.31E+03	1.69E+02 \pm 5.48E+00	2.00E+02 \pm 3.74E+01	2.54E+04 \pm 8.90E+03	2.46E+02 \pm 7.61E+00	4.81E+02 \pm 1.64E+02	1.81E+04 \pm 5.80E+03
f_3	3.30E+02 \pm 1.91E+01	2.18E+02 \pm 2.01E+01	5.64E+04 \pm 1.68E+04	1.72E+03 \pm 1.05E+02	9.45E+02 \pm 1.15E+02	—	4.57E+03 \pm 2.43E+02	2.80E+03 \pm 3.27E+02	2.11E+05 \pm 3.81E+04
f_4	1.23E+03 \pm 5.85E+01	3.39E+04 \pm 1.33E+04	—	2.84E+03 \pm 4.03E+02	—	—	4.78E+03 \pm 5.89E+02	—	—
f_5	5.21E+02 \pm 1.85E+02	—	2.04E+03 \pm 6.31E+02	—	—	9.05E+03 \pm 2.46E+03	—	—	1.27E+04 \pm 4.54E+03
f_6	3.92E+02 \pm 2.44E+02	1.90E+04 \pm 2.19E+04	1.90E+04 \pm 2.19E+04	2.36E+02 \pm 1.67E+01	2.53E+02 \pm 3.29E+01	7.26E+04 \pm 4.95E+04	3.17E+02 \pm 1.14E+01	5.63E+02 \pm 4.44E+01	4.48E+04 \pm 2.83E+04
f_7	4.50E+01 \pm 3.38E+00	5.85E+01 \pm 7.33E+00	3.59E+03 \pm 2.77E+03	6.14E+01 \pm 4.93E+00	7.21E+01 \pm 8.89E+00	3.70E+03 \pm 4.44E+03	7.06E+01 \pm 5.21E+00	8.74E+01 \pm 1.23E+01	2.05E+03 \pm 3.22E+03
f_8	5.33E+02 \pm 1.32E+02	—	2.36E+03 \pm 7.9E+02	—	—	1.11E+04 \pm 3.32E+03	—	—	1.20E+04 \pm 4.23E+03
f_9	2.40E+02 \pm 1.66E+02	3.97E+04 \pm 2.96E+04	2.24E+02 \pm 1.17E+02	1.13E+02 \pm 2.21E+01	1.13E+02 \pm 2.21E+01	1.08E+05 \pm 4.20E+04	2.53E+02 \pm 1.22E+02	1.47E+02 \pm 0.00E+00	—
f_{10}	2.68E+02 \pm 1.31E+01	1.77E+02 \pm 1.61E+01	3.24E+04 \pm 1.94E+04	1.26E+03 \pm 5.44E+01	8.96E+02 \pm 4.40E+02	—	2.96E+03 \pm 1.28E+02	2.98E+03 \pm 7.84E+02	2.72E+05 \pm 3.36E+03
f_{11}	1.61E+02 \pm 4.08E+00	1.77E+02 \pm 1.49E+01	6.93E+02 \pm 6.42E+02	2.96E+02 \pm 6.82E+00	7.51E+02 \pm 4.64E+02	1.30E+04 \pm 9.92E+03	4.17E+02 \pm 1.15E+01	—	4.72E+03 \pm 7.16E+03
f_{12}	5.66E+01 \pm 5.33E+00	8.13E+01 \pm 9.73E+01	2.52E+03 \pm 1.36E+03	1.07E+02 \pm 5.02E+00	2.44E+04 \pm 4.54E+04	2.00E+04 \pm 9.05E+03	1.50E+02 \pm 7.71E+00	1.08E+05 \pm 1.40E+05	8.14E+04 \pm 4.37E+04
f_{13}	3.69E+01 \pm 4.15E+00	3.50E+01 \pm 3.94E+00	1.28E+02 \pm 6.27E+01	8.27E+01 \pm 4.63E+00	8.68E+01 \pm 8.17E+00	7.40E+02 \pm 3.59E+02	1.29E+02 \pm 6.28E+00	1.70E+02 \pm 2.21E+01	2.00E+03 \pm 9.29E+02
f_{14}	1.80E+04 \pm 1.58E+04	1.34E+02 \pm 3.92E+01	2.12E+04 \pm 2.55E+04	1.91E+04 \pm 1.98E+04	1.53E+02 \pm 5.99E+01	4.70E+04 \pm 4.92E+04	3.93E+04 \pm 2.82E+04	2.11E+02 \pm 2.94E+02	4.33E+04 \pm 6.18E+04
f_{15}	8.10E+01 \pm 5.75E+00	8.20E+01 \pm 7.37E+00	—	1.44E+02 \pm 6.32E+00	2.05E+02 \pm 7.46E+01	—	1.98E+02 \pm 1.01E+01	7.29E+02 \pm 6.37E+02	—
f_{16}	—	3.83E+02 \pm 5.45E+01	—	4.42E+04 \pm 4.67E+04	2.62E+03 \pm 1.89E+03	—	1.18E+05 \pm 7.21E+04	—	—
f_{17}	8.36E+01 \pm 3.87E+00	8.82E+01 \pm 8.18E+00	7.28E+02 \pm 3.00E+02	1.58E+02 \pm 7.40E+00	1.54E+02 \pm 7.07E+01	7.18E+03 \pm 2.89E+03	2.18E+02 \pm 7.89E+00	—	8.88E+03 \pm 3.05E+03
f_{18}	1.57E+02 \pm 4.75E+00	—	5.59E+03 \pm 1.05E+03	2.80E+02 \pm 7.64E+00	—	2.89E+04 \pm 3.68E+03	3.94E+02 \pm 9.44E+00	—	3.14E+04 \pm 5.82E+03
f_{19}	8.26E+01 \pm 4.92E+00	8.38E+01 \pm 5.33E+00	1.87E+03 \pm 5.35E+02	1.54E+02 \pm 7.66E+00	1.76E+02 \pm 2.02E+01	1.10E+04 \pm 3.49E+03	2.18E+02 \pm 6.55E+00	4.70E+02 \pm 3.24E+02	1.25E+04 \pm 2.75E+03
f_{20}	1.16E+02 \pm 5.17E+00	1.21E+02 \pm 5.81E+00	3.38E+03 \pm 1.00E+03	2.12E+02 \pm 7.97E+00	2.49E+02 \pm 2.46E+01	2.05E+04 \pm 4.16E+03	3.02E+02 \pm 7.37E+00	6.98E+02 \pm 3.67E+02	2.11E+04 \pm 4.24E+03
f_{21}	1.25E+02 \pm 4.85E+00	1.51E+02 \pm 8.11E+00	4.88E+03 \pm 1.05E+03	2.20E+02 \pm 6.22E+00	2.74E+02 \pm 2.99E+01	2.53E+04 \pm 5.45E+03	3.05E+02 \pm 8.38E+00	8.12E+02 \pm 4.91E+02	2.59E+04 \pm 6.54E+03
f_{22}	1.37E+02 \pm 6.72E+00	1.64E+02 \pm 1.14E+01	—	2.98E+02 \pm 2.05E+01	3.01E+03 \pm 1.52E+03	—	5.07E+02 \pm 3.02E+01	2.81E+05 \pm 0.00E+00	—
f_{23}	1.27E+03 \pm 1.25E+03	—	6.54E+04 \pm 1.43E+04	1.42E+03 \pm 1.59E+03	—	9.50E+04 \pm 0.00E+00	2.34E+03 \pm 1.89E+03	—	—
f_{24}	9.76E+01 \pm 4.62E+00	3.03E+00 \pm 3.20E+01	8.16E+03 \pm 3.32E+03	1.83E+02 \pm 4.67E+00	3.27E+00 \pm 5.21E+01	3.26E+04 \pm 1.31E+04	2.60E+02 \pm 7.35E+00	3.40E+00 \pm 4.98E+01	6.19E+04 \pm 1.94E+04
f_{25}	1.04E+02 \pm 3.15E+00	3.13E+00 \pm 3.46E+01	1.03E+04 \pm 3.17E+03	1.89E+02 \pm 4.89E+00	3.37E+00 \pm 5.56E+01	3.21E+04 \pm 6.75E+03	2.68E+02 \pm 6.37E+00	3.57E+00 \pm 5.04E+01	7.08E+04 \pm 2.16E+04
f_{26}	2.12E+01 \pm 4.78E+00	3.03E+00 \pm 1.83E+01	2.74E+01 \pm 9.37E+00	3.12E+01 \pm 1.33E+01	3.00E+00 \pm 0.00E+00	3.24E+01 \pm 1.60E+01	3.65E+01 \pm 5.91E+00	2.87E+00 \pm 3.46E+01	1.57E+01 \pm 6.78E+00
f_{27}	2.31E+02 \pm 2.05E+01	2.18E+02 \pm 2.38E+01	—	4.09E+02 \pm 2.27E+01	9.27E+02 \pm 2.96E+02	—	5.49E+02 \pm 2.28E+01	—	—
f_{28}	2.53E+02 \pm 1.42E+01	—	1.31E+03 \pm 9.83E+02	5.12E+02 \pm 1.81E+01	—	1.59E+03 \pm 8.26E+02	7.79E+02 \pm 2.87E+01	—	1.30E+03 \pm 8.97E+02
f_{29}	3.42E+01 \pm 3.39E+01	2.23E+01 \pm 1.53E+01	4.80E+01 \pm 5.54E+01	3.45E+01 \pm 3.05E+01	2.38E+01 \pm 1.36E+01	2.19E+02 \pm 5.88E+02	3.05E+01 \pm 2.44E+01	1.61E+01 \pm 1.33E+01	8.63E+01 \pm 1.19E+02
f_{30}	—	—	—	6.68E+01 \pm 2.77E+01	1.16E+01 \pm 8.32E+00	3.41E+04 \pm 5.31E+04	1.12E+01 \pm 6.17E+00	3.03E+00 \pm 1.03E+00	2.85E+03 \pm 8.15E+03

For fair comparison between PSO, GA, and TSDE, the population size, dimensions, number of iterations, and standard benchmark functions are kept the same. Results of average fitness value and NFC are presented in Tables 5 and 6, respectively. Proposed strategy TSDE has dominating performance than evolutionary metaheuristics PSO and GA in most of the cases of 10D, 20D, and 30D for average fitness value and NFC. TSDE proves itself to be one of the powerful variants of DE algorithm that can be a significant addition in DE research.

9. Conclusion

In this paper, a novel technique TSDE mutation variant based on the selection of parent vectors to generate the new population is proposed. The proposed variants “ $TSDE(V_{41})$ ” and “ $TSDE(V_{42})$ ” are compared with the existing DE mutation strategies. $TSDE(V_{41})$ mutation variant has dominating performance among the DE mutation strategies. This new mutation variant TSDE will prove to be a valuable addition in DE literature. A comprehensive set of well known N -dimensional benchmark functions have been used to evaluate the performance of the proposed TSDE as well as existing mutation strategies of DE algorithm. In literature, such an extensive comparison performed with many number of mutation strategies is lacking. This comparison unveils some very beneficial facts related to the DE research improvements. The other important aspect of this research is the statistical comparison of the DE mutation strategies; such a comparison is also not available in the literature. Experimental results consist of average fitness value and number of function call parameters of DE algorithm that are reported in Tables 3 and 4. Since total strategies are 42 and it is not possible to report all the mutation strategies in few tables, so proposed TSDE mutation variant, five commonly used mutation strategies, and one better performing mutation strategy are reported in the results. It can be concluded that there is a deviation in the performance of DE algorithm variants and the selection of DE variant affects the performance result of DE algorithm. Better results can be obtained by choosing the better performing strategies. The proposed binomial “ $TSDE(V_{41})$ ” has the leading performance among the DE mutation strategies for both fitness value and number of function call performance parameters, the proposed exponential “ $TSDE(V_{42})$,” mutation strategy “ $DE/rand$ to $best/1/bin$ (V_{10}),” “ V_7 ” and “ $DE/best/1/bin$ (V_2)” are the other better performance mutation strategies. This research work also reveals one of the best performing mutation strategies “ $DE/rand$ to $best/1/bin$ (V_{10})” that has been rarely brought into play. This research work will prove to be a significant addition in DE literature. The results of TSDE are also compared with two well known heuristics, GA and PSO, that show dominating performance of TSDE. The further studies are still required to explore the limitations, advantages, and flaws in this direction. The main focus in this work is not the overwhelming and existing DE variants but new tracks to work on the DE variants in optimization. One possible direction of future work of this paper is to develop a mathematical model for the DE mutation strategies.

Conflict of Interests

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

References

- [1] R. Storn and K. Price, “Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces,” Tech. Rep. TR-95-012, ICSI, Berkeley, Calif, USA, 1995.
- [2] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, Natural Computing Series, Springer, New York, NY, USA, 1st edition, 2005.
- [3] Y. Lu, J. Zhou, H. Qin, Y. Li, and Y. Zhang, “An adaptive hybrid differential evolution algorithm for dynamic economic dispatch with valve-point effects,” *Expert Systems with Applications*, vol. 37, no. 7, pp. 4842–4849, 2010.
- [4] I. L. Lopez Cruz, L. G. van Willigenburg, and G. van Straten, “Efficient differential evolution algorithms for multimodal optimal control problems,” *Applied Soft Computing Journal*, vol. 3, no. 2, pp. 97–122, 2003.
- [5] A. Qing, “Dynamic differential evolution strategy and applications in electromagnetic inverse scattering problems,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 1, pp. 116–125, 2006.
- [6] R. Burman, S. Paul, and S. Das, “A differential evolution approach to multi-level image thresholding using type II fuzzy sets,” in *Swarm, Evolutionary, and Memetic Computing*, vol. 8297 of *Lecture Notes in Computer Science*, pp. 274–285, Springer, 2013.
- [7] R. Xu, G. K. Venayagamoorthy, and D. C. Wunsch II, “Modeling of gene regulatory networks with hybrid differential evolution and particle swarm optimization,” *Neural Networks*, vol. 20, no. 8, pp. 917–927, 2007.
- [8] S.-T. Pan, B.-Y. Tsai, and C.-S. Yang, “Differential evolution algorithm on robust IIR filter design and implementation,” in *Proceedings of the 8th International Conference on Intelligent Systems Design and Applications (ISDA '08)*, pp. 537–542, Kaohsiung, Taiwan, November 2008.
- [9] R. Angira and B. V. Babu, “Optimization of process synthesis and design problems: a modified differential evolution approach,” *Chemical Engineering Science*, vol. 61, no. 14, pp. 4707–4721, 2006.
- [10] 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.
- [11] A. K. Qin, V. L. Huang, and P. N. Suganthan, “Differential evolution algorithm with strategy adaptation for global numerical optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2009.
- [12] J. Zhang and A. C. Sanderson, “JADE: adaptive differential evolution with optional external archive,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 945–958, 2009.
- [13] R. Mallipeddi, P. N. Suganthan, Q. K. Pan, and M. F. Tasgetiren, “Differential evolution algorithm with ensemble of parameters and mutation strategies,” *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1679–1696, 2011.
- [14] S. M. Islam, S. Das, S. Ghosh, S. Roy, and P. N. Suganthan, “An adaptive differential evolution algorithm with novel mutation

- and crossover strategies for global numerical optimization,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 2, pp. 482–500, 2012.
- [15] K. Zielinski, M. Joost, R. Laur, and B. Orlik, “Choosing suitable variants of differential evolution and particle swarm optimization for the optimization of a PI cascade control,” in *Proceedings of the 11th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM '08)*, pp. 55–62, May 2008.
- [16] V. L. Huang, A. K. Qin, and P. N. Suganthan, “Self-adaptive differential evolution algorithm for constrained real-parameter optimization,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '06)*, pp. 17–24, Vancouver, Canada, July 2006.
- [17] R. Storn, “On the usage of differential evolution for function optimization,” in *Proceedings of the Biennial Conference of the North American Fuzzy Information Processing Society (NAFIPS '96)*, pp. 519–523, Berkeley, Calif, USA, 1996.
- [18] Y. Ao and H. Chi, “Dynamic differential evolution for constrained real-parameter optimization,” *Journal of Advances in Information Technology*, vol. 1, no. 1, pp. 43–51, 2010.
- [19] G. T. S. De Oliveira and S. F. P. Saramago, “A contribution to the study about differential evolution,” *Ciência & Engenharia*, vol. 16, pp. 1–8, 2007.
- [20] T. Podoba, L. Tomsu, K. Vlcek, and M. Heczko, “Surface reconstruction by means of AI,” in *Proceedings of the International Conference on Computational Experimental Engineering and Sciences*, pp. 111–121, 2010.
- [21] I. Zelinka, R. Senkerik, M. Bialic-Davendra, and D. Davendra, “Chaos driven evolutionary algorithm for the traveling salesman problem,” in *Traveling Salesman Problem, Theory and Applications*, D. Davendra, Ed., chapter 4, pp. 55–70, InTech, Rijeka, Croatia, 2010.
- [22] M. Ali, M. Pant, and A. Abraham, “Simplex differential evolution,” *Acta Polytechnica Hungarica*, vol. 6, no. 5, pp. 95–115, 2009.
- [23] S. Almeida-Luz, M. A. Vega-Rodríguez, J. A. Gómez-Pulido, and J. M. Sánchez-Perez, “Applying differential evolution to the reporting cells problem,” in *Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT '08)*, pp. 65–71, IEEE, Wisła, Poland, October 2008.
- [24] G. Jeyakumar and C. S. Velayutham, “An empirical performance analysis of differential evolution variants on unconstrained global optimization problems,” *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 2, pp. 77–86, 2010.
- [25] I. Triguero, S. García, and F. Herrera, “Differential evolution for optimizing the positioning of prototypes in nearest neighbor classification,” *Pattern Recognition*, vol. 44, no. 4, pp. 901–916, 2011.
- [26] M. Weber, *Parallel global optimization: structuring populations in differential evolution [Ph.D. thesis]*, University of Jyväskylä, Jyväskylä, Finland, 2010.
- [27] S. M. Elsayed, R. A. Sarker, and D. L. Essam, “Differential evolution with multiple strategies for solving CEC2011 real-world numerical optimization problems,” in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '11)*, pp. 1041–1048, June 2011.
- [28] B. Xin, J. Chen, J. Zhang, H. Fang, and Z.-H. Peng, “Hybridizing differential evolution and particle swarm optimization to design powerful optimizers: a review and taxonomy,” *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 42, no. 5, pp. 744–767, 2012.
- [29] B. Y. Qu and P. N. Suganthan, “Novel multimodal problems and differential evolution with ensemble of restricted tournament selection,” in *Proceedings of the IEEE World Congress on Computational Intelligence*, pp. 1–7, Barcelona, Spain, 2010.
- [30] M. G. Epitropakis, V. P. Plagianakos, and M. N. Vrahatis, “Finding multiple global optima exploiting differential evolution’s niching capability,” in *Proceedings of the IEEE Symposium on Differential Evolution (SDE '11)*, pp. 1–8, IEEE, Paris, France, April 2011.
- [31] C. Qiu, M. Liu, and W. Gong, “Differential evolution with tournament-based mutation operators,” *International Journal of Computer Science Issues*, vol. 10, no. 2, pp. 180–187, 2013.
- [32] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, “Differential evolution using a neighborhood-based mutation operator,” *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 3, pp. 526–553, 2009.
- [33] A. M. Sutton, M. Lunacek, and L. D. Whitley, “Differential evolution and non-separability: using selective pressure to focus search,” in *Proceedings of the 9th Annual Genetic and Evolutionary Computation Conference*, pp. 1428–1435, London, UK, July 2007.
- [34] M. G. Epitropakis, D. K. Tasoulis, N. G. Pavlidis, V. P. Plagianakos, and M. N. Vrahatis, “Enhancing differential evolution utilizing proximity-based mutation operators,” *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 99–119, 2011.
- [35] D. Zaharie, “A comparative analysis of crossover variants in differential evolution,” in *Proceedings of the International Multiconference on Computer Science and Information Technology*, pp. 171–181, 2008.
- [36] B. V. Babu and S. A. Munawar, “Differential evolution strategies for optimal design of shell-and-tube heat exchangers,” *Chemical Engineering Science*, vol. 62, no. 14, pp. 3720–3739, 2007.
- [37] E. Mezura-Montes, J. Velázquez-Reyes, and C. A. Coello Coello, “A comparative study of differential evolution variants for global optimization,” in *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation (GECCO '06)*, pp. 485–492, Washington, DC, USA, 2006.
- [38] M. Ali, M. Pant, A. Abraham, and V. Snasel, “Differential evolution using mixed strategies in competitive environment,” *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 8, pp. 5063–5084, 2011.
- [39] Y. Zhang, S. Wang, P. Phillips, and G. Ji, “Binary PSO with mutation operator for feature selection using decision tree applied to spam detection,” *Knowledge-Based Systems*, vol. 64, pp. 22–31, 2014.
- [40] S. M. Islam, S. Das, S. Ghosh, S. Roy, and P. N. Suganthan, “An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 2, pp. 482–500, 2012.
- [41] Y. Wang, Z. Cai, and Q. Zhang, “Differential evolution with composite trial vector generation strategies and control parameters,” *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 55–66, 2011.
- [42] W. Gong, Z. Cai, C. X. Ling, and H. Li, “Enhanced differential evolution with adaptive strategies for numerical optimization,” *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 2, pp. 397–413, 2011.
- [43] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, “Differential evolution using a neighborhood-based mutation

- operator," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 3, pp. 526–553, 2009.
- [44] 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.
- [45] C.-K. Ting and C.-H. Huang, "Varying number of difference vectors in differential evolution," in *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 1351–1358, Chiayi, Taiwan, May 2009.
- [46] B. L. Miller and D. E. Goldberg, "Genetic algorithms, tournament selection, and the effects of noise," *Complex Systems*, vol. 9, no. 3, pp. 193–212, 1995.
- [47] T. Blicke and L. Thiele, "A comparison of selection schemes used in genetic algorithms," TIK-Report 11, TIK Institut für Technische und Kommunikationsnetze, Swiss Federal Institute of Technology, Zurich, Switzerland, 1995.
- [48] H. Takagi and D. Pallez, "Paired comparisons-based interactive differential evolution," in *Proceedings of the World Congress on Nature & Biologically Inspired Computing*, pp. 475–480, Coimbatore, India, 2009.
- [49] P. Pošik and J. Kubalík, "Experimental comparison of six population-based algorithms for continuous black box optimization," *Evolutionary Computation*, vol. 20, no. 4, pp. 483–508, 2012.
- [50] M. G. Epitropakis, V. P. Plagianakos, and M. N. Vrahatis, "Balancing the exploration and exploitation capabilities of the differential evolution algorithm," in *Proceedings of the IEEE World Congress on Computational Intelligence, IEEE Congress on Evolutionary Computation (CEC '08)*, pp. 2686–2693, Hong Kong, June 2008.
- [51] R. Mendes and A. S. Mohais, "DynDE: a differential evolution for dynamic optimization problems," in *Proceedings of the IEEE Congress on Evolutionary Computation (IEEE CEC '05)*, pp. 2808–2815, September 2005.
- [52] N. Noman and H. Iba, "Accelerating differential evolution using an adaptive local search," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 107–125, 2008.
- [53] M. M. Ali and A. Torn, "Population set-based global optimization algorithms: some modifications and numerical studies," *Computers & Operations Research*, vol. 31, no. 10, pp. 1703–1725, 2004.
- [54] A. P. Piotrowski and J. J. Napiorkowski, "The grouping differential evolution algorithm for multi-dimensional optimization problems," *Control and Cybernetics*, vol. 39, no. 2, pp. 527–550, 2010.
- [55] X.-L. Dong, S.-Q. Liu, T. Tao, S.-P. Li, and K.-L. Xin, "A comparative study of differential evolution and genetic algorithms for optimizing the design of water distribution systems," *Journal of Zhejiang University: Science A*, vol. 13, no. 9, pp. 674–686, 2012.
- [56] Y. Wang, Z. Cai, and Q. Zhang, "Enhancing the search ability of differential evolution through orthogonal crossover," *Information Sciences*, vol. 185, pp. 153–177, 2012.
- [57] Y. Zhang, S. Wang, Y. Sun, G. Ji, P. Phillips, and Z. Dong, "Binary structuring elements decomposition based on an improved recursive dilation-union model and RSAPSO method," *Mathematical Problems in Engineering*, vol. 2014, Article ID 272496, 12 pages, 2014.
- [58] Y. Yoon and H. Kim, "The roles of crossover and mutation in real-coded genetic algorithms," in *Bio-Inspired Computational Algorithms and Their Applications*, S. Gao, Ed., InTech, 2012.
- [59] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

Research Article

Fuzzy Wavelet Neural Network Using a Correntropy Criterion for Nonlinear System Identification

Leandro L. S. Linhares,^{1,2} Aluisio I. R. Fontes,^{1,2} Allan M. Martins,³
Fábio M. U. Araújo,² and Luiz F. Q. Silveira²

¹Postgraduate Program in Electrical and Computer Engineering (PPgEEC), Federal University of Rio Grande do Norte, 59078-970 Natal, RN, Brazil

²Department of Computer Engineering, Federal University of Rio Grande do Norte, 59078-970 Natal, RN, Brazil

³Department of Electrical Engineering, Federal University of Rio Grande do Norte, 59078-970 Natal, RN, Brazil

Correspondence should be addressed to Aluisio I. R. Fontes; aluisio.igor@gmail.com

Received 17 September 2014; Accepted 28 November 2014

Academic Editor: Yudong Zhang

Copyright © 2015 Leandro L. S. Linhares 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 have demonstrated that the Fuzzy Wavelet Neural Networks (FWNNs) are an efficient tool to identify nonlinear systems. In these structures, features related to fuzzy logic, wavelet functions, and neural networks are combined in an architecture similar to the Adaptive Neurofuzzy Inference Systems (ANFIS). In practical applications, the experimental data set used in the identification task often contains unknown noise and outliers, which decrease the FWNN model reliability. In order to reduce the negative effects of these erroneous measurements, this work proposes the direct use of a similarity measure based on information theory in the FWNN learning procedure. The Mean Squared Error (MSE) cost function is replaced by the Maximum Correntropy Criterion (MCC) in the traditional error backpropagation (BP) algorithm. The input-output maps of a real nonlinear system studied in this work are identified from an experimental data set corrupted by different outliers rates and additive white Gaussian noise. The results demonstrate the advantages of the proposed cost function using the MCC as compared to the MSE. This work also investigates the influence of the kernel size on the performance of the MCC in the BP algorithm, since it is the only free parameter of correntropy.

1. Introduction

System identification is a modeling procedure where the mathematical representation of the input-output maps for dynamical systems can be obtained with the aid of experimental data. This procedure is a prominent alternative for the efficient modeling of complex systems without the need for using complex mathematical concepts. For this reason, this system identification plays an important role in some control engineering related tasks such as classification and decision making, monitoring, control, and prediction [1–8].

Artificial Neural Networks (ANNs) represent one of the most successful identification techniques used to model nonlinear dynamical systems [9]. This is due to their ability to learn by examples associated with intrinsic robustness

and nonlinear characteristics [10–13]. Recently, a wide variety of network structures have been used to model the input-output maps of nonlinear systems [5, 14, 15]. Multilayer Perceptron (MLP), Radial Basis Function (RBF) network, Neurofuzzy Hybrid Structures, for example, Adaptive Neurofuzzy Inference Systems (ANFIS), and Wavelet Neural Networks (WNN) are examples of ANNs commonly used in applications involving nonlinear systems [9, 13, 16, 17].

WNNs combine the flexibility of ANNs and the curve fitting ability of wavelet functions [18–20]. Besides, it can be improved in terms of extending the domain of validity by the addition of an extra layer of fuzzy structures to achieve the course delimitation of the universe of discourse, resulting in Fuzzy Wavelet Neural Networks (FWNNs) [5]. The architecture of the FWNN is very close to the traditional ANFIS [21],

although wavelets are used as membership functions (MFs) [22, 23], or in the consequent part of fuzzy rules, through the use of WNNs as local models. In literature, it is often possible to find several research works applying FWNN to deal with modeling, control, function approximation, and nonlinear system identification, among others [6, 24–28].

In [29], Linhares et al. evaluate an alternative FWNN structure to identify the nonlinear dynamics of a multisection liquid tank. The aforementioned proposed structure is similar to the ones presented by Yilmaz and Oysal [5], Abiyev and Kaynak [6], and Lu [24]. However, the FWNN presented in [29] uses only wavelets in the consequent fuzzy rules. The wavelets in each node of the FWNN consequent layer are weighted by the activation signals of the fuzzy rules. Therefore, the local models of such FWNN are solely represented by a set of wavelet functions, which differs from [5, 6, 24]. The results presented in [29] demonstrate that the modified FWNN structure maintains the generalization capability and also other important features presented by traditional FWNNs, despite the reduction in the complexity of these structures.

In practical applications, the experimental data set used in the identification procedure is often corrupted by unknown noise and outliers. The outliers are incorrect measurements which markedly deviate from the typical ranges of other observations [30]. The main source of the outliers comes from sporadic malfunctioning of sensors and equipments [31]. The presence of noise and outliers in experimental data negatively affects the performance and reliability of the dynamical model under identification, because it tries to fit such undesired measurements [30, 32, 33]. Despite the fact that there are many outlier detection methods presented in literature, many approaches are not able to detect all the outliers. Therefore, the resulting data obtained after the application of such methods may still be contaminated with outliers [30, 31].

Generally, the learning process of the neural networks is based on a given gradient method, for example, the classical error backpropagation (BP) algorithm which uses the Mean Squared Error (MSE) as its cost function. However, the applicability of MSE to obtain a model that represents an input-output relationship is optimal only if the probability distribution function (pdf) of the errors is Gaussian [34]. However, the error distribution in most cases is nongaussian and nonlinear [8]. In literature we can find some researches that demonstrate that the use of the Maximum Correntropy Criterion (MCC) replacing the traditional MSE is an effective approach to handle the problem of prediction and identification when the dynamical system has unknown noise and outliers [7, 8, 30, 35]. The correntropy evaluation allows the extraction of additional information from available data because such similarity measure takes into account all the moments of a probability distribution that are typically not observed by MSE [7].

In this work, the reliability of the FWNN recently proposed in [29] is evaluated when different percentages of outliers and noise contaminate the experimental data used to identify a nonlinear system. The aforementioned neural network is used to identify the dynamic relationship between

the input and output of a multisection liquid tank. In order to train the FWNN, the BP algorithm is used, although the traditional MSE cost function is replaced by the Maximum Correntropy Criterion using an adaptive adjustment of its kernel size, which is the free parameter of the MCC. The obtained models using each one of the quality measures are properly evaluated and compared. Despite the advantages of correntropy over MSE, little effort has been reported towards the application of correntropy to identify nonlinear systems using neural networks [7, 8]. The results presented in this work demonstrate that the FWNN architecture proposed in [29] is less sensitive to the presence of outliers and noise when it is trained using the MCC. In addition, this work also investigates the influence of the kernel size on the performance of the MCC in BP algorithm.

This paper is organized as follows. Section 2 presents the definition and the basic mathematical theory of the similarity measure of correntropy. Then, Section 3 describes the FWNN proposed in [29], which is applied in this work to identify an experimental nonlinear dynamical system considering the presence of outliers and noise. Section 4 presents the updating equations of BP algorithm, which are modified according to the MCC. Section 5 describes the proposed identification architecture in detail. Section 6 presents the multisection liquid tank under study, while the performance of FWNN models obtained using MSE and MCC cost functions is evaluated, considering the presence of both outliers and noise in experimental data. Finally, concluding remarks are given in Section 7.

2. Correntropy

Correntropy is a generalized similarity measure between two arbitrary scalar random variables X and Y defined by [36]

$$V_{\sigma}(X, Y) = E_{XY} [k_{\sigma}(X, Y)] = \int \int k(x, y) p_{X,Y}(x, y) dx dy, \quad (1)$$

where $p_{X,Y}$ is the joint probability distribution, $E[\cdot]$ is the expectation operator, and $k_{\sigma}(\cdot, \cdot)$ is a symmetric positive definite kernel. In this work, $k_{\sigma}(\cdot, \cdot)$ is a Gaussian kernel given as

$$k_{\sigma}(x_j, x_i) = K_{\sigma}(x_j - x_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left[-\frac{(x_j - x_i)^2}{2\sigma^2} \right], \quad (2)$$

where σ is the variance defined as the kernel size. The kernel size may be interpreted as the resolution for which correntropy measures similarity in a space with characteristics of high dimensionality [36].

By applying a Taylor series expansion to the Gaussian function in (1) and assuming that all the moments of the joint pdf are finite, such equation becomes

$$V_{\sigma}(X, Y) = \frac{1}{\sqrt{2\pi}\sigma} \sum_{k=0}^{\infty} \frac{(-1)^k}{2^k \sigma^{2k} k!} E[(X - Y)^{2k}]. \quad (3)$$

In practice, the joint pdf in (1) is unknown and only a finite amount of data $\{(x_i, y_i)\}_{i=1}^N$ is available, leading to the sample correntropy estimator defined by

$$\hat{v}_\sigma(X, Y) = \frac{1}{n} \sum_{i=1}^N k_\sigma(x_i, y_i). \quad (4)$$

Correntropy involves all the even moments of difference between X and Y . Compared with MSE $E[(X - Y)^2]$ which is a quadratic function in the joint input space, correntropy includes second-order and higher-order statistical information [37]. However, for sufficiently large values of σ , the second-order moment is predominant and the measure approaches correlation [38].

Nowadays, correntropy has been successfully used in a wide variety of applications where the signals are non-Gaussian or nonlinear, for example, automatic modulation classification [39], classification systems of pathological voices [40], and principal component analysis (PCA) [41].

2.1. Maximum Correntropy Criterion for Model Estimation. The correntropy concept can be extended to the model estimation. The variable X can be considered as a mathematical expression of the unknown function $f(X, w)$, where X is an input set $X = \{x_i \in R^m\}_{i=1, \dots, N}$ and the model parameters are $w = [w_1, \dots, w_m]^T$, which approximates the dependence on an output set $Y = \{y_i \in R^m\}_{i=1, \dots, N}$ [42].

Therefore, it is possible to determine the optimal solution for the MCC from (4) as [43]

$$\begin{aligned} \arg \max_w J(w) &= \frac{1}{M} \sum_{i=1}^M k_\sigma(f(X, w), Y) \\ &= \frac{1}{M} \sum_{i=1}^N K_\sigma(e_i), \end{aligned} \quad (5)$$

where $e_i = f(X, w) - y_i$ and $i = 1, \dots, N$, which are the errors generated by the model during the supervised learning for each of the N training samples. It is worth mentioning that such criterion is used as the cost function of the BP algorithm to adjust the parameters of the FWNN.

One of the advantages of using correntropy in system identification lies in the robustness of such measure against impulsive noise due to the use of the Gaussian kernel in (5), which is close to zero; that is, $k_\sigma(f(X, w), Y) \approx 0$ when $f(X, w)$ or Y is an outlier. Correntropy is positive and bounded, and it gives $0 < \hat{v}_\sigma(X, Y) \leq 1/\sqrt{2\pi}\sigma$ for the Gaussian kernel.

The Gaussian variance (also called kernel size) is a free parameter that must be selected by the user [38]. Therefore, when the correntropy is estimated, the resulting values depend on the selected kernel size. In addition, the kernel size of correntropy influences the nature of the performance surface, presence of local optima, rate of convergence, and robustness to impulsive noise during adaption [37, 43]. If the training data size is not large enough, the kernel size must be chosen considering tradeoffs between outlier rejection and estimation efficiency [44].

Some approaches can be employed to determine the kernel size, for example, the statistical method [45], Silverman's rule [46], cross validation techniques [47, 48], and shape of the prediction error distribution [44]. This work uses an adaptive kernel size algorithm [42], which is given by

$$\sigma = \frac{\max |e_i|}{2\sqrt{2}}, \quad i = 1, \dots, N. \quad (6)$$

In order to assess the improved performance of an adaptive kernel size over fixed ones, Section 6 is supposed to show how the error evolves during the FWNN training for different values of the kernel size.

3. Fuzzy Wavelet Neural Networks

3.1. Brief Review. Wavelets are obtained by scaling and translating a special function $\psi(x)$ localized in both time/space and frequency called mother wavelet, which can be defined in such a way to serve as a basis to describe other functions. Wavelets are extensively used in the fields of signal analysis, identification and control of dynamical systems, computer vision, and computer graphics, among other applications [49–52]. Given $\psi(x)$, the corresponding family of wavelets is obtained by

$$\Psi_j(\mathbf{x}) = |\mathbf{d}_j|^{-1/2} \psi\left(\frac{\mathbf{x} - \mathbf{t}_j}{\mathbf{d}_j}\right), \quad \mathbf{d}_j \neq 0, \quad j = 1, 2, \dots, k, \quad (7)$$

where $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$ and $\Psi_j(\mathbf{x})$ is obtained from $\psi(\mathbf{x})$ by scaling it by a factor $\mathbf{d}_j = \{d_{1j}, d_{2j}, \dots, d_{nj}\}$ and translating it by $\mathbf{t}_j = \{t_{1j}, t_{2j}, \dots, t_{nj}\}$.

A WNN is a nonlinear regression structure that can represent input-output maps by combining wavelets with appropriate scalings and translations [53]. The output of a WNN is determined as follows:

$$y = \sum_{j=1}^k \mathbf{w}_j \Psi_j(\mathbf{x}) = \sum_{j=1}^k \mathbf{w}_j |\mathbf{d}_j|^{-1/2} \psi\left(\frac{\mathbf{x} - \mathbf{t}_j}{\mathbf{d}_j}\right), \quad (8)$$

where \mathbf{w}_j are the synaptic weights, \mathbf{x} is the input vector, and \mathbf{d}_j and \mathbf{t}_j are parameters characterizing the wavelets.

In a concise manner, the purpose of FWNNs is to incorporate WNNs into the ANFIS structure in order to obtain faster convergence and better approximation capabilities, eventually with a greater number of parameters to be adjusted. The fuzzy rules allow tackling the uncertainties, while wavelets contribute to improving the accuracy in the process of approximating input-output maps [6].

3.2. FWNN Architecture. A particular instance of FWNN proposed in [29] is applied in this work to identify a real nonlinear system, investigating its performance and reliability when the experimental data set is corrupted by unknown noise and outliers. In this FWNN architecture, the consequent part of its fuzzy rules is described only by wavelet functions. It differs from other structures such as those proposed in [5, 6, 24]. The basic architecture of the FWNN can be seen in Figure 1 and its layers are described as follows.

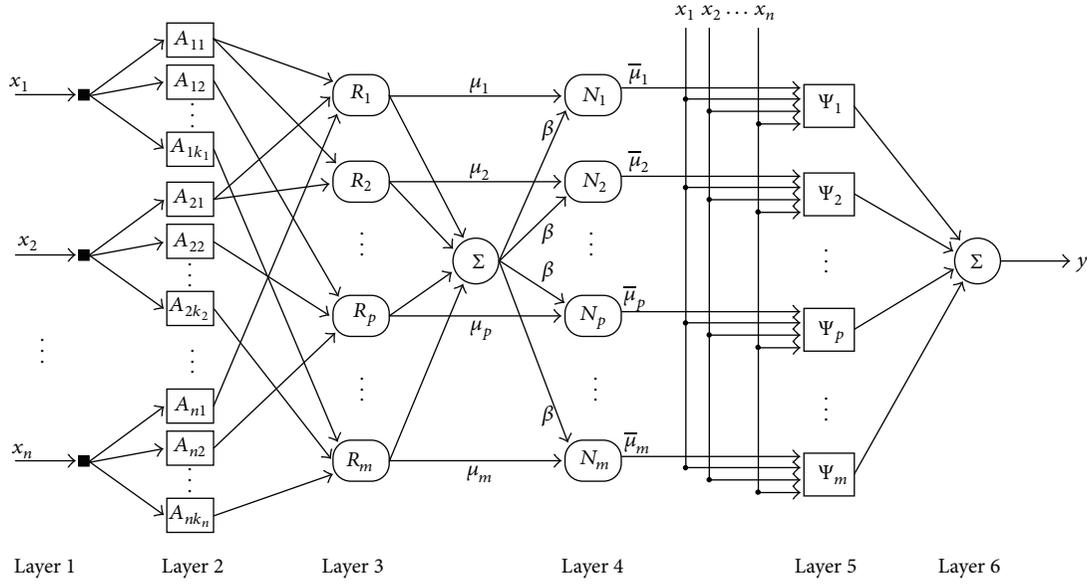


FIGURE 1: Graphical representation of the FWNN architecture.

Layer 1. The input layer just transfers the input signal vector $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$ to the next layer.

Layer 2. In the fuzzification layer, the membership functions are parameterized to match the specific requirements of a variety of applications. For instance, a Gaussian membership function can be described by the following equation:

$$A_{qr}(x_q) = \exp \left[-\frac{1}{2} \left(\frac{x_q - a_{qr}}{b_{qr}} \right)^2 \right], \quad (9)$$

where for $q = 1, 2, \dots, n$ and $r = 1, 2, \dots, k_q$, A_{qr} would be associated with the r th membership function appearing in a given rule and evaluated for the q th component of the input vector. The adjustable parameters are a_{qr} and b_{qr} , representing the center and width of the membership function, respectively.

Layer 3. This is the inference layer. Assuming that there are m rules, where R_i is a given rule and $i = 1, 2, \dots, m$, each rule is supposed to produce and output μ_i by aggregating A_{qr} using a T-norm. The output of the p th rule in this layer is

$$\mu_p = \prod_{q=1}^n A_{qr}(x_q), \quad p = r_1, r_2, \dots, r_n, \quad (10)$$

where $r_1 = 1, \dots, k_1, r_2 = 1, \dots, k_2, r_n = 1, \dots, k_n$.

All the rule outputs of this layer are added up to the summation node located between Layers 3 and 4. The output β of this node is later used in the normalization stage.

Layer 4. In the normalization layer, the normalization factor for the output of the i th rule, denoted by $\bar{\mu}_i$ is given by

$$\bar{\mu}_i = \frac{\mu_i}{\beta} = \frac{\mu_i}{\sum_{i=1}^m \mu_i}, \quad i = 1, 2, \dots, m. \quad (11)$$

Layer 5. This is the consequent layer of the FWNN. In this work, the Mexican Hat family of wavelets is adopted as in [5, 6, 54]. Its mathematical representation is given by

$$\psi(x) = \frac{1}{\sqrt{d}} \left(1 - \left(\frac{x-t}{d} \right)^2 \right) \exp \left[-0.5 \left(\frac{x-t}{d} \right)^2 \right]. \quad (12)$$

The inputs of the wavelet layer are the normalized weights $\bar{\mu}_i$ and the input vector $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$, while the outputs of this layer represented by f_j are given by

$$f_j = \bar{\mu}_j \Psi_j, \quad j = 1, 2, \dots, m,$$

$$f_j = \bar{\mu}_j \sum_{i=1}^n (1 - T_{ij}^2) \exp \left[-0.5 T_{ij}^2 \frac{1}{\sqrt{d_{ij}}} \right], \quad (13)$$

where the term $T_{ij} = (x_i - t_{ij})/d_{ij}$, $d_{ij} > 0$ is adopted to simplify the mathematical notation and n is the number of wavelet functions in a node of Layer 5.

Layer 6. In the output layer, all signals from the wavelet neurons are summed up as follows:

$$y = \sum_{j=1}^m f_j. \quad (14)$$

By observing Figure 1 and considering (9) to (14), it is possible to notice that the FWNN related parameters are located in the second and fifth layers. The membership functions and wavelet functions are adjusted according to the application using any learning algorithm, such as BP algorithm.

4. Error Backpropagation Algorithm with MCC

The classical BP algorithm is the learning algorithm used in this work to adjust the free parameters of the FWNN models. According to [54], this algorithm is probably the most frequently used technique to train a FWNN. Despite its functionality, it presents some shortcomings such as the fact that it may get stuck on a local minimum of the error surface and that the training convergence rate is generally slow [55–57]. However, it is well known that the use of wavelet functions in neural network structures reduces such inconveniences [6, 58].

A neural system should be designed to present a desired behavior; hence, it is necessary to define a cost function for this task. It provides an evaluation of the quality of the solution obtained by the neural model [59]. The gradient based learning algorithms, such as the BP algorithm, require the differentiation of the chosen cost function with respect to the adjustable parameters of the FWNN model. Therefore, it is necessary to obtain the partial derivatives of the chosen cost function with respect to parameters d_{ij} and t_{ij} of the wavelets and parameters a_{qr} and b_{qr} of the membership functions A_{qr} .

Typically, MSE is the cost function used with BP algorithm [10]. Such classical cost criterion is replaced by MCC in this work in order to increase the reliability of the FWNN model when the identified dynamical system presents outliers and noise. When using MCC, the main goal is to maximize the correntropy similarity measure between two random process variables. In the FWNN learning procedure, such variables are the desired output y_d and the estimated output \hat{y} provided by the FWNN model. Considering the estimation error of the FWNN model given by $e = y_d - \hat{y}$, maximizing the MCC is equivalent to minimizing

$$\mathcal{E}_{av} = \frac{1}{N} \sum_{n=1}^N \left[\frac{1}{\sqrt{2\pi}\sigma} - K_\sigma(e_n) \right], \quad (15)$$

where N is the number of samples in the experimental data. Equation (15) corresponds to the cost function used during the minimization process of the BP algorithm applied to adjust the parameters of the FWNN models. As such parameters are adjusted sequentially, (16) defines the instantaneous correntropy used to update the wavelet functions and membership functions parameters of the FWNN after each training pair is presented to this network. Consider

$$\mathcal{E} = \frac{1}{\sqrt{2\pi}\sigma} - K_\sigma(e_n). \quad (16)$$

By differentiating \mathcal{E} with respect to d_{ij} and t_{ij} , it gives

$$\begin{aligned} \frac{\partial \mathcal{E}}{\partial d_{ij}} &= -\frac{e_n K_\sigma(e_n) \bar{\mu}_i (\exp[-0.5T_{ij}^2] (3T_{ij}^2 - T_{ij}^4))}{\sigma^2 \sqrt{d_{ij}^3}}, \\ \frac{\partial \mathcal{E}}{\partial t_{ij}} &= -\frac{e_n K_\sigma(e_n) \bar{\mu}_i (\exp[-0.5T_{ij}] (3T_{ij} - T_{ij}^3))}{\sigma^2 \sqrt{d_{ij}^3}}. \end{aligned} \quad (17)$$

Now, differentiating \mathcal{E} with respect to a_{qr} and b_{qr} , it gives

$$\begin{aligned} \frac{\partial \mathcal{E}}{\partial a_{qr}} &= \frac{\partial \mathcal{E}}{\partial y} \frac{\partial y}{\partial \Psi} \frac{\partial \Psi}{\partial \bar{\mu}} \frac{\partial \bar{\mu}}{\partial \mu} \frac{\partial \mu}{\partial A} \frac{\partial A}{\partial a_{qr}}, \\ \frac{\partial \mathcal{E}}{\partial b_{qr}} &= \frac{\partial \mathcal{E}}{\partial y} \frac{\partial y}{\partial \Psi} \frac{\partial \Psi}{\partial \bar{\mu}} \frac{\partial \bar{\mu}}{\partial \mu} \frac{\partial \mu}{\partial A} \frac{\partial A}{\partial b_{qr}}, \end{aligned} \quad (18)$$

where

$$\begin{aligned} \frac{\partial A_{qr}}{\partial b_{qr}} &= \begin{cases} A_{qr} \frac{(x_q - b_{qr})}{a_{qr}^2}, & \text{if } x_q \neq b_{qr}, \\ 0, & \text{if } x_q = b_{qr}, \end{cases} \\ \frac{\partial A_{qr}}{\partial a_{qr}} &= \begin{cases} A_{qr} \frac{(x_q - b_{qr})^2}{a_{qr}^3}, & \text{if } x_q \neq b_{qr}, \\ 0, & \text{if } x_q = b_{qr}. \end{cases} \end{aligned} \quad (19)$$

Following the delta rule mentioned in [10], the parameters of the proposed FWNN are updated as follows:

$$\begin{aligned} d_{ij}(k+1) &= d_{ij}(k) - \eta \frac{\partial \mathcal{E}}{\partial d_{ij}}, \\ t_{ij}(k+1) &= t_{ij}(k) - \eta \frac{\partial \mathcal{E}}{\partial t_{ij}}, \\ a_{qr}(k+1) &= a_{qr}(k) - \eta \frac{\partial \mathcal{E}}{\partial a_{qr}}, \\ b_{qr}(k+1) &= b_{qr}(k) - \eta \frac{\partial \mathcal{E}}{\partial b_{qr}}, \end{aligned} \quad (20)$$

where η is the learning rate. For the training algorithm initialization, wavelets and membership functions parameters are set with random numbers from a uniform distribution.

The replacement of the traditional MSE by MCC inserts another learning parameter to BP algorithm. As already explained, the success of the correntropy is based on the appropriate adjustment of the kernel size of its Gaussian functions. This new parameter influences the nature of the performance surface, presence of local optima, rate of convergence, and robustness. Therefore, if an unsuitable kernel size is chosen, the expected improved performance of the MCC will not be confirmed [60]. For this reason, an adaptive kernel size method is applied in this work (see (6)) to adjust the kernel size over the learning epochs.

5. Proposed Identification Architecture

The proposed architecture adopted in this work identifies the dynamic relationship between the input and output of a multisection tank for water storage. The system is evaluated when the experimental data used during the identification task is corrupted with noise and outliers. The proposed architecture is based on the series-parallel identification scheme described in [13], with small modifications due to the experimental data set characteristic and the learning

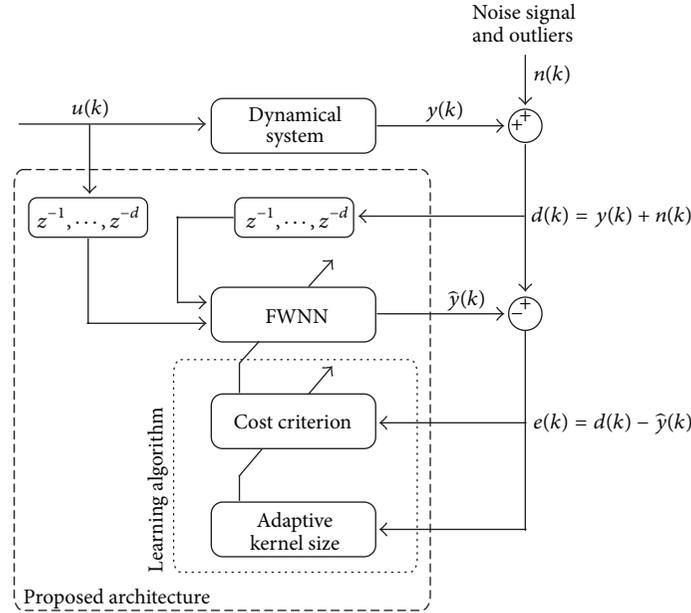


FIGURE 2: Proposed identification architecture.

procedure used to adjust the parameters of the FWNN model. Figure 2 presents a schematic diagram of the proposed identification architecture in this work.

The inputs of the FWNN model are past values of input signal $u(k)$ and the system output when corrupted with noise and outliers $d(k) = y(k) + n(k)$, while the estimated output is given by $\hat{y}(k)$. The work developed in [9] shows that well-known linear modeling structures, such as FIR (Finite Impulse Response), ARX (AutoRegressive, eXogenous input), ARMAX (AutoRegressive, Moving Average, eXogenous input), OE (Output Error), and SSIF (State Space Innovations Form) may be extended by using nonlinear functions or representations, thus leading to the nonlinear modeling structures NFIR, NARX, NARMAX, NOE, and NSSIF. This concept is used to define the inputs of the FWNN models obtained in this study.

According to [9], the advantage of a NARX model is that none of its regressors depends on past outputs of the model, which ensures that the predictor remains stable. This is particularly important in the nonlinear case since the stability issue in this particular case is much more complex than in linear systems. Considering that the inputs of the FWNN models in this work are described exactly as the regression vector of the NARX modeling structure, they inherit important characteristics from such structure. Figure 3 shows more details on the FWNN inputs in accordance with the NARX structure, where λ , τ , and α are constants that define a model of order α and delay τ .

Figure 2 illustrates that the FWNN model parameters are updated according to the error signal $e(k) = d(k) - \hat{y}(k)$, by using a learning algorithm, for example, the BP algorithm. By adopting the MCC as its respective cost criterion, the learning algorithm is applied to the FWNN model. As it was previously explained in Section 2, the success of the MCC also

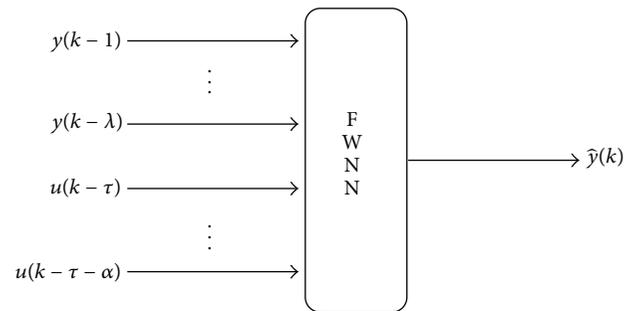


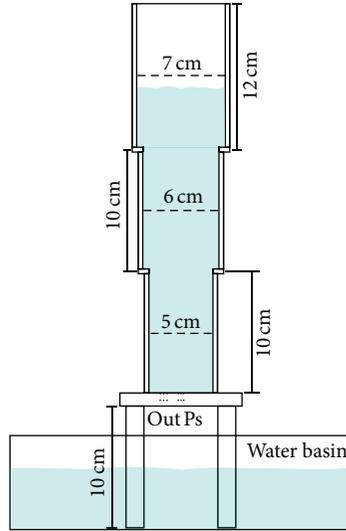
FIGURE 3: NARX model structure using FWNN.

depends on the correct choice of the kernel size. Therefore, the adaptive method described by (6) is used in this work to adjust the kernel size during the learning epochs.

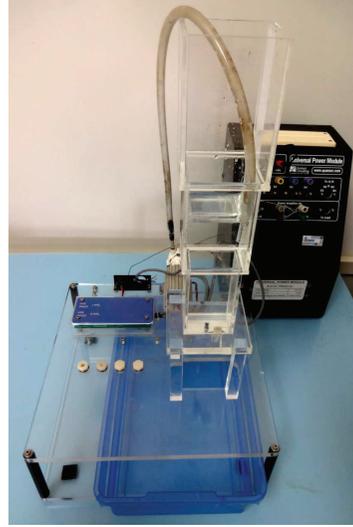
6. Experiments and Results

In order to evaluate the performance of the FWNN when the traditional MSE cost criterion of the error backpropagation algorithm is replaced by MCC, the aforementioned neural network is used to identify a real dynamical system, considering that its experimental data is corrupted by noise and outliers.

6.1. Multisection Liquid Tank. The multisection liquid tank consists of an acrylic tank for containment of liquids with three abrupt changes in its cross-sectional area, as it can be seen in Figure 4. The liquid tank was originally designed for educational purposes in order to be used in studies of identification and control of dynamical systems [61]. It was also used in [29] to evaluate the performance of



(a) Schematic diagram of the multisection liquid tank



(b) Multisection liquid tank

FIGURE 4: Three-section liquid tank with distinct cross-sectional areas.

the alternative FWNN structure employed in this work. In addition to the acrylic tank structure, the system is composed by a water reservoir, a water pump, a pressure sensor, an electronic power driver, and an electronic interface with A/D (analog-to-digital) and D/A (digital-to-analog) converters.

The nonlinearity presented in the liquid flow output, which is due to the different pressure levels at the tank base in accordance with the height of the liquid column, can be clearly noticed in the aforementioned dynamical system. Besides, the distinct cross-sectional areas make such nonlinearity even more evident. It is worth mentioning that the abrupt transitions between the tank sections are also responsible for discontinuities. The whole system can be seen as a set of three coupled nonlinear systems, since each tank section has its own dynamic behavior.

6.2. System Identification. Initially, in order to collect the experimental data set used during the learning and testing phase of the identified FWNN models in this work, the water pump is excited with an APRBS (Amplitude Modulated Pseudorandom Binary Sequence) and the water level inside the tank is registered at a sample rate of $T_s = 2$ Hz. For the generation of the persistent excitation signal, the following parameters are considered: the minimum hold time $T_h = 10$ s, minimal amplitude $A_{\min} = 0$ V, and maximum amplitude $A_{\max} = 15$ V. Since only positive values of voltage are considered in this case study, the pump only operates in order to shift the liquid from the reservoir to the multisection tank.

After the system excitation, the collected data is corrupted with additive white Gaussian noise and two different percentages of outliers (1% and 3%). The resulting data are divided into two sets comprising approximately 80% and 20% of the total amount. The first set is used to train the FWNN model and the second one is used during the testing phase.

The whole data set is normalized to fit within the range $[0, 1]$ in order to avoid numerical problems during the FWNN learning procedure. Since the multisection tank is a first-order nonlinear system and also considering Figure 3, the inputs of the FWNN models are defined with $\lambda = \alpha = 1$ and $\tau = 0$. Thus, $u(k)$, $u(k-1)$, and $y(k)$ are defined as inputs to the FWNN models to predict $y(k+1)$.

The BP algorithm presented in Section 4 is used to adjust the parameters d_{ij} , t_{ij} , a_{qr} , and b_{qr} of the FWNN. After a trial-and-error procedure the learning rate $\eta = 0.0001$ was found as a good choice to identify the multisection tank. It is worth mentioning that the results presented in this work were obtained after 350 learning epochs.

Figure 5 presents the model validation when 1% of the original experimental data set is corrupted with outliers and additive white Gaussian noise is inserted. In this figure, the tank water level in cm is in function of the sample time step, where each time step is equivalent to 0.5 seconds, defined by the sample rate $T_s = 2$ Hz. The terms FWNN-MCC and FWNN-MSE are used to identify the FWNN models obtained using MCC and MSE as cost criterion of BP algorithm, respectively. It is evident that FWNN-MCC has the best performance due to the use of the higher-order statistical information. On the other hand, the FWNN-MSE model based on second-order moments presents some problems to efficiently identify the input-output dynamic relationship of the multisection tank at some points of the validation curve. The presence of outliers in the experimental data has a significant negative impact on the FWNN model when the MSE criterion is used in the learning procedure, once the error due to the outliers is increased by a square rate. The same behavior is not observed in FWNN-MCC when 1% of the experimental data are corrupted by outliers because the outliers power is weighted by the Gaussian kernel.

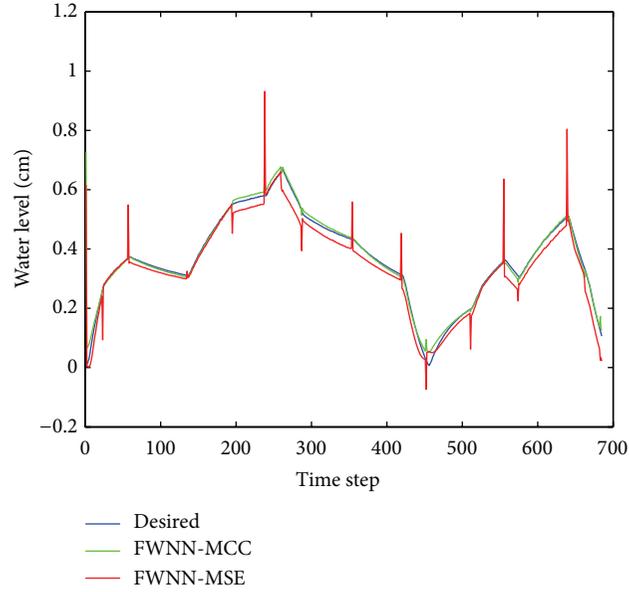


FIGURE 5: Validation of the FWNN-MCC with 1% outlier rate in experimental data.

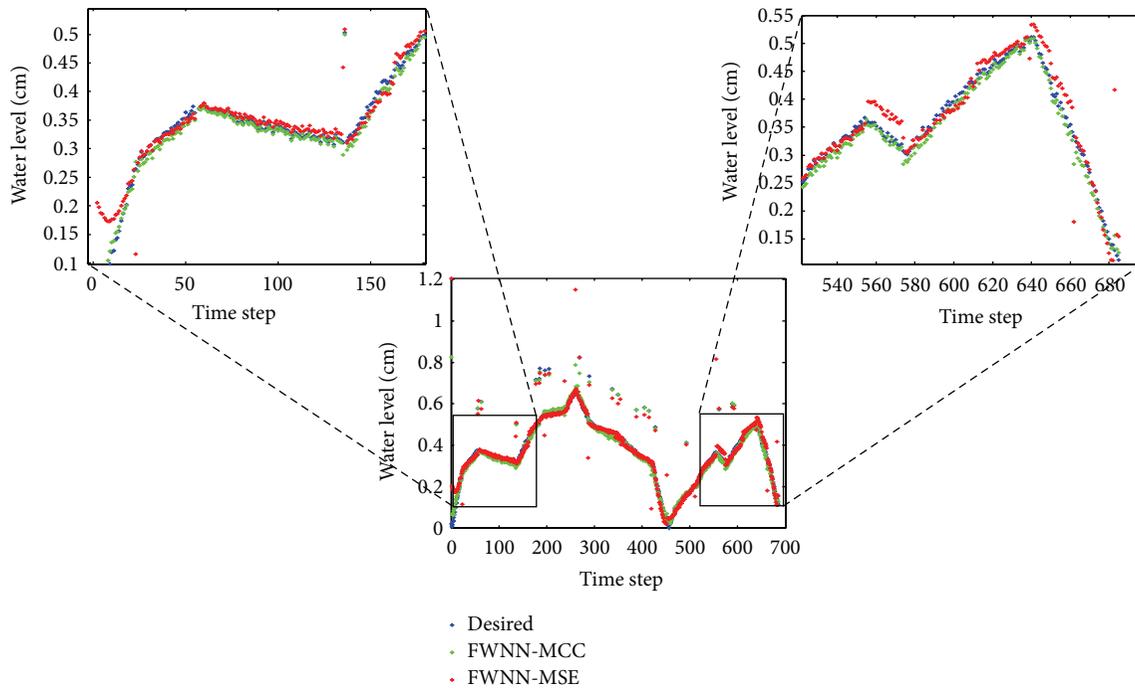


FIGURE 6: Validation of the FWNN-MCC with 3% outlier rate in experimental data.

Figure 6 shows the model validation when 3% of the original experimental data is corrupted with outliers and additive white Gaussian noise is inserted. In Figure 6, only the validation points are plotted to allow the better visualization of outliers and its respective effects in the FWNN-MCC and FWNN-MSE models. Two regions are highlighted in Figure 6, thus demonstrating the improvement of the FWNN-MCC model. Both models present problems at some points, although the performance of FWNN-MCC one is

improved in the identification of the multisection tank dynamics, as it also seems to be less sensitive to outliers and noise than FWNN-MSE model. It is noteworthy that MCC has intrinsic robustness due to the local estimation produced by the kernel size.

It is also important to mention that the correntropy criterion has a free parameter, that is, the kernel size, which is at the core of the learning process [38]. An adaptive kernel is applied in this work to improve the performance of

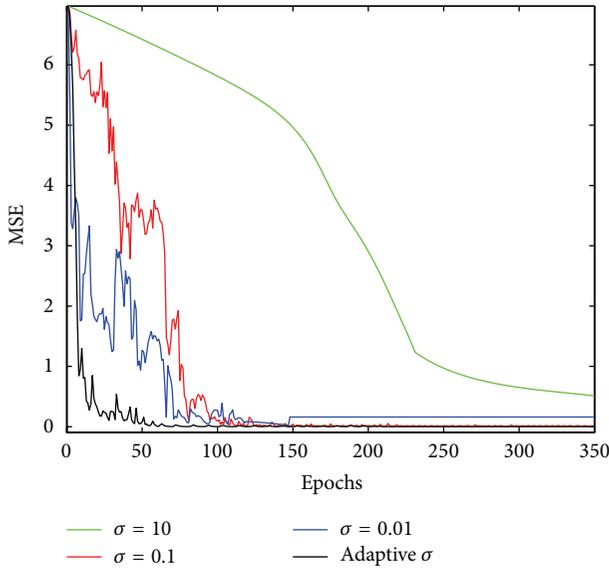


FIGURE 7: Evolution of MSE for different kernel sizes.

the FWNN learning procedure performance. Figure 7 shows MSE obtained over the 350 epochs, for three different fixed kernel sizes, that is, 0.01, 0.1, and 10, and also using the adaptive kernel. The adaptive kernel size method mathematically described by (6) has the highest convergence rate and the best performance in the attenuation of outliers and noise.

Figure 8 presents the behavior of the adaptive kernel size during the learning stage of the FWNN-MCC model when the experimental data is composed by 1% and 3% of outliers. During the initial epochs of the BP algorithm, the kernel size is quite oscillatory. However, the behavior of the kernel size becomes more stable as it comes to the hundredth epoch.

7. Conclusions

This work has analyzed the performance of a FWNN when applied to identify a real nonlinear dynamical system in the presence of unknown noise and outliers. Such erroneous measurements in experimental data reduce the reliability of the identified model, once it tries to fit some behaviors that are not part of the dynamical system. The most common learning techniques applied to adjust the FWNN parameters in identification applications are methods based on gradient that use the MSE as their cost function. This paper has then proposed the replacement of this traditional evaluation measure by a similarity measure based on information theory denominated correntropy. Therefore, the MCC was used in this paper as the cost function of the error backpropagation algorithm in order to reduce the negative effects of the unknown noise and outliers. The results have demonstrated that the FWNN-MCC models based on the MCC cost function represent the input-output dynamics of the multisection liquid tank more properly, being also less sensitive to outliers and noise than the FWNN-MSE models. This work also has investigated the influence of the kernel size on the performance of the MCC in the BP algorithm, since it is a free

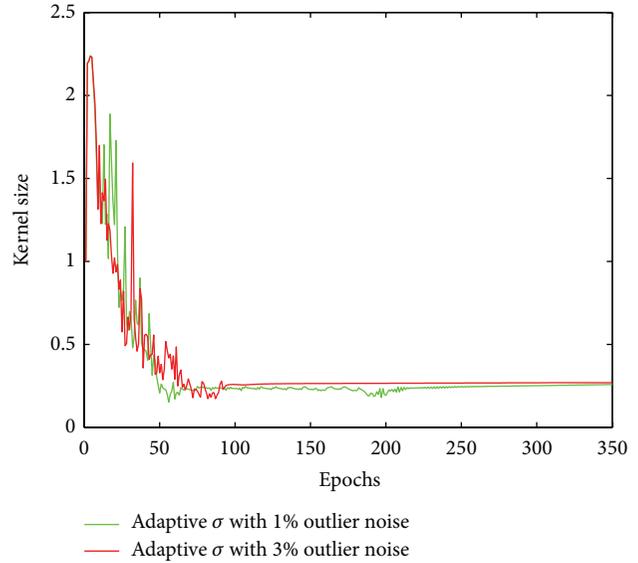


FIGURE 8: Adaptive kernel size behavior.

parameter of correntropy. The addition of this new parameter in the learning procedure of the FWNN can be considered a disadvantage of the proposed architecture, mainly because the MCC is very dependent on its proper adjustment. Within this context, the adopted adaptive kernel has shown to be more efficient if compared to the case when this parameter remains fixed during the whole FWNN learning process. The adaptive kernel size method has improved the convergence rate of the backpropagation algorithm and contributed to attenuating the effects of the outliers and noise. Due to the use of the BP algorithm, the proposed architecture is susceptible to local minima falls, limiting the correntropy action to remove the outliers.

The further research work will focus on the following items: (1) analyzing the application of the MCC associated with different algorithms in order to train the FWNN architecture to avoid the outliers harmful effects. The meta-heuristic algorithms such as Genetic Algorithm, Particle Swarm Optimization, and Bat Algorithm are good options since they are less sensitive to local minima than the BP algorithm; (2) including and comparing different adaptive kernel methods to improve the functionality of the MCC; (3) applying the proposed architecture to identify reliable dynamical models to be used in advanced control strategies, such as the predictive controllers; (4) evaluating the feasibility to apply the FWNN-MCC as an inferential system to estimate chemical compositions, calibrate sensors [62], and fault diagnosis, among others.

Acronyms

- A/D: Analog-to-digital
- ANFIS: Adaptive Neurofuzzy Inference System
- ANN: Artificial Neural Network
- APRBS: Amplitude Modulated Pseudorandom Binary Sequence

ARMAX:	AutoRegressive, Moving Average, eXogenous input model
ARX:	AutoRegressive, eXogenous input model
BP:	Error backpropagation algorithm
D/A:	Digital-to-analog
FIR:	Finite Impulse Response model
FWNN:	Fuzzy Wavelet Neural Network
FWNN-MCC:	FWNN obtained using MCC
FWNN-MSE:	FWNN obtained using MSE
MCC:	Maximum Correntropy Criterion
MF:	Membership function
MLP:	Multilayer Perceptron network
MSE:	Mean Squared Error
NARMAX:	Nonlinear AutoRegressive, Moving Average, eXogenous input model
NARX:	Nonlinear AutoRegressive, eXogenous input model
NFIR:	Nonlinear Finite Impulse Response model
NOE:	Nonlinear Output Error model
NSSIF:	Nonlinear State Space Innovations Form model
OE:	Output Error model
PCA:	Principal component analysis
RBF:	Radial Basis Function network
SSIF:	State Space Innovations Form model
WNN:	Wavelet Neural Network.

Conflict of Interests

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

References

- [1] X. Lu, K. L. V. Iyer, K. Mukherjee, and N. C. Kar, "A dual purpose triangular neural network based module for monitoring and protection in Bi-directional off-board level-3 charging of EV/PHEV," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1670–1678, 2012.
- [2] Z. Tian and M. J. Zuo, "Health condition prediction of gears using a recurrent neural network approach," *IEEE Transactions on Reliability*, vol. 59, no. 4, pp. 700–705, 2010.
- [3] Y.-Y. Lin, J.-Y. Chang, and C.-T. Lin, "Identification and prediction of dynamic systems using an interactively recurrent self-evolving fuzzy neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 2, pp. 310–321, 2013.
- [4] Z. Shao, Y. Zhan, and Y. Guo, "Fuzzy neural network-based model reference adaptive inverse control for induction machines," in *Proceedings of the International Conference on Applied Superconductivity and Electromagnetic Devices (ASEMD '09)*, pp. 56–59, Chengdu, China, 2009.
- [5] S. Yilmaz and Y. Oysal, "Fuzzy wavelet neural network models for prediction and identification of dynamical systems," *IEEE Transactions on Neural Networks*, vol. 21, no. 10, pp. 1599–1609, 2010.
- [6] R. H. Abiyev and O. Kaynak, "Fuzzy wavelet neural networks for identification and control of dynamic plants—a novel structure and a comparative study," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 8, pp. 3133–3140, 2008.
- [7] R. J. Bessa, V. Miranda, and J. Gama, "Entropy and correntropy against minimum square error in offline and online three-day ahead wind power forecasting," *IEEE Transactions on Power Systems*, vol. 24, no. 4, pp. 1657–1666, 2009.
- [8] H. Qu, W. Ma, J. Zhao, and T. Wang, "Prediction method for network traffic based on maximum correntropy criterion," *China Communications*, vol. 10, no. 1, pp. 134–145, 2013.
- [9] O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*, Springer, Berlin, Germany, 2001.
- [10] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice-Hall, Upper Saddle River, NJ, USA, 1999.
- [11] Y. Zhang and L. Wu, "Crop classification by forward neural network with adaptive chaotic particle swarm optimization," *Sensors*, vol. 11, no. 5, pp. 4721–4743, 2011.
- [12] Y. Zhang, S. Wang, G. Ji, and P. Phillips, "Fruit classification using computer vision and feedforward neural network," *Journal of Food Engineering*, vol. 143, pp. 167–177, 2014.
- [13] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Transactions on Neural Networks*, vol. 1, no. 1, pp. 4–27, 1990.
- [14] A. Banakar and M. F. Azeem, "Identification and prediction of nonlinear dynamical plants using TSK and wavelet neuro-fuzzy models," in *Proceedings of the 3rd IEEE Conference on Intelligent Systems*, pp. 617–620, London, UK, September 2006.
- [15] T. Kara and I. Eker, "Experimental nonlinear identification of a two mass system," in *Proceedings of the IEEE Conference on Control Applications (CCA '03)*, vol. 1, pp. 66–71, Istanbul, Turkey, 2003.
- [16] M. O. Efe, "A comparison of ANFIS, MLP and SVM in identification of chemical processes," in *Proceedings of the IEEE Control Applications & Intelligent Control*, pp. 689–694, St. Petersburg, Russia, 2009.
- [17] X. Xue, J. Lu, and W. Xiang, "Nonlinear system identification with modified differential evolution and RBF networks," in *Proceedings of the 5th IEEE International Conference on Advanced Computational Intelligence (ICACI '12)*, pp. 332–335, Nanjing, China, October 2012.
- [18] Q. Zhang and A. Benveniste, "Wavelet networks," *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 889–898, 1992.
- [19] S. A. Billings and H.-L. Wei, "A new class of wavelet networks for nonlinear system identification," *IEEE Transactions on Neural Networks*, vol. 16, no. 4, pp. 862–874, 2005.
- [20] F.-J. Lin, P.-H. Shen, and Y.-S. Kung, "Adaptive wavelet neural network control for linear synchronous motor servo drive," *IEEE Transactions on Magnetics*, vol. 41, no. 12, pp. 4401–4412, 2005.
- [21] J.-S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.
- [22] L. Zhao, J.-P. Zhang, J. Yang, and Y. Chu, "Software reliability growth model based on fuzzy wavelet neural network," in *Proceedings of the 2nd International Conference on Future Computer and Communication (ICFCC '10)*, pp. V1664–V1668, IEEE, Wuhan, China, May 2010.
- [23] J.-R. Song and H.-B. Shi, "Dynamic system modeling based on wavelet recurrent fuzzy neural network," in *Proceedings of the 7th International Conference on Natural Computation (ICNC '11)*, vol. 2, pp. 766–770, Shanghai, China, July 2011.
- [24] C. H. Lu, "Wavelet fuzzy neural networks for identification and predictive control of dynamic systems," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 7, pp. 3046–3058, 2011.

- [25] D. W. C. Ho, P.-A. Zhang, and J. Xu, "Fuzzy wavelet networks for function learning," *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 1, pp. 200–211, 2001.
- [26] A. Ebadat, N. Noroozi, A. A. Safavi, and S. H. Mousavi, "New fuzzy wavelet network for modeling and control: the modeling approach," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 8, pp. 3385–3396, 2011.
- [27] S. H. Mousavi, N. Noroozi, A. . Safavi, and A. Ebadat, "Modeling and control of nonlinear systems using novel fuzzy wavelet networks: the output adaptive control approach," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 9, pp. 3798–3814, 2011.
- [28] S. Ganjefar and M. Tofighi, "A fuzzy wavelet neural network stabilizer design using genetic algorithm for multi-machine systems," *Przełąd Elektrotechniczny*, vol. 89, no. 5, pp. 19–25, 2013.
- [29] L. L. S. Linhares, J. M. Araújo Jr., F. M. U. Araújo, and T. Yoneyama, "A nonlinear system identification approach based on fuzzy wavelet neural network," *Journal of Intelligent and Fuzzy Systems*, 2014.
- [30] Y. Liu and J. Chen, "Correntropy-based kernel learning for nonlinear system identification with unknown noise: an industrial case study," in *Proceedings of the 10th IFAC Symposium on Dynamics and Control of Process Systems (DYCOPS '13)*, vol. 10, pp. 361–366, Mumbai, India, December 2013.
- [31] J. C. Munoz and J. Chen, "Removal of the effects of outliers in batch process data through maximum correntropy estimator," *Chemometrics and Intelligent Laboratory Systems*, vol. 111, no. 1, pp. 53–58, 2012.
- [32] T. Söderström, "System identification for the errors-in-variables problem," *Transactions of the Institute of Measurement and Control*, vol. 34, no. 7, pp. 780–792, 2012.
- [33] S. Khatibisepehr and B. Huang, "Dealing with irregular data in soft sensors: Bayesian method and comparative study," *Industrial and Engineering Chemistry Research*, vol. 47, no. 22, pp. 8713–8723, 2008.
- [34] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK, 1995.
- [35] V. Miranda, C. Cerqueira, and C. Monteiro, "Training a FIS with epsu under an entropy criterion for wind power prediction," in *Proceedings of the International Conference on Probabilistic Methods Applied to Power Systems (PMAPS '06)*, pp. 1–8, Stockholm, Sweden, 2006.
- [36] I. Santamaría, P. P. Pokharel, and J. C. Principe, "Generalized correlation function: definition, properties, and application to blind equalization," *IEEE Transactions on Signal Processing*, vol. 54, no. 6, pp. 2187–2197, 2006.
- [37] S. Zhao, B. Chen, and J. C. Principe, "Kernel adaptive filtering with maximum correntropy criterion," in *Proceedings of the International Joint Conference on Neural Network (IJCNN '11)*, pp. 2012–2017, San Jose, Calif, USA, August 2011.
- [38] J. C. Principe, *Information Theoretic Learning: Rényi's Entropy and Kernel Perspectives*, Springer, 2010.
- [39] A. I. Fontes, A. D. M. Martins, L. F. Silveira, and J. Principe, "Performance evaluation of the correntropy coefficient in automatic modulation classification," *Expert Systems with Applications*, vol. 42, no. 1, pp. 1–8, 2014.
- [40] A. I. R. Fontes, P. T. V. Souza, A. D. D. Neto, A. M. Martins, and L. F. Q. Silveira, "Classification system of pathological voices using correntropy," *Mathematical Problems in Engineering*, vol. 2014, Article ID 924786, 7 pages, 2014.
- [41] R. He, B.-G. Hu, W.-S. Zheng, and X.-W. Kong, "Robust principal component analysis based on maximum correntropy criterion," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1485–1494, 2011.
- [42] Y. Liu and J. Chen, "Correntropy-based kernel learning for nonlinear system identification with unknown noise: an industrial case study," in *Proceedings of the 10th IFAC Symposium on Dynamics and Control of Process Systems*, pp. 361–366, December 2013.
- [43] W. Liu, P. P. Pokharel, and J. C. Principe, "Correntropy: properties and applications in non-Gaussian signal processing," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5286–5298, 2007.
- [44] S. Zhao, B. Chen, and J. C. Principe, "An adaptive kernel width update for correntropy," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '12)*, pp. 1–5, Brisbane, Australia, 2012.
- [45] M. C. Jones, J. S. Marron, and S. J. Sheather, "A brief survey of bandwidth selection for density estimation," *Journal of the American Statistical Association*, vol. 91, no. 433, pp. 401–407, 1996.
- [46] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*, vol. 3, CRC Press, New York, NY, USA, 1986.
- [47] A. W. Bowman, "An alternative method of cross-validation for the smoothing of density estimates," *Biometrika*, vol. 71, no. 2, pp. 353–360, 1984.
- [48] D. W. Scott and G. R. Terrell, "Biased and unbiased cross-validation in density estimation," *Journal of the American Statistical Association*, vol. 82, no. 400, pp. 1131–1146, 1987.
- [49] N. Terzija and H. McCann, "Wavelet-based image reconstruction for hard-field tomography with severely limited data," *IEEE Sensors Journal*, vol. 11, no. 9, pp. 1885–1893, 2011.
- [50] A. Bouzida, O. Touhami, R. Ibtouen, A. Belouchrani, M. Fadel, and A. Rezzoug, "Fault diagnosis in industrial induction machines through discrete wavelet transform," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 9, pp. 4385–4395, 2011.
- [51] J. Gao, Z. Leng, Y. Qin, Z. Ma, and X. Liu, "Short-term traffic flow forecasting model based on wavelet neural network," in *Proceedings of the 25th Chinese Control and Decision Conference (CCDC '13)*, pp. 5081–5084, Guiyang, China, May 2013.
- [52] J. J. Cordova, W. Yu, and X. Li, "Haar wavelet neural networks for nonlinear system identification," in *Proceedings of the 6th IEEE Multi-Conference on Systems and Control (MSC '12)*, pp. 276–281, Dubrovnik, Croatia, October 2012.
- [53] R. K. Galvão, V. M. Bécerra, J. M. Calado, and P. M. Silva, "Linear-wavelet networks," *International Journal of Applied Mathematics and Computer Science*, vol. 14, no. 2, pp. 221–232, 2004.
- [54] M. Davanipoor, M. Zekri, and F. Sheikholeslam, "Fuzzy wavelet neural network with an accelerated hybrid learning algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 3, pp. 463–470, 2012.
- [55] V. V. J. Rajapandian and N. Gunaseeli, "Modified standard back propagation algorithm with optimum initialization for feed forward neural networks," *International Journal of Imaging Science and Engineering*, vol. 1, no. 3, pp. 86–89, 2007.
- [56] T. Kathirvalavakumar and P. Thangavel, "A modified backpropagation training algorithm for feedforward neural networks," *Neural Processing Letters*, vol. 23, no. 2, pp. 111–119, 2006.

- [57] S. Abid, R. Fnaicch, and M. Najim, "A fast feedforward training algorithm using a modified form of the standard backpropagation algorithm," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 424–430, 2001.
- [58] M. Davanipoor, M. Zekri, and F. Sheikholeslam, "The preference of fuzzy wavelet neural network to ANFIS in identification of nonlinear dynamic plants with fast local variation," in *Proceedings of the 18th Iranian Conference on Electrical Engineering (ICEE '10)*, pp. 605–609, Isfahan, Iran, May 2010.
- [59] E. Soria, J. D. Martin, and P. J. G. Lisboa, "Classical training methods," in *Metaheuristic Procedures for Training Neural Networks*, E. Alba and R. Marti, Eds., pp. 25–32, Springer, New York, NY, USA, 2006.
- [60] A. Singh and J. C. Príncipe, "Information theoretic learning with adaptive kernels," *Signal Processing*, vol. 91, no. 2, pp. 203–213, 2011.
- [61] C. A. G. Fonseca, *Estrutura ANFIS Modificada para identificação e Controle de Plantas com Ampla Faixa de Operação e não Linearidade Acentuada [Ph.D. thesis]*, Universidade Federal do Rio Grande do Norte (UFRN), Natal, Brazil, 2012.
- [62] J. M. de Araújo Jr., J. M. P. de Menezes, A. A. M. de Albuquerque, O. D. M. Almeida, and F. M. U. de Araújo, "Assessment and certification of neonatal incubator sensors through an inferential neural network," *Sensors*, vol. 13, no. 11, pp. 15613–15632, 2013.

Research Article

A Self-Adaptive Hidden Markov Model for Emotion Classification in Chinese Microblogs

Li Liu,^{1,2,3} Dashi Luo,² Ming Liu,⁴ Jun Zhong,² Ye Wei,² and Letian Sun²

¹School of Software Engineering, Chongqing University, Chongqing 401331, China

²School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China

³School of Computing, National University of Singapore, Singapore 117417

⁴Faculty of Computer and Information Science, Southwest University, Chongqing 400715, China

Correspondence should be addressed to Ming Liu; mingliu@swu.edu.cn

Received 24 September 2014; Accepted 18 December 2014

Academic Editor: Praveen Agarwal

Copyright © 2015 Li 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.

Microblogging is increasingly becoming one of the most popular online social media for people to express ideas and emotions. The amount of socially generated content from this medium is enormous. Text mining techniques have been intensively applied to discover the hidden knowledge and emotions from this huge dataset. In this paper, we propose a modified version of hidden Markov model (HMM) classifier, called self-adaptive HMM, whose parameters are optimized by Particle Swarm Optimization algorithms. Since manually labeling large-scale dataset is difficult, we also employ the entropy to decide whether a new unlabeled tweet shall be contained in the training dataset after being assigned an emotion using our HMM-based approach. In the experiment, we collected about 200,000 Chinese tweets from Sina Weibo. The results show that the F -score of our approach gets 76% on happiness and fear and 65% on anger, surprise, and sadness. In addition, the self-adaptive HMM classifier outperforms Naive Bayes and Support Vector Machine on recognition of happiness, anger, and sadness.

1. Introduction

In the recent years, online social media, such as Microblogging, has generated enormous content on the world wide web. Microblogs are extremely limited in the length of 140 characters. This allows users to easily update a Microblog or receive updates from mobile devices, such as cell phones. Twitter has grown to one of the most popular Microblogging websites and generated hundreds of millions of updates per day. Similar to Twitter, Weibo is a Microblogging website in China. In the past few years, it has become an increasingly important source of online social media information with its population of Chinese users growing rapidly to 309 million in 2012 (http://news.xinhuanet.com/english/sci/2013-01/15/c_132104473.htm), as well as with more than 1,000 tweets generated in every second.

The emotional states of the users are able to be inferred from these large numbers of short tweets. Emotions in tweets play a significant role in many fields. The stock market and other socioeconomic phenomena could be predicted by using

emotion analysis of the Twitter users [1]. Even the gross happiness of a community or a country could be estimated from Twitter [2].

Text mining on Chinese tweets is a challenging work. First, word segmentation in Chinese is more difficult than in English since there is no space between Chinese characters and it requires disambiguating segmentation strings. Second, like English Twitter, new words are coming out every day and it is difficult to develop a system to recognize the unknown emotional words. Third, words are ambiguous in various contexts, especially those emotional words. Most of the recent sentiment analysis approaches [3, 4] for Chinese tweets employ emotional word count as main feature and build emotional word dictionary for inference.

Considering these issues, we propose a self-adaptive hidden Markov model (HMM) based method to perform the emotion analysis for Chinese tweets. Our method implements a self-adaptive mechanism, which learns the parameters of HMM models and appends new recognized emotional words or sentences into emotional word dictionary, to deal

with the issues of the sentiment ambiguities of words and the generation of new words. The main contributions of this paper are shown as follows. (a) More fine grained emotional categories are recognized. We employ the well-known six basic emotional categories defined by Ekman [5]: happiness, sadness, fear, anger, disgust, and surprise, which are more intuitive and useful than traditional sentiment analysis categories: positive, negative, and neutral categories. (b) More useful features, other than word count, are defined. Our method employs the category-based features extracted from the short sentences to train our proposed HMM models. (c) A self-adaptive mechanism is used to test on a real dataset collected from Sina Weibo.

The rest of the paper is organized as follows. In Section 2, we survey the related works of emotion analysis either in Twitter or in Weibo and provide background on the concepts and methods related to emotion analysis for tweets, especially for Chinese tweets. In Section 3, we describe our proposed self-adaptive HMM-based method. In Section 4, we illustrate the dataset and the experiment setup and present results from our study. We discuss limitations of the method and future work and conclude in Section 5.

2. Background

2.1. Related Work. Weibo plays a significant role in people's lives; therefore its opinion mining and emotion analysis become interesting researches. There are a lot of researches on Twitter emotions abroad. Spencer and Uchyigit [6] identified subjective tweets and detected their emotion polarity by using Naive Bayes. In their test using bigrams without part of speech (POS) tags has a high accuracy of 52.31%. In the study carried out by Pak and Paroubek [7], they utilized Naive Bayes to classify tweets into positive, negative, and neutral. It is shown that using the presence of an n -gram as a binary feature yielded the best results. A research [8] has shown that machine learning algorithms (Naive Bayes, Maximum Entropy, and SVM) had high accuracy rate with above 80% when sentiment categories consisted only of positive and negative. In the detection based on Twitter corpus, the authors explored various strategies of selecting features and found that bigram features outperform unigram and POS.

However, studies of Chinese Weibo emotions grow just a few years. Zhao et al. [9] used Naive Bayes model to train on emoticon features; they classified Chinese tweets to four categories of emotions (i.e., angry, disgusting, joyful, and sad), with an empirical precision of 64.3%. Yuan and Purver [10] classified Chinese tweets to six emotions which are defined by Ekman as well. In their experiments, character-based features achieved 80% accuracies for "happiness" and "fear" based on the SVM, but there is insufficient analysis to the other four emotions.

Our self-adaptive HMM is to recognize more grained emotional categories by category-based features. Besides, the self-adaptive mechanism can continually enhance the recognition accuracy.

2.2. Emotion Models. Two models are normally used to represent emotions: the categorical model and dimensional model [11]. The categorical models are based on the assumption that the emotional categories are distinct. Ekman defined six basic emotions: anger, disgust, fear, joy, sadness, and surprise, and found a high agreement of expressions in multiple culture groups in his study. D'Mello et al. [12] proposed five categories (boredom, confusion, delight, flow, and frustration) for describing the affect states in ITS interactions.

In the dimensional model, core affects represent emotions in a two- or three-dimensional space. A valence dimension represents positive and negative emotions on different ends of scale. The arousal dimension distinguishes excited states from calm states. Sometimes a third, dominance dimension is used to differentiate if the subject feels in control of the situation or not. For example, Positive Affect and Negative Affect Schedule (PANAS) [13] provides two opposite mood factors (positive and negative) which has been widely used for opinion mining.

In this study, the reason we employ Ekman's emotion model is that the six basic emotions are distinct expressions and contain people daily mood.

2.3. Chinese Text Preprocessing. Chinese, a unique language, differing from another language like English, is written without word delimiters. Therefore, its word segmentation is a significant task. Chinese text preprocessing is divided into two steps: word segmentation and stop words removing.

To split words, there are some Chinese lexical analysis systems (Institute of Computing Technology, Chinese Lexical Analysis System (ICTCLAS) [14], IKAnalyzer (<http://code.google.com/p/ik-analyzer/>), HTTPCWS (<http://code.google.com/p/httpcws/>), Simple Chinese Word Segmentation (SCWS) (<http://gc.codehum.com/p/libscws/>), which are developed to perform word segmentation task). The accuracy rates of those systems are all evaluated above 90 percent.

We utilized NLPPIR (namely, ICTCLAS 2013) as a word splitter in this study. Its latest lexical database contains new words often appearing in Weibo. It also can recognize bloggers' nickname. Furthermore, an adaptive word splitter can add new words in its lexical database, which can be applied on our self-adaptive mechanism to update word-emotional vocabulary.

For reducing dimension of feature space, the next important step is to remove stop words (quantifier, pronoun, digits, notations, etc., e.g., "hundred," "we," "3," and "*%"). Similarly, Weibo has many unique properties like usernames and usage of links that must be removed.

2.4. Features. In the field of document classification, each document is represented as a vector of term. Effective feature extraction is essential to make learning task effective. We adopt four features for our emotion classification model as follows.

Mutual Information (MI). MI is a useful information measure, which refers to the correlation between two sets of

events. MI reflects the relevance between terms and text categorization on emotions. Two events mutual information of term t and emotion c_i is defined as

$$\text{MI}(t, c_i) = \log \frac{P(t | c_i)}{P(t)}. \quad (1)$$

Chi-Square (CHI). CHI is a statistical method to measure the lack of independence between term t and emotion c_i . The higher the CHI values, the more dependence between them:

$$\chi(t, c_i) = \frac{(AD - BC)^2 \times N}{(A + B)(C + D)(A + C)(B + D)}, \quad (2)$$

where A represents presence of t and membership in c_i ; B represents presence of t and nonmembership in c_i ; C represents absence of t and membership in c_i ; D represents absence of t and nonmembership in c_i ; N is the total number of tweets.

Term Frequency-Inverse Document Frequency (TF-IDF). The main idea of TF-IDF is as follows: if a word or phrase in a tweet appears in a high frequency and rarely appears in other tweets; then it has a good ability to distinguish among emotions:

$$\text{TF-IDF}(t, c_i) = \text{TF}_{i,t} \times \text{IDF}_t, \quad (3)$$

where $\text{TF}_{i,t}$ represents the frequency of the term t in the emotion c_i ; the main idea of IDF_t is that if the term t rarely appears in other emotions, IDF_t can be an indicator feature for c_i . Their definition is

$$\text{TF}_{i,t} = \frac{N_{i,t}}{\sum_k N_{k,t}}; \quad \text{IDF}_t = \log \left(\frac{N}{n_t} + 0.01 \right), \quad (4)$$

where $N_{i,t}$ means the number of how many times t occurs in emotion c_i , N denotes the total number of tweets in the dataset, and n_t is the number of tweets where t appears.

Expected Cross Entropy (ECE). ECE reflects the probability distribution of the text category on emotions and distances between the probability distribution given term t . Its definition is

$$\text{ECE}(t, c_i) = P(c_i | t) \log \frac{P(c_i | t)}{P(c_i)}, \quad (5)$$

where $P(c_i | t)$ represents the probability of emotion c_i given term t and $P(c_i)$ represents the probability of tweets associated with emotion c_i .

2.5. Classification Methods

Hidden Markov Model (HMM). HMM is used to describe a Markov process with unknown parameters. It is difficult to determine implicit parameters of the process through observable parameters, which are then used to make further analysis.

An HMM describes two related discrete-time stochastic processes. The first process pertains to hidden state variables, denoting (S_1, S_2, \dots, S_u) , which emit observed variables with different probabilities, and the second process pertains to observed variables (y_1, y_2, \dots, y_u) .

Main parameters of HMM are the transition and emission probabilities [15]:

$P(S_k = s_p | S_{k-1} = s_q)$ (transition probabilities): it means current state is depend on the previous state s_q ;
 $P(y_k | S_k = s_p)$ (emission probabilities): observation symbol is released by current state s_p .

In our model, an HMM model $\lambda^{(c_i)}$ is built for each emotion c_i . The features extracted from a tweet T are observed variables, while each hidden state $S_k^{(c_i)}$ of $\lambda^{(c_i)}$ is considered as a state associated with the feature y_k . If the tweet T gets the highest probability on the model $\lambda^{(c_i)}$, it means that T is associated with emotion c_i .

3. Method

3.1. Features as Observed Variables. Feature extraction methods fall into two main categories: category-based extraction and global-based extraction. The features exacted by the latter methods can reflect the importance of words in global corpus but cannot be used to distinguish the differences between emotions. Therefore, we adopt category-based feature extraction methods in our models. All the selected features we introduced in Section 2.4 are category-based extracted features.

Suppose a set of u features extraction methods $\{F_1, F_2, \dots, F_u\}$. A tweet T is first divided into n terms (w_1, w_2, \dots, w_n) . Let x_{jk} be the k th feature of the word w_j extracted by the method F_k . Then, an intermediate $n \times u$ matrix of term-level features is obtained for calculating the tweet-level feature vector. For each emotion c_i ($1 \leq i \leq 6$), the tweet-level feature vector of the tweet T can be calculated as follows:

$$\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \rightarrow \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1u} \\ x_{21} & x_{22} & \cdots & x_{2u} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nu} \end{bmatrix} \rightarrow [y_1 \ y_2 \ \cdots \ y_u]^{(c_i)}, \quad (6)$$

where $y_k = \sum_{j=1}^n x_{jk}/n$, ($1 \leq k \leq u$). Actually, y_k is a mean value of k th features over all the words.

In our experiment, a tweet is mapped into a four-dimensional vector using the four features: MI, CHI, TF-IDF and ECE. For example, the tweet ‘‘How lovely!’’ is divided into two terms: ‘‘how’’ and ‘‘lovely’’ in Chinese. The features of the terms on happiness and anger are calculated, respectively, as listed in Table 1.

Therefore, the tweet’s feature vector is $[0.0022, 0.0107, 0.0052, 0.0016]$ in happiness and $[-2.8E - 4, 3.9E - 4, 1.1E - 4, -3.2E - 5]$ in anger.

Input: feature vector $[y_1 \ y_2 \ \dots \ y_u]^{(c_i)}$
Output: probability $P([y_1 \ y_2 \ \dots \ y_u]^{(c_i)} | \lambda^{(c_i)})$
for $k = 1$ **to** u **do**
 Calculate $P(y_k | S_k)^{(c_i)}$
end for
Calculate $P([y_1 \ y_2 \ \dots \ y_u]^{(c_i)} | \lambda^{(c_i)})$ using forward algorithm.

ALGORITHM 1: $\lambda^{(c_i)}$ model algorithm.

TABLE 1: The features of the terms “how” and “lovely” on happiness and anger, respectively.

Term	Emotions	MI	CHI	TF-IDF	ECE
How	Happiness	0.0014	0.0147	0.00088	0.00059
	Anger	-0.00056	0.00063	0.00023	-0.000065
Lovely	Happiness	0.0030	0.0067	0.0095	0.0027
	Anger	0	0.00014	0	0

3.2. HMM-Based Emotional Classification Model. In our case each hidden state is supposed to emit a value, and the whole model generates the sequence of values that constitutes the tweet’s feature vector. States are considered to be a set of values that represent the best emotional category. There is a one-to-one mapping between HMM states and tweet features, which requires hidden states transition to be stationary and the states to begin at the first state s_1 . When the classifier works, transition probabilities indicate features of test tweets being drawn closer to the emotion, which is given as follows:

$$P(S_k = s_p | S_{k-1} = s_q) = \begin{cases} 1, & (p = q + 1) \\ 0, & (p \neq q + 1) \end{cases} \quad (7)$$

The emission probability $P(y_k | S_k)$ also can be calculated by the known state S_k , feature y_k , and training data. Our calculation method is given at the end of this chapter.

Algorithm 1 depicts the procedure of building an HMM model $\lambda^{(c_i)}$ for emotion c_i . The tweet-level feature vector is denoted as $[y_1 \ y_2 \ \dots \ y_u]^{(c_i)}$.

We construct an HMM model for each of the six emotions. When a new tweet arrives, the probabilities of the tweet are calculated on the six models, respectively. The tweet is labeled with the emotion whose model is associated with the maximum probability.

A Strategy of Calculating $P(y_k | S_k)^{(c_i)}$. To find a way to calculate the $P(y_k | S_k)$ of emotion c_i , denoted as $P(y_k | S_k)^{(c_i)}$, we attempt to use Jaccard similarity measure [16] to test correlation between value y_k and state S_k :

$$P(y_k | S_k)^{(c_i)} = J(y_k | S_k)^{(c_i)} = \frac{M_{11}}{M_{11} + M_{10} + M_{01}}, \quad (8)$$

where M_{11} represents the number of tweets, which contains both y_k and S_k in c_i ; M_{01} represents the number of tweets,

which contains only y_k in c_i ; and M_{10} represents the number of tweets, which contains only S_k in c_i . In order to check whether emotion c_i associates to observation y_k or state S_k , we introduce an associated factor, denoted by δ_k . Let y'_k be the feature extracted from training data. The following describes the association:

$$|y'_k - y_k| \leq \delta_k \quad \text{or} \quad |y'_k - S_k| \leq \delta_k. \quad (9)$$

We assume that observation y_k or state S_k is associated with c_i if the above inequality is met.

3.3. Self-Adaptive Mechanism

3.3.1. Parameters Computed by PSO. To build an excellent HMM classifier, an important problem we have to solve is to find optimized sequences of HMM states. There are a variety of strategies we can utilize for optimizing state parameters.

Particle Swarm Optimization (PSO) algorithm [17] is a population based stochastic optimization. In our algorithm, each particle represents a candidate solution of an HMM parameter. These particles move around in the search-space. Through particle’s local best known position, it is guided toward the best known positions. Eventually this iterative method can find best parameters. Compared to Genetic Algorithm [18], PSO has simple rules and more powerful ability of global optimization that has been good applications in our study:

$$v[] = W \times v[] + c_1 \times r \times (pbest[] - present[]) \quad (a)$$

$$+ c_2 \times R \times (gbest[] - present[])$$

$$present[] = present[] + v[], \quad (b)$$

where $[]$ indicates that its corresponding variable is a vector. The variables r and R are random numbers between $[0, 1]$. $pbest[]$ records the individual extremes and $gbest[]$ records the global extremes. The constant W is the inertia weight, and c_1 and c_2 present acceleration constants. Moreover, (a) updates velocities of particles according to previous velocity and the distance to the best particle. The equation (b) updates the particles’ position $present[]$ according to its previous position and current velocity.

PSO Parameter Settings. Different PSO parameters may have a large impact on optimization performance. Following are guidelines to help us select PSO parameters [19, 20].

```

Input: training data pool  $L$ , test data pool  $U$ , query strategy  $\phi(\bullet)$ , query batch size  $B$ 
repeat
  for  $i = 1$  to  $|C|$  do
    Optimized  $\lambda^{(c_i)}$  by using current  $L$  and PSO algorithm;
  end for
  for  $b = 1$  to  $B$  do
     $T_b^* = \arg \max_{T \in U} \phi(T)$ ;
    Move  $T_b^*$  from  $U$  to  $L$ ;
  end for
until some stopping criterion.

```

ALGORITHM 2: Algorithm of pool-based feedback.

v_{\max} : decide the granularity of searching space. We set it through values of training data.

W : keep the particles motion inertia. We set the inertia weight at 0.8.

c_1, c_2 : represent accelerated weight of pushing each particle to $pbest[]$ and $gbest[]$. We set both weights to the value of 2.

Fitness Function. We use fitness function to find two kinds of parameters in HMM models, that is, the optimized associated factor δ_k and the $S_k^{(c_i)}$, which is the k th hidden states of the HMM model $\lambda^{(c_i)}$. Accordingly, the fitness function can be defined as follows:

$$\text{fitness}(\delta_k, S_k^{(c_1)}, \dots, S_k^{(c_u)}) = F_1\text{-Measure}, \quad (1 \leq k \leq u), \quad (10)$$

where F_1 -Measure is the metrics (in Section 4.3) of classifiers' accuracy. There are a total number of $7 \times u$ parameters that need to be searched. Since finding the whole set of parameters is time intensive by PSO, we divide the set of parameters into u independent parts. Seven parameters $(\delta_k, S_k^{(c_1)}, \dots, S_k^{(c_u)})$, $(1 \leq k \leq u)$, are learned for each part according to the fitness function.

3.3.2. Feedback. Since it is time-consuming and expensive to manually label all the tweets, we introduce a feedback method that can automatically decide whether an unlabeled tweet shall be chosen and contained in a training data pool after it is assigned with an emotion by our HMM model. Unlike the strategy (Lewis and Catlett [21], Scheffer et al. [22]), which merely concerns the assigned emotions, a suitable strategy is to compute the entropy of a tweet T to identify the discrimination of emotions; that is, an emotion is more discriminating than all other emotions on the tweet T . Entropy is an information-theoretic measure and its formula is given as follows:

$$\phi(T) = -\sum_i P(c_i | T) \log P(c_i | T), \quad (11)$$

where $P(c_i | T)$ means the probability of the tweet T recognized as emotion c_i . The less the $\phi(T)$, the more the

certainty about the tweet T on the emotion c_i . The algorithm of pool-based feedback is shown in Algorithm 2 to decide whether a tweet should be contained in the training dataset.

4. Experiments

4.1. Dataset. Since there are no public datasets of Chinese tweets associated with emotions, many studies employ emoticons to label tweets [7, 8, 10]. However, only a small percentage of Chinese users post tweets with emoticons. Besides, several emoticons cannot find their corresponding emotions. Hence, we build our own dataset as follows.

- (i) About 200,000 tweets were collected through Sina API.
- (ii) For each emotion, more than twelve seed terms were chosen for term-level feature extraction. We refer interested readers to Aman's annotation scheme [23] for seed term selection. For instance, a set of seed terms on happiness may contain "enjoy" and "pleased."
- (iii) Manually screening. Not all tweets are associated with a corresponding emotion. We asked ten students in computer science to choose good indicator tweet for the six emotions.

Notice that duplicate tweets are removed. Eventually, we selected 271, 159, 251, 229, 118, and 194 tweets on happiness, anger, surprise, fear, sadness, and disgust, respectively.

4.2. Setup. The procedure of emotion recognition for a tweet is shown as follows:

- (a) preprocess dataset;
- (b) build emotion lexicons; each lexicon contains a feature vector of terms extracted by feature extraction methods;
- (c) calculate feature vectors of the whole dataset;
- (d) optimize hidden states of each $\lambda^{(c_i)}$ by PSO;
- (e) for each $\lambda^{(c_i)}$ ($c_i \in |C|$), calculate feature vector $[y_1 \ y_2 \ \dots \ y_u]^{(c_i)}$ of T in emotion c_i and obtain

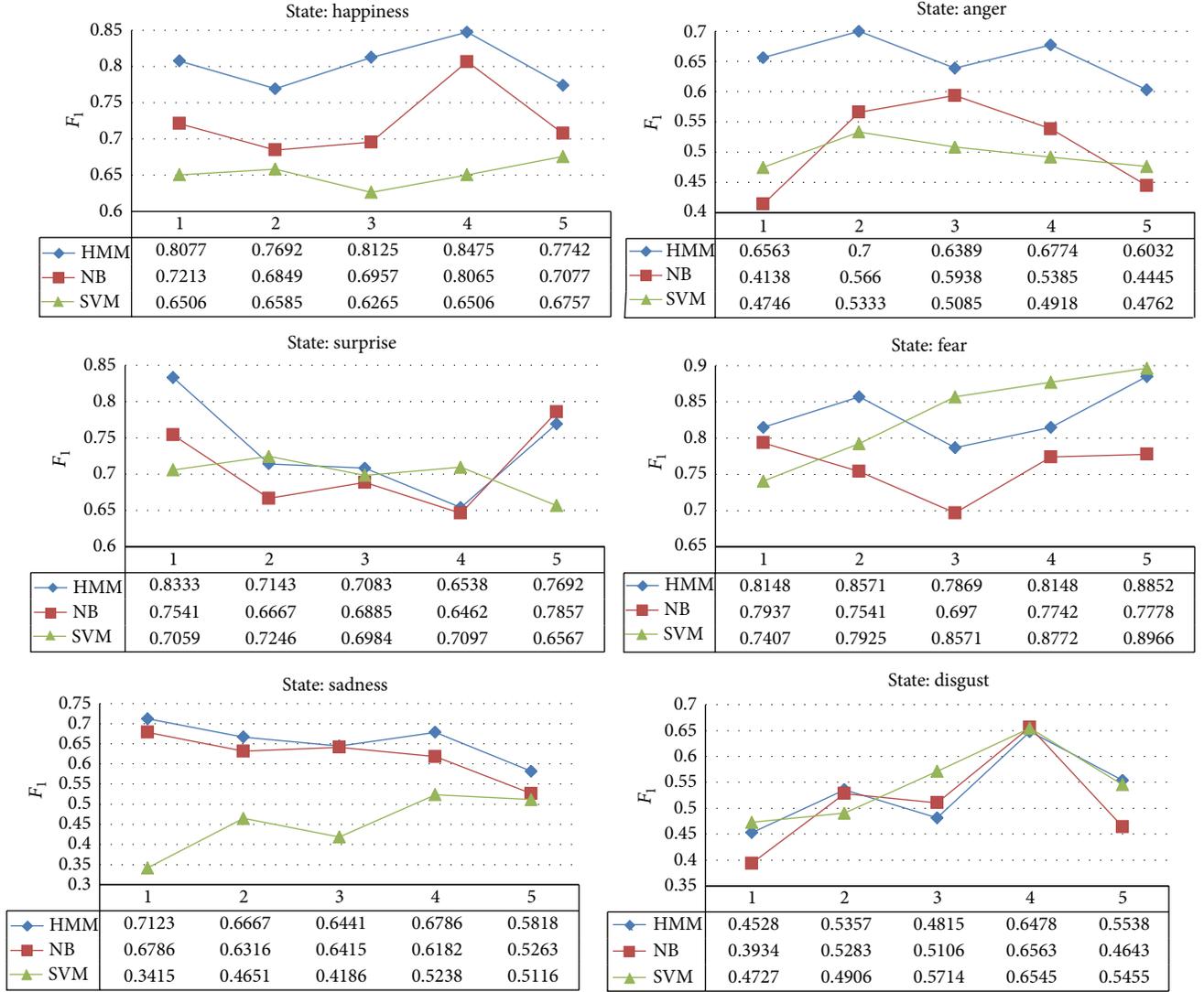


FIGURE 1: Performance comparison of three approaches on the six emotions, respectively. The x-axis means the number of test runs.

the output value $P([y_1 \ y_2 \ \dots \ y_u]^{(c_i)} \mid \lambda^{(c_i)})$ through model $\lambda^{(c_i)}$;

(f) return $c^* = \operatorname{argmax}_{c_i \in C_i} \{P([y_1 \ y_2 \ \dots \ y_u] \mid \lambda^{(c_i)})\}$.

To evaluate the performance of each of the HMM models, we randomly select several hand-annotated tweets. The test dataset contains 30 tweets for each emotion. Each test run is executed five times. We use the average results for our evaluation.

4.3. Evaluation Metrics. Precision, recall, and F -measure are the most commonly used evaluation methods for text classification tasks [7]. We employ the three metrics in our experiment. For each emotion c_i , precision and recall are defined as

$$\text{Precision} = \frac{a}{a+c}; \quad \text{Recall} = \frac{a}{a+b}, \quad (12)$$

where a represents the number of tweets that are correctly recognized; b represents the number of tweets that are falsely recognized as c_i ; c represents the number of tweets that are actually associated with c_i , but are recognized as another emotion.

To balance precision and recall rates, F_1 is defined as

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (13)$$

4.4. Evaluation

4.4.1. Compare with Another Classifier. We compared our approach with two well-known classifiers, that is, Naive Bayes and Support Vector Machine (SVM) [24]. They are often used for sentiment classification in literatures because of their easiness of implementation.

As shown in Figure 1, our HMM-based approach outperforms Naive Bayes and SVM on happiness, anger, and

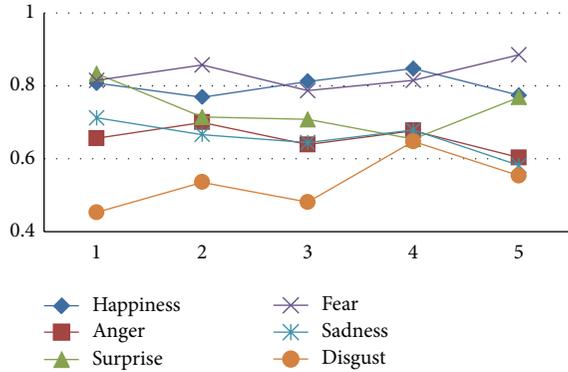


FIGURE 2: The accuracy of our HMM-based approach on the six emotions. The x -axis means the number of test runs.

sadness. The performances produced by all these classifiers have less difference on the other three emotions.

The results also show that, for the emotions of happiness, surprise, and fear, all the three classifiers get good performance. The F -measure is greater than 65%. Furthermore, the F -measure exceeds 78% on fear. None of the three classifiers recognize disgust accurately.

4.4.2. The Comparison of Six Emotions. Figure 2 shows the classification results of the six emotions using our HMM-based approach. We find that our approach gets the best accuracy over 76% on happiness and fear. It also gets a good performance on anger, surprise, and sadness with an average accuracy of 65%.

4.4.3. Analysis of HMM Results. We also attempt to analyze the reasons for false recognition on emotions. Twitters with specific characteristics may lead to false recognition using our HMM-base approach.

- (i) Some twitters may contain multiple emotions. For example, "Wow, I'm so smart!" (original sentence is in Chinese) contains both happiness and surprise. Multiple emotional tweets may cause false recognition. It also explains why our approach gets low F -score on disgust, which is often falsely recognized as anger.
- (ii) A number of tweets contain new words that can not be recognized.

In addition, puns and polysemous words are significant factors but rather difficult to be recognized. According to these reasons, our HMM-based approach can be improved concerning the above characteristics in our future experiments.

5. Conclusion

In this paper, we present an approach to extract features using the four methods, that is, MI, TF-IDF, CHI, and ECE. Our classifier is based on HMM, in which hidden states are found by PSO algorithm. Since manually labeling large-scale dataset is difficult, we employ the entropy to decide whether a new

unlabeled tweet shall be contained in the training dataset after being assigned an emotion using our HMM-based approach. The experimental results show that HMM outperforms SVM and NB, especially on happiness, anger, and sadness. In terms of the recognition precisions on the six emotions, HMM gets better performance on happiness and fear than on anger, surprise, and sadness.

In the future, we will optimize HMM to accurately recognize twitters associated with other emotions and automatically add new words for emotional seed terms selection. Moreover, self-adaptive mechanism will be implemented in our HMM model.

Conflict of Interests

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

Acknowledgments

This work is supported by Cuiying Grant of China Telecom, Gansu Branch (Grant no. lzudxcy-2013-3), Science and Technology Planning Project of Chengguan District, Lanzhou (Grant no. 2013-3-1), and Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry (Grant no. 44th).

References

- [1] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.
- [2] D. Quercia, J. Ellis, L. Capra, and J. Crowcroft, "Tracking 'gross community happiness' from tweets," in *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW '12)*, pp. 965–968, ACM, February 2012.
- [3] X. Lou, Y. Chai, H. Zan, R. Xu, Y. Han, and K. Zhang, "Research on micro-blog sentiment analysis," in *Chinese Lexical Semantics*, pp. 466–479, Springer, Berlin, Germany, 2013.
- [4] A. Cui, H. Zhang, Y. Liu, M. Zhang, and S. Ma, "Lexicon-based sentiment analysis on topical chinese microblog messages," in *Semantic Web and Web Science*, Springer Proceedings in Complexity, pp. 333–344, Springer, New York, NY, USA, 2013.
- [5] P. Ekman, "Universals and cultural differences in facial expressions of emotion," in *Proceedings of the Nebraska Symposium on Motivation*, University of Nebraska Press, 1971.
- [6] J. Spencer and G. Uchyigit, "Sentimentor: sentiment analysis of twitter data," in *Proceedings of the 1st International Workshop on Sentiment Discovery from Affective Data (SDAD '12)*, M. Gaber, M. Cocea, S. Weibelzahl, E. Menasalvas, and C. Labbé, Eds., p. 56, CEUR, 2012.
- [7] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," in *Proceedings of the 7th Conference on International Language Resources and Evaluation (LREC '10)*, Valletta, Malta, May 2010.
- [8] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," Project Report CS224N, Stanford University, Stanford, Calif, USA, 2009.

- [9] J. Zhao, L. Dong, J. Wu, and K. Xu, "MoodLens: an emoticon-based sentiment analysis system for Chinese tweets," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12)*, pp. 1528–1531, August 2012.
- [10] Z. Yuan and M. Purver, "Predicting emotion labels for chinese microblog texts," in *Proceedings of the 1st International Workshop on Sentiment Discovery from Affective Data (SDAD '12)*, M. Gaber, M. Cocca, S. Weibelzahl, E. Menasalvas, and C. Labbé, Eds., pp. 40–47, CEUR, 2012.
- [11] R. A. Calvo and S. M. Kim, "Emotions in text: dimensional and categorical models," *Computational Intelligence*, vol. 29, no. 3, pp. 527–543, 2013.
- [12] S. D'Mello, A. Graesser, and R. W. Picard, "Toward an affect-sensitive autotutor," *IEEE Intelligent Systems*, vol. 22, no. 4, pp. 53–61, 2007.
- [13] D. Watson, L. A. Clark, and A. Tellegen, "Development and validation of brief measures of positive and negative affect: the panas scales," *Journal of Personality and Social Psychology*, vol. 54, no. 6, pp. 1063–1070, 1988.
- [14] H.-P. Zhang, H.-K. Yu, D.-Y. Xiong, and Q. Liu, "HHMM-based Chinese lexical analyzer ICTCLAS," in *Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing (SIGHAN '03)*, pp. 184–187, Sapporo, Japan, July 2003.
- [15] P. Frasconi, G. Soda, and A. Vullo, "Hidden Markov models for text categorization in multi-page documents," *Journal of Intelligent Information Systems*, vol. 18, no. 2-3, pp. 195–217, 2002.
- [16] P. Jaccard, *Nouvelles Recherches sur la Distribution Florale*, 1908.
- [17] 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.
- [18] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [19] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *Evolutionary Programming VII*, vol. 1447 of *Lecture Notes in Computer Science*, pp. 591–600, Springer, Berlin, Germany, 1998.
- [20] I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection," *Information Processing Letters*, vol. 85, no. 6, pp. 317–325, 2003.
- [21] D. D. Lewis and J. Catlett, "Heterogenous uncertainty sampling for supervised learning," in *Proceedings of the International Conference on Machine Learning*, vol. 94, pp. 148–156, 1994.
- [22] T. Scheffer, C. Decomain, and S. Wrobel, "Active hidden markov models for information extraction," in *Advances in Intelligent Data Analysis*, vol. 2189 of *Lecture Notes in Computer Science*, pp. 309–318, Springer, Berlin, Germany, 2001.
- [23] S. Aman and S. Szpakowicz, "Identifying expressions of emotion in text," in *Text, Speech and Dialogue*, pp. 196–205, Springer, Amsterdam, The Netherlands, 2007.
- [24] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, 1998.